République Algérienne Démocratique et Populaire
Ministère de l'Enseignement Supérieur et de la Recherche Scientifique
Université Mouloud Mammeri Tizi-Ouzou

Mémoire de fin d'étude

En vue de l'obtention de diplôme master II en informatique

Option : réseaux, mobilité et systèmes embarqués.

Thème

Etat de l'art des systèmes de détection d'intrusion dans les réseaux ah doc pour le protocole AODV

Proposé par : **Mme** *BOURAKACHE.G*

Réalisé par :

M. *CHIBANE FERHAT*
M. *CHABANE AHMED*

promotion 2013/2014

✁ Dédicaces ✁

A Mes chers parents,
A la mémoire de mon grand-père,
A mes frères et mes sœurs,
A ma bien aimée Nassira,
A mes neveux Axel et Meziane,
A tous mes amis (es),

Ahmed.

A Mes très chers parents,
A mes frères,
A ma bien aimée,
A Hanna et Dihia,
A tous mes amis (es),

Ferhat.

Remerciements

Nous tenons à exprimer notre profonde gratitude à notre promotrice, Madame Ghenima Bourkache pour nous avoir encadrés durant cette année, ainsi que pour ses conseils judicieux.

Nos remerciements vont également aux membres du jury pour l'honneur qu'ils nous font en acceptant d'examiner et de juger notre travail.

Nous remercions aussi tous ceux, et celles qui ont contribué de prés ou de loin pour l'accomplissement de ce modeste travail.

Liste des abréviations

SQL: *Structured Query Language*

SNMP: Sample Network Management Protocol

TTL: *Time To Live*

DNS: *Domain Name System*

TCP: *Transmission Control Protocol*

REQ: Request

VPN: Virtual Private Network

MANET: Mobile Ad Hoc Network

SSH: Secure Shell

IDS: Intrusion Detection System

DoS: denial of services

QoS: Quality of Service

AODV: Ad Hoc On Demand Distance Vector

ARAN: Authenticated Routing for Ad hoc Networks

DSR: Dynamic Source Routing

SLSP: Secure Link State Protocol

DARPA: Defense Advanced Research Projects Agency

IEEE: Institute of Electrical and Electronics Engineers

DSDV: Destination-Sequenced Distance-Vector

RREP: Route Response

RREQ: Route Request

RERR: Route Error

GSR: Global State Routing

FSR: Fisheye State Routing

CBRP: Cluster Based Routing Protocol

ARA: Ant-Colony-Based Routing Algorithm

MAC: Media Access Control

IP: Internet Protocol

Liste des figures :

Figure 1.1 : les niveaux de vulnérabilité dans un système informatique 5
Figure 2.1 : Cryptographie symétrique 22
Figure 2.2 : Cryptographie asymétrique – authentification 23
Figure 2.3 : Cryptographie asymétrique – authentification 24
Figure 2.4 : signature d'un message 26
Figure 2.5 : vérification de la signature d'un message 27
Figure 2.6 : contenue d'un certificat 28
Figure 3.1 : Architecture classique d'un IDS (Intrusion Detection System) 37
Figure 3.2 : Modèle simplifié d'un système de détection d'intrusions 38
Figure 3.3 : Classification des systèmes de détection d'intrusions 41
Figure 3.4 : Détection d'abus 42
Figure 3.5 : Détection d'anomalie 43
Figure 3.6 : Les emplacements possibles des NIDS (Network IDS) 49
Figure 4.1 : La topologie d'un réseau ad hoc a un instant donné 58
Figure 4.2 : Précision des informations d'un nœud dans FSR en utilisant la technique œil de poisson 60
Figure 4.3 : Découverte de route dans DSR 61
Figure 4.4 : exemple d'assemblage en clusters dans CBRP 62
Figure 4.5 : Recherche de nourriture par une colonie de fourmis 63
Figure 4.6 : Découverte de route dans Ant-AODV 64
Figure 5.1 : Une demande de route 68
Figure 5.2 : Réponse de route 69
Figure 5.3 : Exemple d'établissement de route entre 1 et 5 70
Figure 5.4 : Boucle de routage 75
Figure 5.5 : Attaque du tunnel 75
Figure 5.6 : Attaque Black Hole 78
Figure 5.7 : Attaques dans AODV MANETs 79
Figure 5.8 : Architecture du moniteur réseau 84

Figure 5.9 : Manipulation des attaques internes 85
Figure 5.10 : un modèle conceptuel pour un agent IDS 86
Figure 5.11 : Architecture du Détecteur d'AODVSTAT 88
Figure 5.12 : Architecture à haut niveau des composantes logiques RIDAN 89

Liste des tableaux :

Tableau 4.1 : A gauche topologie du réseau, à droite table de routage du nœud (1) dans le Protocole DSDV 59
Tableau 4.2 : Les classes des protocoles de routage pour les réseaux ad hoc 65
Tableau 4.3 : Les protocoles de routage pour les réseaux ad hoc 66
Tableau 5.1 : ATTAQUES CONTRE AODV 76

Sommaire :

Introduction générale 0

Chapitre 1 :

Introduction 1

1. **La sécurité informatique 1**
2. **Objectifs de la sécurité informatique 1**
3. **Présentation de l'insécurité informatique 2**

3.1. Les menaces 2
3.2. Vulnérabilité des systèmes informatiques 3
3.3. Les risques 3
3.4. Les attaques 4
3.4.1. Définition 4
3.4.2. Type d'attaques 4
3.4.3. Les différents types d'attaques 5

A. **Anatomie d'une attaque 6**
B. **Les attaques réseau 6**
- **B.1. Les techniques de scan 6**
- **B.2. IP Spoofing 7**
- **B.3. ARP Spoofing (ou ARP Redirect) 7**
- **B.4. DNS Spoofing 8**
- **B.5. Fragments attacks 8**
- **B.6. TCP Session Hijacking 9**

C. **Les attaques applicatives 9**
D. **Le Déni de service 10**
E. **Les attaques virales 10**

4. **Mécanismes de défenses 13**

4.1. Authentification 13
4.2. Cryptographie 13
4.3. Réseau privé virtuel (VPN) 15
4.4. Antivirus 15
4.5. Firewall (Pare-feu) 16
4.6. Systèmes de détection et de prévention d'intrusions 17
4.7. De l'IDS à l'IPS 18

5. **Mise en place d'une politique de sécurité 18**

Conclusion 19

Chapitre 2:

Introduction 20

1. **Définition de la sécurité 20**

2. Définition de la sécurité 21
3. Les mécanismes de sécurité 22
3.1. La cryptographie 22
a. La cryptographie symétrique 22
b. la cryptographie asymétrique ou à clés publiques 23
3.2. les fonctions de hachage 24
3.3. les chaines de hachage 24
3.4. La signature numérique 25
3.5. Certificats électroniques 27
3.6. La réputation 28
4. Attaques et vulnérabilités contre les protocoles de routage 29
4.1. Classification des attaques 29
4.2. Présentation de quelques attaques 29
5. Etat de l'art sur le routage sécurisé 31
5.1. Solutions utilisant la cryptographie 31
5.2. Solutions basées sur la réputation 32
5.3. Approches basées sur les IDS 33
5.4. Solutions intégrées aux cartes à puces 34
5.5. Analyse des solutions de routage sécurisé 34

Conclusion 35

Chapitre 3 :

Introduction 36

1. Définitions 36
1.1. Intrusion 36
1.2. Détection d'intrusion 36
1.2.1. Autre définition 37
2. Architecture classique d'un système de détection d'intrusion 37
3. Les caractéristiques souhaitées d'un IDS 39
4. Classification des systèmes de détection d'intrusions 40
4.1. La méthode d'analyse 41
4.1.1. La détection d'abus (par scénario) 42
4.1.2. La détection d'anomalie (comportementale) 42
4.2. Le comportement de la détection (la réponse) 44
4.2.1. Les réponses actives 44
4.2.2. Les réponses passives 45
4.3. L'emplacement des sources d'audits 45
4.3.1. NIDS (Network-Based IDS) 46
4.3.2. HIDS (Host-Based IDS) 46
4.3.3. IDS d'application 46
4.3.4. IDS hybrides 47
4.4. La fréquence d'utilisation (la synchronisation) 47
4.4.1. En temps différé (Périodique) 47

4.4.2. En temps réel (Continu) 47
4.5. L'architecture 47
4.5.1. Host Target Colocation (Cohabitation de la cible et l'hôte) 48
4.5.2. Host Target Separation (Séparation entre la cible et l'hôte) 48
4.6. La stratégie de contrôle 48
4.6.1. Centralisée 48
4.6.2. Partiellement distribuée 48
4.6.3. Entièrement distribuée 49
5. Emplacement des systèmes de détection d'intrusions 49
5.1. En amont 49
5.2. En aval 50
5.3. Dans la DMZ 51
5.4. Dans le LAN 51
6. Imperfection dans les implémentations actuelles des IDS 51
7. Méthodes de classification et d'IA pour la détection d'intrusions 52
7.1. réseaux bayésiens 52
7.2. Arbre de décision 53
7.3. Réseaux de neurones 54
8. Les systèmes de détection d'intrusions actuels 54
9. Imperfection des systèmes de détections d'intrusions 55

Conclusion 56

Chapitre4

Introduction 57
1. Classification des protocoles de routage Ad 57
1.1. Les protocoles Table-driven (pro-actifs) 57
1.2. Les protocoles on-demand (réactifs) 57
1.3. Les protocoles Hybrides 58
2. Quelques protocoles de routage pour les réseaux ad hoc 58
2.1. Le protocole DSDV 58
2.2. Le protocole GSR 59
2.3. Le protocole FSR 59
2.4. Le protocole AODV 60
2.5. Le protocole DSR 61
2.6. Le protocole CBRP 62
2.7. Le protocole ARA 63
2.8. Le protocole Ant-AODV 64
3. Tableaux récapitulatifs 64
Conclusion 66

Chapitre 5 :

Introduction 67

1. PROTOCOLE AODV 67
   Découverte des routes 67
   Maintenance de routes 69
2. Propriétés d'AODV 71
   2.1. Les mérites d'AODV 71
   2.2. Les inconvénients d'AODV 71
3. Vulnérabilités dans AODV 72
   3.1. Attaques élémentaires portant sur les demandes de route 72
   3.1.1. Suppression d'une demande de route 72
   3.1.2. Modification d'une demande de route 72
   3.1.3. Fabrication d'une demande de route 73
   3.1.4. Rushing d'une demande de route 73
   3.2. Attaques élémentaires portant sur les réponses de route 73
   3.2.1. Suppression d'une réponse de route 73
   3.2.2. Modification d'une réponse de route 73
   3.2.3. Fabrication d'une réponse de route 73
   3.3. Attaques élémentaires portant sur les erreurs de route 74
   3.3.1. Suppression d'une erreur de route 74
   3.3.2. Modification d'une erreur de route 74
   3.3.3. Fabrication d'une erreur de route 74
   3.4. Attaques composés 74
   3.4.1. Répétition régulière d'attaques élémentaires 74
   3.4.2. Création d'une boucle de routage 75
   3.4.3. Création d'un tunnel 75
4. Etude de l'attaque Blackhole 76
5. Attaques dans les Réseaux Ad hoc Sans fil 78
6. Solutions de l'attaque Black Hole 81
7. Les IDSs conçus dans le cadre d'AODV 83
   7.1. Specification-Based IDS for AODV 83
   7.2. IDS à base du protocole AODV (AODV protocol-base IDS) 84
   7.3. IDS distribué et Coopératif (Distributed and Cooperative IDS) 86
   7.4. Un outil de détection d'intrusion pour AODV basée sur des réseaux sans fil Ad hoc (An Intrusion Detection Tool for AODV-based Ad hoc Wireless Networks) 87
   7.5. Real-time Intrusion Detection for Ad hoc Networks (RIDAN) 88
   7.6. Un modèle IDS intégrant différentes techniques (An IDS Model Integrating Different Techniques) 89

Conclusion 90

Conclusion générale 91

Bibliographie 92

Introduction générale

Introduction générale :

Les réseaux sans fil et les réseaux mobiles sont devenus très populaires ces dernières années. Ceci est dû à leurs caractéristiques : Installation simple et facile, absence de câblage, les coûts de matériel ne sont pas prohibitifs, les utilisateurs se déplacent librement au sein de la zone de couverture du réseau. Donc ils peuvent être mis en place facilement et économiquement selon les besoins. Ils offrent en effet un large éventail d'applications, notamment dans les situations géographiques avec des contraintes terrestres telles que les champs de bataille, les applications militaires, et d'autres situations d'urgence et de catastrophe.

Cependant le grand problème de ces réseaux est la sécurité, les travaux de recherche indique que les réseaux sans fil sont plus vulnérables que les réseaux filaires en raison de leurs caractéristiques tels que le milieu ouvert, la topologie dynamique, l'absence d'administration centrale, la coopération distribuée, et la capacité restreinte (en termes de puissance et de calcul). L'utilisation de liaisons sans fil rend ces réseaux plus sujets à des menaces de sécurité physiques que les réseaux câblés, allant de l'écoute passive à l'interférence active. Sans aucune sécurité adéquate, les hôtes mobiles sont facilement capturés, compromis et détournés par des nœuds malveillants. L'adversaire peut écouter et / ou modifier les messages dans le canal de communication, injecter des messages erronés, supprimer des messages, et même passer par d'autres nœuds. Par conséquent, les mécanismes de sécurité dans de tels réseaux sont essentiels pour protéger les données émises par les utilisateurs et les systèmes de détection d'intrusions (IDS) constituent une bonne alternative pour mieux protéger le réseau informatique.

La détection d'intrusions consiste à scruter le trafic réseau, collecter tous les événements, les analyser et générer des alarmes en cas d'identification de tentatives malveillantes.

Dans le but d'assurer la connectivité du réseau, malgré l'absence totale d'infrastructure fixe et la mobilité des stations dans les réseaux ad hoc, chaque nœud est susceptible d'être mis à contribution pour participer au routage et pour retransmettre les paquets d'un nœud qui n'est pas en mesure d'atteindre sa destination, tout nœud joue ainsi le rôle de station et de routeur. Pour cela, le routage est considéré comme l'un des majeures problématiques des réseaux MANETs, donc plusieurs protocoles de routage ont été proposés. Ces derniers peuvent être classifiés suivant la manière de création et de maintenance de routes lors de l'acheminement selon plusieurs critères.

On peut définir deux grandes familles de protocoles : Les protocoles réactifs et les protocoles proactifs, et pour se rapprocher plus du mécanisme de ces protocoles on s'est intéressé dans ce mémoire au protocole AODV qui est un protocole de routage réactif.

Le présent mémoire est organisé comme suit :

Nous avons commencé par une introduction générale dans laquelle nous avons défini la problématique.

Le premier chapitre représente une introduction générale au domaine de la sécurité informatique d'une façon générale.

Dans le deuxième chapitre, nous avons donné une brève description à la sécurité dans les réseaux mobiles ad hoc ainsi que les solutions proposées

Le troisième chapitre constitue une brève description des systèmes de détection d'intrusion. Dans ce chapitre nous avons définir la détection et le processus de la détection d'intrusion.

Dans le chapitre quatre, nous avons présenté les différents protocoles de routage dans les réseaux mobiles ad hoc.

Dans le cinquième chapitre nous avons présenté le protocole de routage AODV, ainsi que les attaques liées à son encontre et enfin avons défini les différentes solutions et les IDSs proposés pour pallier au problème de sécurité dans le protocole de routage AODV

Enfin, une conclusion générale et des perspectives font la fin de ce mémoire.

CHAPITRE 1

Généralités sur La sécurité informatique

INTRODUCTION

La sécurité informatique est une notion de plus en plus présente dans le monde actuel. En effet, l'informatique joue un rôle de plus en plus important au sein de l'entreprise et la moindre défaillance du système informatique aurait de lourdes conséquences sur sa productivité. De plus, les systèmes d'informations de l'entreprise sont de plus en plus ouverts sur le monde ce qui ouvre une voie d'accès à l'entreprise à n'importe qui.

Dans ce qui suit, nous parlerons de la sécurité en générale puis des différentes menaces qui pèsent sur les systèmes informatiques et nous terminerons par quelques mécanismes de défense.

1. La sécurité informatique: [01]

La sécurité informatique est l'ensemble des techniques qui assurent que les ressources du système d'information (matérielles ou logicielles) d'une organisation sont utilisées uniquement dans le cadre où il est prévu qu'elles le soient.

2. Objectifs de la sécurité informatique : [02]

La sécurité des données couvre quatre objectifs principaux, et est représentée sous forme d'acronymes (C.I.D.P):

- La disponibilité est l'assurance que les personnes autorisées ont accès à l'information quand elles le demandent ou dans les temps requis pour son traitement.
- L'intégrité est la certitude de la présence non modifiée ou non altérée d'une information et de la complétude des processus de traitement. Pour les messages échangés, il concerne la protection contre l'altération accidentelle ou volontaire d'un message transmis.
- La confidentialité est l'assurance que l'information n'est accessible qu'aux personnes autorisées, qu'elle ne sera pas divulguée en dehors d'un environnement spécifié. Elle traite de la protection contre la consultation de données stockées ou échangées. Cela est réalisable par un mécanisme de chiffrement pour le transfert ou le stockage des données.
- La preuve consiste à garantir que l'émetteur d'une information soit bien identifié et qu'il a les droits et les accès logiques, que le récepteur identifié est bien autorisé à accéder à l'information.

D'autres principes de sécurité peuvent être établis, il s'agit de :

- ✓ La non-répudiation, considérée comme le cinquième principe, a été introduite dans la norme ISO 7498-2 comme un service de sécurité pouvant être rendu par un mécanisme comme la signature numérique, l'intégrité des données ou la notarisation. L'élément de la preuve de non-répudiation doit permettre l'identification de celui qu'il représente, il doit être positionné dans le temps (horodatage), il doit présenter l'état du contexte dans lequel il a été élaboré (certificats).

- ✓ L'authentification est le moyen qui permet d'établir la validité de la requête émise pour accéder à un système. L'authenticité est la combinaison d'une authentification et de l'intégrité.
- ✓ Les mécanismes de chiffrement procèdent du principe que l'émetteur et le récepteur conviennent d'un mot de passe connu d'eux seuls. L'émetteur utilise ce mot de passe comme clé de chiffrement pour le message à transmettre, seul le récepteur qui connaît ce mot de passe peut l'utiliser comme clé pour déchiffrer le message et y accéder.

Les objectifs de base peuvent être traités sous la forme de solutions de sécurité sous la forme de matériels, de logiciels, de procédures, de support opérationnel pour :

- ❖ l'intégrité des données et la confidentialité : gestion des accès physiques et logiques, sécu rité des réseaux,
- ❖ la disponibilité : redondance des systèmes et du stockage des données, sauvegarde et archivage des données.

3. Présentation de l'insécurité informatique :

3.1 Les menaces : [03]

La menace informatique représente le type d'actions susceptibles de nuire dans l'absolu à un système informatique. En termes de sécurité informatique les menaces peuvent être le résultat de diverses actions en provenance de plusieurs origines :

- **Origine opérationnel:**
  Ces menaces sont liées à un état du système à un moment donné. Elles peuvent être le résultat d'un bogue logiciel (Buffer Overflows, format string …etc.), d'une erreur de filtrage des entrées utilisateur (typiquement les XSS et SQL injection), d'un dysfonctionnement de la logique de traitement ou d'une erreur de configuration
- **Origine physique:**
  Elles peuvent être d'origine accidentelle, naturelle ou criminelle. On peut citer notamment les désastres naturels, les pannes ou casses matérielles, le feu ou les coupures électriques.
- **Origine humaine**
  Ces menaces sont associées directement aux erreurs humaines, que ce soit au niveau de la conception d'un système d'information ou au niveau de la manière dont on l'utilise. Ainsi elles peuvent être le résultat d'une erreur de conception ou de configuration comme d'un manque de sensibilisation des utilisateurs face au risque lié à l'usage d'un système informatique.

Les principales menaces effectives auxquelles un système d'information peut être confronté sont : **[04]**

- ✓ **Un utilisateur du système :** l'énorme majorité des problèmes liés à la sécurité d'un système d'information est l'utilisateur, généralement insouciant ;
- ✓ **Une personne malveillante :** une personne parvient à s'introduire sur le système, légitimement ou non, et à accéder ensuite à des données ou à des programmes auxquels elle n'est pas censée avoir accès en utilisant par exemple des failles connues et non corrigées dans les logiciels ;
- ✓ **Un programme malveillant :** un logiciel destiné à nuire ou à abuser des ressources du système est installé (par mégarde ou par malveillance) sur le système, ouvrant la porte à des intrusions ou modifiant les données ; des données personnelles peuvent être collectées à l'insu de l'utilisateur et être réutilisées à des fins malveillantes ou commerciales ;
- ✓ **Un sinistre (vol, incendie, dégât des eaux) :** une mauvaise manipulation ou une malveillance entraînant une perte de matériel et/ou de données.

3.2 Vulnérabilité des systèmes informatiques : [05]

La vulnérabilité d'un système, appelée parfois faille de sécurité ou brèche d'un système est un bug particulier dont l'exploitation permet d'effectuer des actions qui ne sont pas possibles dans le cadre d'une utilisation normale d'un logiciel. Ces failles sont dues aux faiblesses dans la conception ou à la mauvaise configuration du système d'exploitation, des protocoles réseaux ou des applications utilisateurs. Les failles les plus redoutables sont celles permettant l'exécution de code à distance, c.-à-d. permettant à un individu malveillant d'exécuter un programme malicieux sur l'ordinateur de sa victime via internet, même si celle-ci se trouve à l'autre bout de la planète.

Pour remédier à une faille de sécurité il faut corriger le code défectueux en appliquant un correctif ou en installant une nouvelle version du logiciel.

3.3 Les risques : [06]

Le risque est la probabilité qu'une menace donnée exploite des vulnérabilités d'un bien ou d'un groupe de biens et donc occasionne un dommage à l'organisme (exemple : effacement d'un fichier par un virus). Il est mesuré en termes de combinaison de probabilité d'un événement et de ses conséquences.

Un risque est donc caractérisé par deux facteurs : la probabilité qu'un incident se produise et l'importance des conséquences directes et des impacts indirects potentiels.

Le risque peut aussi dépendre du facteur temps : après un incident, la situation peut se dégrader progressivement si l'on ne prend pas les mesures correctives suffisamment tôt (exemple : erreur de logiciel affectant une base de données, programme espion collectant

des mots de passe, des clés cryptographiques ou des codes PIN). Un incident bénin au moment de sa survenue peut ainsi conduire à des situations catastrophiques.

3.4 Les attaques :

3.4.1. Définition : [07]

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

Une « **attaque** » est l'exploitation d'une faille d'un système informatique (système d'exploitation, logiciel ou bien même de l'utilisateur) à des fins non connues par l'exploitant du système et généralement préjudiciables.

Sur internet des attaques ont lieu en permanence, à raison de plusieurs attaques par minute sur chaque machine connectée. Ces attaques sont pour la plupart lancées automatiquement à partir de machines infectées (par des virus, chevaux de Troie, vers, etc.), à l'insu de leur propriétaire. Plus rarement il s'agit de l'action de pirates informatiques.

Afin de contrer ces attaques il est indispensable de connaître les principaux types d'attaques afin de mettre en œuvre des dispositions préventives.

Les motivations des attaques peuvent être de différentes sortes :

- obtenir un accès au système ;
- voler des informations, tels que des secrets industriels ou des propriétés intellectuelles ;
- glaner des informations personnelles sur un utilisateur ;
- récupérer des données bancaires ;
- s'informer sur l'organisation (entreprise de l'utilisateur, etc.) ;
- troubler le bon fonctionnement d'un service ;
- utiliser le système de l'utilisateur comme « rebond » pour une attaque ;
- utiliser les ressources du système de l'utilisateur, notamment lorsque le réseau sur lequel il est situé possède une bande passante élevée

3.4.2 Type d'attaques : [07]

Les systèmes informatiques mettent en œuvre différentes composantes, allant de l'électricité pour alimenter les machines au logiciel exécuté via le système d'exploitation et utilisant le réseau.

Les attaques peuvent intervenir à chaque maillon de cette chaîne, pour peu qu'il existe une vulnérabilité exploitable. La figure ci-dessous rappelle très sommairement les différents niveaux pour lesquels un risque en matière de sécurité existe :

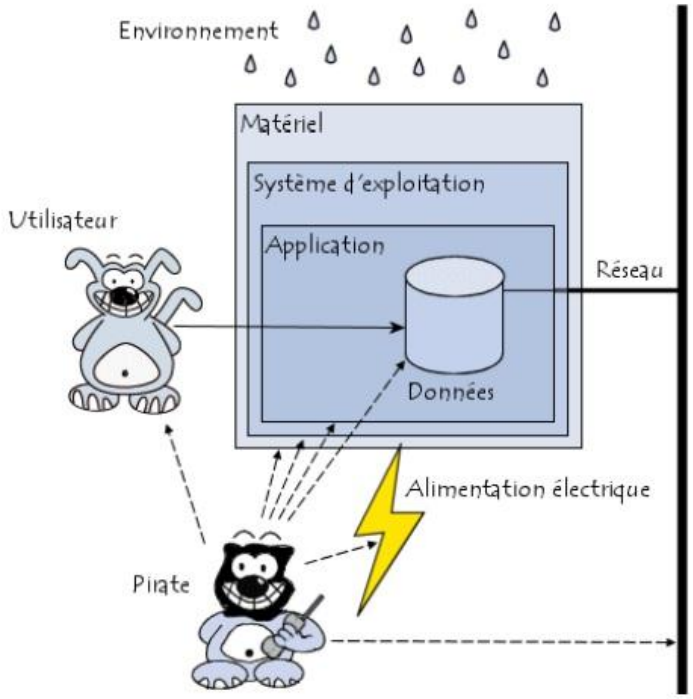


Figure 1.1 : les niveaux de vulnérabilité dans un système informatique

3.4.3 Les différents types d'attaques : [07]

L'informatique étant un domaine très vaste, le nombre de vulnérabilités présentes sur un système peut donc être important. Ainsi, les attaques visant ces failles peuvent être à la fois très variées et très dangereuses. C'est pourquoi nous allons dans un premier temps analyser ce que nous appellerons « l'anatomie d'une attaque », puis dans un second temps, nous caractériserons ces attaques et observerons leur déroulement.

A. Anatomie d'une attaque : [07]

Fréquemment appelés « **les 5 P** » dans la littérature, ces cinq verbes anglophones constituent le squelette de toute attaque informatique : **Probe**, **Penetrate**, **Persist**, **Propagate**, **Paralyze**.

Observons le détail de chacune de ces étapes :

- **Probe** : consiste en la collecte d'informations par le biais d'outils comme whois, Arin, DNS lookup. La collecte d'informations sur le système cible peut s'effectuer de plusieurs manières, par exemple un scan de ports grâce au programme Nmap pour déterminer la version des logiciels utilisés, ou encore un scan de vulnérabilités à l'aide du programme Nessus.

Pour les serveurs web, il existe un outil nommé Nikto qui permet de rechercher les failles connues ou les problèmes de sécurité.

Des outils comme firewalk, hping ou SNMP Walk permettent quant à eux de découvrir la nature d'un réseau ;

- **Penetrate** : utilisation des informations récoltées pour pénétrer un réseau. Des techniques comme le brute force ou les attaques par dictionnaires peuvent être utilisées pour outrepasser les protections par mot de passe. Une autre possibilité pour s'infiltrer dans un système est d'utiliser des failles applicatives que nous verrons ci-après ;
- **Persist** : création d'un compte avec des droits de super utilisateur pour pouvoir se réinfiltrer ultérieurement. Une autre technique consiste à installer une application de contrôle à distance capable de résister à un reboot (ex. : un cheval de Troie) ;
- **Propagate** : cette étape consiste à observer ce qui est accessible et disponible sur le réseau local;
- **Paralize** : cette étape peut consister en plusieurs actions. Le pirate peut utiliser le serveur pour mener une attaque sur une autre machine, détruire des données ou encore endommager le système d'exploitation dans le but de planter le serveur.

Après ces cinq étapes, le pirate peut éventuellement tenter d'effacer ses traces, bien que cela ne soit rarement utile. En effet, les administrateurs réseau sont souvent surchargés de logs à analyser. De plus, il est très difficile de supprimer entièrement des traces.

B. Les attaques réseau : [07]

Ce type d'attaque se base principalement sur des failles liées aux protocoles ou à leur implémentation. Les RFC(1) ne sont parfois pas assez spécifiques, et un choix particulier d'implémentation dans les différents services ou clients peut entraîner un problème de sécurité.

Observons quelques attaques bien connues.

B.1 Les techniques de scan :

Les scans de ports ne sont pas des attaques à proprement parler. Le but des scans est de déterminer quels sont les ports ouverts, et donc en déduire les services qui sont exécutés sur la machine cible (ex. : port 80/TCP pour un service HTTP). Par conséquent, la plupart des attaques sont précédées par un scan de ports lors de la phase Probe qui est comme nous l'avons vu, la première phase des 5P's dans le déroulement d'une attaque.

Il existe un nombre important de techniques de scan. Idéalement, la meilleure technique de scan est celle qui est la plus furtive afin de ne pas alerter les soupçons de la future victime. Voici une description des techniques de scan les plus répandues :

- **Le scan simple :** aussi appelé le scan connect (), il consiste à établir une connexion TCP complète sur une suite de ports. S'il arrive à se connecter, le port est ouvert ; sinon, il est fermé. Cette méthode de scan est très facilement détectable ;
- **Le scan furtif :** aussi appelé scan SYN, il s'agit d'une amélioration du scan simple. Ce scan essaie également de se connecter sur des ports donnés, mais il n'établit pas complètement la connexion : pas de commande ACK (acquittement) après avoir reçu

l'accord de se connecter. Grâce à ceci, la méthode est bien plus furtive que le scan normal ;

- **Les scans XMAS, NULL et FIN** : se basent sur des détails de la RFC du protocole TCP pour déterminer si un port est fermé ou non en fonction de la réaction à certaines requêtes. Ces scans sont moins fiables que le scan SYN, mais ils sont un peu plus furtifs. La différence entre ces trois types de scan se situe au niveau des flags TCP utilisés lors de la requête ;
- **le scan à l'aveugle :** s'effectue via une machine intermédiaire et avec du spoofing (voir plus bas). Le système attaqué pense que le scan est réalisé par la machine intermédiaire et non par le pirate ;
- **Le scan passif :** est la méthode la plus furtive. Consiste à analyser les champs d'en-tête des paquets (TTL, ToS, MSS…) et à les comparer avec une base de signatures qui pourra déterminer les applications qui ont envoyé ces paquets.

Remarque : l'utilitaire incontournable pour réaliser des scans de ports se nomme Nmap.

B.2 IP Spoofing :

But : usurper l'adresse IP d'une autre machine.

Finalité : se faire passer pour une autre machine en truquant les paquets IP. Cette technique peut être utile dans le cas d'authentifications basées sur une adresse IP (services tels que rlogin ou ssh par exemple).

Déroulement : il existe des utilitaires qui permettent de modifier les paquets IP ou de créer ses propres paquets (ex. : hping2). Grâce à ces utilitaires, il est possible de spécifier une adresse IP différente de celle que l'on possède, et ainsi se faire passer pour une autre « machine ».

Cependant, ceci pose un problème : en spécifiant une adresse IP différente de notre machine, nous ne recevrons pas les réponses de la machine distante, puisque celle-ci répondra à l'adresse spoofée. Il existe toutefois deux méthodes permettant de récupérer les réponses :

- ✓ **Source routing :** technique consistant à placer le chemin de routage directement dans le paquet IP. Cette technique ne fonctionne plus de nos jours, les routeurs rejetant cette option.
- ✓ **Reroutage :** cette technique consiste à envoyer des paquets RIP aux routeurs afin de modifier les tables de routage. Les paquets avec l'adresse spoofée seront ainsi envoyés aux routeurs contrôlés par le pirate et les réponses pourront être également reçues par celui-ci.

B.3 ARP Spoofing (ou ARP Redirect):

But : rediriger le trafic d'une machine vers une autre.

Finalité : grâce à cette redirection, une personne mal intentionnée peut se faire passer pour une autre. De plus, le pirate peut rerouter les paquets qu'il reçoit vers le véritable destinataire, ainsi

l'utilisateur usurpé ne se rendra compte de rien. La finalité est la même que l'IP spoofing, mais on travaille ici au niveau de la couche liaison de données.

Déroulement : pour effectuer cette usurpation, il faut corrompre le cache ARP de la victime. Ce qui signifie qu'il faut lui envoyer des trames ARP en lui indiquant que l'adresse IP d'une autre machine est la sienne. Les caches ARP étant régulièrement vidés, il faudra veiller à maintenir l'usurpation.

B.4 DNS Spoofing :

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

Finalité : rediriger, à leur insu, des internautes vers des sites pirates. Grâce à cette fausse redirection, l'utilisateur peut envoyer ses identifiants en toute confiance par exemple.

Déroulement : il existe deux techniques pour effectuer cette attaque :

- **DNS Cache Poisoning :** les serveurs DNS possèdent un cache permettant de garder pendant un certain temps la correspondance entre un nom de machine et son adresse IP. Le DNS Cache Poisoning consiste à corrompre ce cache avec de fausses informations. Ces fausses informations sont envoyées lors d'une réponse d'un serveur DNS contrôlé par le pirate à un autre serveur DNS, lors de la demande de l'adresse IP d'un domaine (ex. : www.ledomaine.com). Le cache du serveur ayant demandé les informations est alors corrompu ;
- **DNS ID Spoofing** : pour communiquer avec une machine, il faut son adresse IP. On peut toutefois avoir son nom, et grâce au protocole DNS, nous pouvons obtenir son adresse IP. Lors d'une requête pour obtenir l'adresse IP à partir d'un nom, un numéro d'identification est placé dans la trame afin que le client et le serveur puissent identifier la requête.
  L'attaque consiste ici à récupérer ce numéro d'identification (en sniffant le réseau) lors de la communication entre un client et un serveur DNS, puis envoyer des réponses falsifiées au client avant que le serveur DNS lui réponde.

Remarque : une attaque que nous allons voir ci-après, le Déni de Service, peut aider à ralentir le trafic du serveur DNS et ainsi permettre de répondre avant lui.

B.5 Fragments attacks :

But : le but de cette attaque est de passer outre les protections des équipements de filtrage IP.

Finalité : en passant outre les protections, un pirate peut par exemple s'infiltrer dans un réseau pour effectuer des attaques ou récupérer des informations confidentielles.

Déroulement : deux types d'attaque sur les fragments IP peuvent être distingués :

- **Fragments overlapping** : quand un message est émis sur un réseau, il est fragmenté en plusieurs paquets IP. Afin de pouvoir reconstruire le message, chaque paquet possède un offset. Le but de l'attaque est de réaliser une demande de connexion et de faire chevaucher des paquets en spécifiant des offsets incorrects. La plupart des filtres analysant les paquets indépendamment, ils ne détectent pas l'attaque. Cependant, lors de la défragmentation, la demande de connexion est bien valide et l'attaque a lieu ;
- **Tiny fragments :** le but de l'attaque est de fragmenter une demande de connexion sur deux paquets IP : le premier paquet de taille minimum (68 octets selon la RFC du protocole IP) ne contient que l'adresse et le port de destination. Le deuxième paquet contient la demande effective de connexion TCP. Le premier paquet est accepté par les filtres puisqu'il ne contient rien de suspect. Quand le deuxième paquet arrive, certains filtres ne le vérifient pas, pensant que si le premier paquet est inoffensif, le deuxième l'est aussi. Mais lors de la défragmentation sur le système d'exploitation, la connexion s'établit!

De nos jours, une grande majorité des firewalls sont capables de détecter et stopper ce type d'attaques.

B.6 TCP Session Hijacking :

But : le but de cette attaque est de rediriger un flux TCP afin de pouvoir outrepasser une protection par mot de passe.

Finalité : 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.

Déroulement : 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.

C. Les attaques applicatives : [07]

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.

C.1 Les problèmes de configuration :

Il est très rare que les administrateurs réseau configurent correctement un programme. En général, ils se contentent d'utiliser les configurations par défaut. Celles-ci sont souvent non sécurisées afin de faciliter l'exploitation du logiciel (ex. : login/mdp par défaut d'un serveur de base de données).

De plus, des erreurs peuvent apparaître lors de la configuration d'un logiciel. Une mauvaise configuration d'un serveur peut entraîner l'accès à des fichiers importants, ou mettant en jeu l'intégrité du système d'exploitation. C'est pourquoi il est important de bien lire les documentations fournies par les développeurs afin de ne pas créer de failles.

C.2 Les bogues :

Liés à un problème dans le code source, ils peuvent amener à l'exploitation de failles. Il n'est pas rare de voir l'exploitation d'une machine bloquée suite à une simple erreur de programmation. On ne peut toutefois rien faire contre ce type de problèmes, si ce n'est attendre un correctif de la part du développeur.

C.3 Les buffers overflows :

Les buffers overflows, ou dépassement de la pile, sont une catégorie de bogue particulière. Issus d'une erreur de programmation, ils permettent l'exploitation d'un shellcode(3) 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 à la taille de l'entrée. On se retrouve donc en situation de débordement, et l'exploitant peut ainsi accéder à la mémoire.

C.4 Les scripts :

Principalement web (ex. : Perl, PHP, ASP), ils s'exécutent sur un serveur et renvoient un résultat au client. Cependant, lorsqu'ils sont dynamiques (i.e. qu'ils utilisent des entrées saisies par un utilisateur), des failles peuvent apparaître si les entrées ne sont pas correctement contrôlées.

L'exemple classique est l'exploitation de fichier à distance, telle que l'affichage du fichier mot de passe du système en remontant l'arborescence depuis le répertoire web.

C.5 Les injections SQL :

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.

C.6 Man in the middle:

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.

D. Le Déni de service : [07]

Évoqué précédemment, 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.

L'utilisation d'un buffer overflow peut permettre de planter l'application à distance.

Grâce à quelques instructions malicieuses et suite à une erreur de programmation, une personne mal intentionnée peut rendre indisponible un service (serveur web, serveur de messagerie…) voire un système complet.

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 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.

Exemple : faire une requête charge (port 19/service de génération de caractères) à une machine en spoofant l'adresse et le port source, pour rediriger vers echo (port 7/service qui répète la chaîne de caractères reçue) d'une autre machine ;

- **Packet Fragment :** utilise une mauvaise gestion de la défragmentation au niveau ICMP.

Exemple : ping of death. La quantité de données est supérieure à la taille maximum d'un paquet IP.

Remarque : pour rappel, nous avons vu que les techniques d'attaque se basant sur la fragmentation des paquets peuvent aussi être utilisées pour outrepasser un filtre IP ;

- **Smurfling :** le pirate fait des requêtes ICMP 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 ;
- **Déni de service distribué** : le but est ici de reproduire une attaque normale à grande échelle. Pour ce faire, le pirate va tenter de se rendre maître d'un nombre important de machines. Grâce à des failles (buffer overflows, failles RPC(4)…) il va pouvoir prendre le contrôle de machines à distance et ainsi pouvoir les commander à sa guise.

Une fois ceci effectué, il ne reste plus qu'à donner l'ordre d'attaquer à toutes les machines en même temps, de manière à ce que l'attaque soit reproduite à des milliers d'exemplaires. Ainsi,

une simple attaque comme un SYN Flooding pourra rendre une machine ou un réseau totalement inaccessible.

E. Les attaques virales :

Les attaques virales permettent de consommer et paralyser les ressources du système.

Voici quelques exemples d'attaque virale :

a. Virus : [08]

Un virus est un programme informatique (souvent court) introduit dans un système à l'insu de l'utilisateur. Le virus a pour mission première de se propager en infectant les cibles désignées par son concepteur. Après cette période d'incubation, il se manifeste de façon plus ou moins agressive. Les actions des virus sont très diverses. Cela peut aller de l'affichage d'un message humoristique accompagné d'une musique au formatage complet du disque dur.

Les virus sont une menace réelle. De petite taille et souvent furtifs, ils se propagent facilement via la messagerie électronique et tout ordinateur connecté à Internet devient une cible potentielle.

La prévention reste la meilleure arme contre les virus. Ne téléchargez rien à partir de sites douteux et n'ouvrez en aucun cas les pièces jointes aux emails suspects ou avec une accroche grossière du genre : " Vous avez gagné un million de francs ! ".

Un bon Antivirus mis à jour régulièrement reste la meilleure solution pour éradiquer les virus.

Il est également conseillé de mettre fréquemment son système d'exploitation à jour en téléchargeant les updates.

b. Les vers (Worms) : [09]

Un ver est un programme indépendant, qui se copie d'ordinateur en ordinateur. La différence entre un ver et un virus est que le ver ne peut pas se greffer à un autre programme et donc l'infecter. Il va simplement se copier via un réseau ou Internet, d'ordinateur en ordinateur. Ce type de réplication peut donc non seulement affecter un ordinateur, mais aussi dégrader les performances du réseau dans une entreprise. Comme un virus, un ver peut contenir une action nuisible du type destruction de données ou envoi d'informations confidentielles.

c. Le cheval de Troie : [10]

On appelle " Cheval de Troie" (en anglais *trojan horse*) un programme informatique effectuant des opérations malicieuses à l'insu de l'utilisateur. Le nom "Cheval de Troie" provient d'une légende narrée dans l'*Iliade* à propos du siège de la ville de Troie par les Grecs.

Un cheval de Troie est donc un programme caché dans un autre qui exécute des commandes sournoises, et qui généralement donne un accès à la machine sur laquelle il est exécuté en

ouvrant une porte dérobée (en anglais *backdoor*), par extension il est parfois nommé troyen par analogie avec les habitants de la ville de Troie.

Les opérations suivantes peuvent être effectuées par l'intermédiaire d'un cheval de Troie :

- ✓ récupération des mots de passe grâce à un keylogger.
- ✓ administration illégale à distance d'un ordinateur.
- ✓ relais utilisé par les pirates pour effectuer des attaques.
- ✓ serveur de spam (envoi en masse des e-mails).

4. Mécanismes de défenses :

Evidemment, nous ne pouvons pas compter sur un seul moyen de sécurité pour protéger l'information d'une entreprise. De même, nous ne pouvons pas compter sur un seul mécanisme pour qu'il fournisse toutes les sécurités nécessaires aux ordinateurs. Dans ce qui suit, nous verrons les mécanismes de défense les plus utilisés :

4.1 Authentification : [11]

La première étape afin de protéger les ressources d'un réseau est de pouvoir vérifier l'identité des utilisateurs. Cette vérification s'appelle authentification. L'authentification est la procédure mise en œuvre notamment pour vérifier l'identité d'une entité et s'assurer que cette dernière fournie, correspond à l'identité de cette entité préalablement enregistrée. On authentifie un utilisateur en lui demandant de fournir quelque chose que seule cette personne a, par exemple un jeton, une information qu'elle seule connait (un mot de passe). Plus l'utilisateur doit fournir des renseignements de ce type, plus faibles sont les risques qu'une personne parvienne à se faire passer pour utilisateur légitime.

Parmi les mécanismes d'authentification, on cite :

- ➢ Authentification par un code d'identification et un mot de passe.
- ➢ Sommation/réponse.
- ➢ Carte à mémoire.
- ➢ Biométrie.

4.2 Cryptographie : [12]

La cryptologie est à la fois une science et une technologie. Science, dont les principes les plus récents sont encore l'occasion de nouvelles découvertes. Technologie, utile et nécessaire dans l'industrie de la sécurité et pour tous ceux qui veulent protéger leur information. La cryptologie couvre couramment quatre grandes fonctions de sécurité :

A. L'authentification :

Il s'agit de garantir l'origine d'une information. En général, on utilise la signature numérique avec un couple de clés dont celle permettant de créer les signatures est gardée secrète, et dont

l'autre permettant de vérifier la signature est rendue publique. Le courrier électronique, un bon de commande transmis en ligne, un acte administratif peuvent être signés pour prouver leur origine et engager le signataire, à l'identique d'un paraphe sur le papier.

La signature numérique n'a réellement pris son sens qu'avec la découverte des systèmes de cryptologie dit «asymétriques». On peut diffuser largement le moyen de vérifier une signature sans risque de donner le moyen d'en contrefaire. Cependant les principes mathématiques employés sont récents, et si l'utilisation de la signature numérique est maintenant techniquement possible, l'édifice juridique et les usages courants ne sont pas encore adaptés à son utilisation. Divers travaux internationaux au sein de la Communauté européenne ou de l'OCDE tentent d'amorcer une évolution favorable à la reconnaissance juridique universelle des signatures numériques.

B. L'identification :

Il s'agit de garantir l'identité et la qualité d'une personne qui souhaite accéder à des informations ou à des ressources matérielles. En général, on utilise le contrôle d'accès par mot de passe. Pour consulter son courrier électronique, pour se connecter à un ordinateur distant, ou pour entrer dans un lieu protégé, on peut ainsi s'assurer de l'identité du demandeur. Ce problème est souvent négligé. En particulier, on voit encore trop de mots de passe circuler en clair sur les réseaux. Lors d'une ouverture de session ftp, ou telnet, vous êtes-vous déjà demandé comment est protégé le mot de passe que vous entrez sur l'ordinateur en local ? Dans la majorité des cas, il est simplement envoyé au serveur, risquant sur son trajet d'être attrapé par un «renifleur» et de finir dans la base de mots de passe d'un pirate qui se fera un plaisir de l'utiliser pour toutes sortes d'accès illégaux. Il conviendrait d'améliorer rapidement les méthodes de contrôle d'accès, mais, paradoxalement, c'est un des domaines de la sécurité où l'évolution est des plus lentes. Pourtant il existe des solutions, qu'il serait bon de valider au plus vite. Des techniques de cryptologie relativement simples, comme le défi réponse, permettent de ne pas diffuser l'information clé réutilisable, que cela soit un mot de passe ou une clé plus complexe sur un support physique.

C. La confidentialité

Il s'agit de garantir le secret de l'information transmise ou archivée. En général, on utilise le chiffrement au moyen d'une clé symétrique. Tout, du courrier électronique aux commandes d'administration d'un ordinateur à distance, peut être ainsi protégé sous une forme chiffrée. Trop souvent, la cryptologie est limitée dans les esprits à cette fonction de protection de la confidentialité. Sans doute des raisons historiques ne sont pas étrangères à cette confusion. En effet, pendant des siècles, c'est à peu près le seul usage qui en était fait. Déjà au Moyen Age, les grands stratèges et l'église utilisaient des formes élémentaires de chiffrement, dans le dessein de cacher le contenu de messages qui traversaient les lignes ennemies ou transitaient sur des terres hostiles.

D. L'intégrité

Il s'agit de garantir l'intégrité, c'est-à-dire l'absence de modification d'un message ou d'un document. On peut utiliser la signature numérique sous sa forme symétrique ou asymétrique, ou encore le chiffrement. Il est particulièrement important que, dans toute négociation et accord contractuel, on puisse vérifier qu'aucune modification du document électronique n'a été faite.

La cryptologie s'intéresse aussi à d'autres problèmes dont l'importance va croissant, comme la non-répudiation – garantissant que l'auteur d'un message ou d'un document ne peut pas nier l'avoir écrit et, le cas échéant, transmis –,l'anonymat (ou la non-traçabilité). La technologie peut rendre nombre de services, mais il lui est aussi possible de protéger l'individu de ses propres abus.

4.3 Réseau privé virtuel (VPN) : [13]

Les réseaux privés virtuels (VPN : Virtual Private Network) permettent à l'utilisateur de créer un chemin virtuel sécurisé entre une source et une destination. Avec le développement d'Internet, il est intéressant de permettre ce processus de transfert de données sécurisé et fiable. Grâce à un principe de tunnel (tunnelling) dont chaque extrémité est identifiée, les données transitent après avoir été chiffrées.

Un des grands intérêts des VPN est de réaliser des réseaux privés à moindre coût. En chiffrant les données, tout se passe exactement comme si la connexion se faisait en dehors d'Internet. Il faut par contre tenir compte de la toile, dans le sens où aucune qualité de service n'est garantie.

Le principe du VPN est basé sur la technique du tunnelling. 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. Le tunneling est l'ensemble des processus d'encapsulation, de transmission et de désencapsulation.

4.4 Antivirus : [14]

Un antivirus est un programme capable de détecter... les virus, les vers, les troyens et parfois les spywares qui peuvent infecter un ordinateur.

L'antivirus devrait être résident sur l'ordinateur, mais il existe aussi des tests d'infection virale disponibles sur le web.

L'antivirus résident :

Un antivirus " résident " est installé sur l'ordinateur comme n'importe quel autre programme classique. Il doit démarrer en même temps que l'ordinateur et rester actif durant tout le temps que dure la session de travail.

Le logiciel antivirus devrait obligatoirement figurer sur tout ordinateur, même non connecté à l'Internet : les disquettes échangées d'ordinateur à ordinateur sont également des vecteurs importants de virus.

Un bon antivirus doit intervenir au moment même de l'entrée ou de la tentative d'entrée d'une peste quelconque.

Lorsque le virus a été découvert, l'antivirus peut :

- nettoyer les fichiers infectés en éradiquant les virus
- supprimer les fichiers infectés (attention aux problèmes que cela peut poser)
- écarter le virus dans une zone du disque dur où il ne peut nuire : on parle alors d'une mise en quarantaine
- signaler son impuissance

4.5 Firewall (Pare-feu) : [15]

Un **pare-feu** est un logiciel ou un matériel qui se charge d'établir une barrière entre vous et le monde extérieur pour faire barrage aux pirates.

Rappelons qu'un ordinateur utilise des ports pour communiquer : par exemple, le port 80 est utilisé pour afficher des pages web. Il y a plus de 65000 ports (65536 exactement), soit autant de portes d'entrée dans votre ordinateur qu'un **firewall** se doit de protéger !

Un **pare-feu** peut vous permettre de "fermer" les ports et de cette manière, vous rendre invulnérable (ou presque). Il peut aussi restreindre le trafic sortant et applique des restrictions au trafic entrant.

Protéger son ordinateur est vital. En effet, les pirates cherchent non seulement à dérober vos données, mais aussi à voler vos mots de passe, usurper votre identité et causer des dommages à d'autres ordinateurs ou sites web à partir de votre PC.

Le pirate peut également installer des logiciels pirates sur votre PC et en faire la distribution à ses "confrères" qui viendront se servir chez vous, sur votre ordinateur.

Le pare-feu est un système aux fonctions de filtrage évoluées. Chaque paquet reçu est examiné, une décision de rejet ou d'acceptation est prise en fonction de nombreux critères :

- ✓ Adresse source.
- ✓ Adresse destination.
- ✓ Port source.
- ✓ Port destination.
- ✓ Le protocole transporté (ICMP, UPD…).
- ✓ La valeur de certains flags (ACK, SYN…).

Le firewall, placé à l'entrée du réseau, constitue ainsi un unique point d'accès par où chacun est obligé de passer…

Les types de **pare-feu** :

- ❖ Le pare-feu logiciel et personnel : il est simple d'utilisation. C'est un logiciel qui contrôle les données entrantes et sortantes. Sachez que Windows XP dispose d'un pare-feu. Un firewall logiciel coûte relativement peu cher.
- ❖ Le routeur : il masque votre adresse IP et vos ports. C'est un périphérique matériel accompagné d'un logiciel qu'il faut mettre souvent à jour. Il est déjà plus cher que le pare-feu personnel. Ce n'est pas un vrai pare-feu dans ce sens que ce n'est pas sa fonction première.
- ❖ Le pare-feu matériel : Il est destiné aux entreprises.

4.6. Systèmes de détection et de prévention d'intrusions : [16] [17]

Un système de détection d'intrusions (IDS, de l'anglais Intrusion Detection System) est un périphérique ou processus actif qui analyse l'activité du système et du réseau pour détecter toute entrée non autorisée et / ou toute activité malveillante. La manière dont un IDS détecte des anomalies peut beaucoup varier ; cependant, l'objectif principal de tout IDS est de prendre sur le fait les auteurs avant qu'ils ne puissent vraiment endommager vos ressources.

Les IDS protègent un système contre les attaques, les mauvaises utilisations et les compromis. Ils peuvent également surveiller l'activité du réseau, analyser les configurations du système et du réseau contre toute vulnérabilité, analyser l'intégrité de données et bien plus. Selon les méthodes de détection que vous choisissez de déployer, il existe plusieurs avantages directs et secondaires au fait d'utiliser un IDS.

Le système de prévention d'intrusions (IPS, de l'anglais Intrusion Prevention System) est un outil de défense proactif contre les attaques actives visant les ordinateurs et les réseaux. Ce dispositif réseau surveille les activités d'un réseau ou d'un système pour cerner des comportements malveillants ou indésirables et peut les bloquer.

4.7 De l'IDS à l'IPS : [17]

Auparavant, on détectait au moyen d'un système de détection d'intrusions (IDS) les tentatives d'intrusion seulement une fois qu'elles s'étaient produites. Il s'agissait d'un système réactif qui permettait aux attaques de pénétrer le réseau sans que l'analyste ne puisse intervenir au moment opportun.

Afin d'aborder la détection d'intrusions de façon proactive, les fournisseurs ont créé le système de prévention d'intrusions (IPS).

En général, l'IPS est inséré directement dans le circuit entre le poste de travail et Internet, là où il est en mesure d'inspecter le trafic qui transite et de décider s'il y a lieu de le laisser passer ou de le bloquer. De cette façon, l'IPS peut identifier et arrêter une menace avant qu'elle ne puisse s'introduire dans le réseau.

L'IPS peut être vulnérable aux attaques s'il n'est pas correctement configuré et maintenu. L'IPS est un système informatique doté d'un système d'exploitation et d'applications propres à la fonction d'identification. On recommande aux ministères de veiller à ce que les mises à jour et les rustines pour l'IPS soient appliquées régulièrement. Un IPS compromis peut laisser passer du trafic malveillant dans le réseau.

5. Mise en place d'une politique de sécurité : [18]

La mise en œuvre d'une politique de sécurité globale est assez difficile, essentiellement par la diversité des aspects à considérer. Une politique de sécurité peut se définir par un certain nombre de caractéristiques : les niveaux où elle intervient, les objectifs de cette politique et enfin les outils utilisés pour assurer cette sécurité.

Chaque aspect différent doit être pris en compte, de façon à atteindre les objectifs de sécurité désirés, en utilisant de façon coordonnée les différents outils à disposition.

Une politique de sécurité s'élabore à plusieurs niveaux.

- ✓ Sécuriser l'accès aux données de façon logicielle (authentification, contrôle d'intégrité).
- ✓ Sécuriser l'accès physique aux données : serveurs placés dans des salles blindées (qui empêchent les ondes électromagnétiques d'être captées) avec badge d'accès…
- ✓ Un aspect très important pour assurer la sécurité des données d'une entreprise est de sensibiliser les utilisateurs aux notions de sécurité, de façon à limiter les comportements à risque : si tout le monde peut accéder aux salles de serveurs, peut importe qu'elles soient sécurisées !
- ✓ De même, si les utilisateurs laissent leur mot de passe écrit à côté de leur PC, son utilité est limitée…
- ✓ Enfin, il est essentiel pour un responsable de sécurité de s'informer continuellement, des nouvelles attaques existantes, des outils disponibles…de façon à pouvoir maintenir à jour son système de sécurité et à combler les brèches de sécurité qui pourraient exister.

Conclusion :

Le présent chapitre a été entièrement dédié à l'étude des différents types de menaces, et des techniques utilisées par les pirates pour violer la sécurité des réseaux ainsi que les différentes techniques et mécanismes de défense pour bien parer aux attaques des pirates.

Nous nous focalisons dans le chapitre suivant sur la sécurité des réseaux mobiles ad-hoc

CHAPITRE 2

La sécurité des réseaux mobiles ad-hoc

INTRODUCTION

Les réseaux sont de plus en plus populaires, ils vont intégrer dans un futur proche toutes les situations de notre vie quotidienne. Une nécessité accrue s'est faite sentir pour rendre ces réseaux fiables et hautement sécurisés afin de protéger la vie privée des utilisateurs et offrir une bonne qualité de service pour les applications. Une tâche qui s'avère difficile et compliquée. Etant donnée le concept et la nature des réseaux ad hoc les rendent facilement vulnérables à différents types d'attaques. Ce qui rend la tâche encore plus difficile est que les nœuds du réseau se chargent eux-mêmes de la fonction de routage des données. Favorisé par la nature vulnérable des communications sans fil, n'importe qui peut se connecter sur le réseau et écouter les messages de contrôle échangés. Il pourra ensuite les supprimer, les modifier, ou mener d'autres attaques plus complexes, ce qui met en danger tout le réseau. Les protocoles de routage proposés dans le cadre du travail du groupe MANET offre un acheminement optimal des données mais n'offre aucun système de sécurité.

De plus l'emploi des stratégies de sécurités robustes utilisées avec succès dans les réseaux filaires et les réseaux sans fil avec infrastructure se trouve contraint par l'absence d'infrastructure centralisée qui pourrait gérer le service de sécurité dans le réseau. Dans un réseau ad hoc, les nœuds doivent exécuter eux-mêmes les mécanismes de sécurité pour se protéger contre les attaques. Mais le problème qui se pose est que les nœuds sont caractérisés par de modestes capacités de calcul, de stockage, et d'énergie. Dans ce cas-là, l'utilisation des systèmes de sécurité robustes et efficaces comme le cryptage par clé ou l'authentification sophistiquée qui consomment beaucoup de ressources ne donne pas toujours de bons résultats en pratique et peut affecter considérablement les performances du réseau.

L'aspect sécuritaire des protocoles de routage des réseaux ad hoc est difficile à mettre en œuvre en pratique. Il constitue l'un des principaux obstacles à un large déploiement de ces réseaux. Les travaux de recherches menés dans ce domaine tentent d'établir un compromis entre l'efficacité et la robustesse de la solution de sécurité proposée et son coût global.

Dans ce chapitre, nous allons mettre le point sur le problème de sécurité des protocoles de routage. Dans la première partie de ce chapitre nous introduisons les concepts et les terminologies fondamentales de la sécurité. Nous donnons sa définition, ces principaux objectifs, ensuite nous décrivons les mécanismes les plus utilisés pour la sécurité. La deuxième partie sera consacrée à l'étude des exigences de la sécurité, les différentes vulnérabilités liées aux protocoles de routage ad hoc, ainsi que les types d'attaques qui peuvent les menacées. Enfin, nous présenterons quelques approches utilisées pour la sécurité du routage ad hoc.

1. Définition de la sécurité

La sécurité informatique **[1]** est un ensemble de techniques assurant que les ressources (matérielles ou logicielles) d'un système d'information d'une organisation donnée sont utilisées uniquement dans le cadre où il est prévu qu'elles le soient.

2. Objectifs de la sécurité

Les réseaux Ad Hoc doivent satisfaire les services de sécurité suivants :

♦ Confidentialité

La confidentialité empêche les données d'être consultées par des entités non autorisées. Des contrôles d'accès strict doivent être mis en place pour garantir la confidentialité des données dans les réseaux ad hoc. Étant donné que les communications sans fil transitent via les airs, elles sont donc potentiellement accessibles à tout possesseur du récepteur adéquat.

♦ ***Contrôle d'accès*** : empêcher les nœuds étrangers d'accéder au réseau. Le contrôle d'accès donne aux nœuds légitimes un moyen de détecter les messages provenant de sources externes au réseau.

♦ Intégrité

C'est un service qui garantit que les données n'ont pas été altérées pendant la transmission. Donc le récepteur d'un message s'assure que le message reçu est le même que le message envoyé. L'intégrité des données est une exigence importante pour les réseaux ad hoc. Elle peut être remise en cause par de nombreux évènements. Parmi ceux-ci, les attaques visant à modifier le contenu des messages et la faible fiabilité des liaisons sans fil. [2]

♦ L'authentification de l'origine des données

Dans un réseau, un adversaire peut facilement injecter des paquets additionnels, ainsi le récepteur doit s'assurer que les données reçues proviennent effectivement de la source supposée.

♦ Disponibilité

Le principe de la disponibilité [3] permet de s'assurer que les services réseau désirés sont toujours disponibles même à la présence des attaques.

♦ La non répudiation de l'origine

C'est un mécanisme destiné à prévenir que la source ou la destination désavoue ses actions ou nie qu'un échange a eu lieu.

♦ La fraîcheur de données ou le non rejeu

Même si l'authentification, l'intégrité et la confidentialité de données sont assurées, la fraîcheur de chaque message doit être également assurée. La fraîcheur des données permet de garantir que les données sont récentes, et qu'aucun vieux message n'a été rejoué.

3. Les mécanismes de sécurité

3.1. La cryptographie

La cryptographie est la science d'écriture et de lecture de messages codés. En effet, elle joue un rôle essentiel dans toutes les communications sécurisée en chiffrant un message dit « texte clair » en un deuxième dit « texte crypté » à l'aide d'une clé en utilisant des moyens, matériels ou logiciels conçus à cet effet. Les informations originales sont restituées à partir de celles codées. Cette opération inverse est nommée décryptage **[4]**. En d'autre terme, la cryptographie est un traitement fait sur une donnée qui sera transmise à un destinataire à travers un canal peu sur en présence d'adversaire. Le défi est que cette donnée atteigne sa destination sans qu'elle soit modifiée et espionnée.

Il existe deux grands types de cryptographie :

a. La cryptographie symétrique

Dans la technique de cryptographie symétrique, chaque entité possède une clé de chiffrement/déchiffrement. Dans le cadre d'échanges sur un réseau, une entité émettrice chiffre les données avec une clé et l'entité destinataire déchiffre les données avec la même clé.

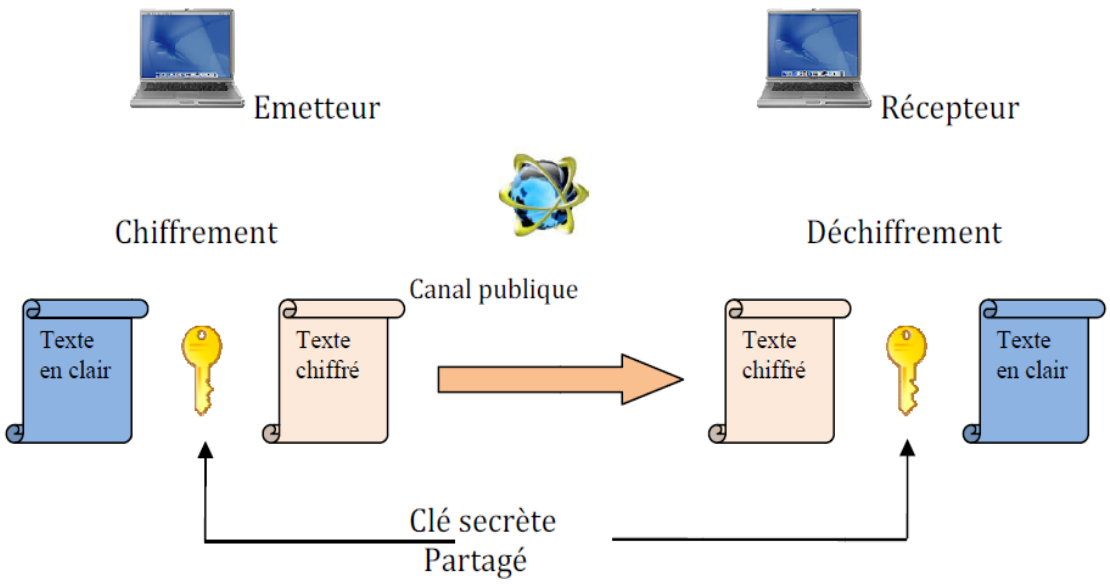


Figure 2.1 : *Cryptographie symétrique. [5]*

b. la cryptographie asymétrique ou à clés publiques

Dans la cryptographie asymétrique, les clés de chiffrement et de déchiffrement sont déférentes. Une des clés appelée clé secrète, est mémoriser et utilisée par une entité.
L'autre clé appelée clé publique, est distribuée à toutes les autres entités. La clé publique est utilisée en général lors du chiffrement et la clé privée pour le déchiffrement. Il est mathématiquement impossible de déduire la clé privée de la clé publique. L'inconvénient est que ces algorithmes utilisent des fonctions mathématiques complexes qui peuvent être réalisés sur des circuits intégrés. Le débit de ce type de chiffrement sera donc très faible.
Cette technique garantie soit la **confidentialité** ou bien **l'authentification** des messages.

- Pour authentifier l'origine d'un message dans une communication sur un réseau, l'émetteur chiffre le message en clair avec sa clé privée et attache le message résultant au message initial. Le récepteur déchiffre la partie chiffrée avec la clé publique de l'émetteur et retrouve donc le message initial. Il lui suffira de comparer le message ainsi obtenu avec la partie en claire pour authentifier l'expéditeur. Ces cryptages ce déroule de la manière suivante :

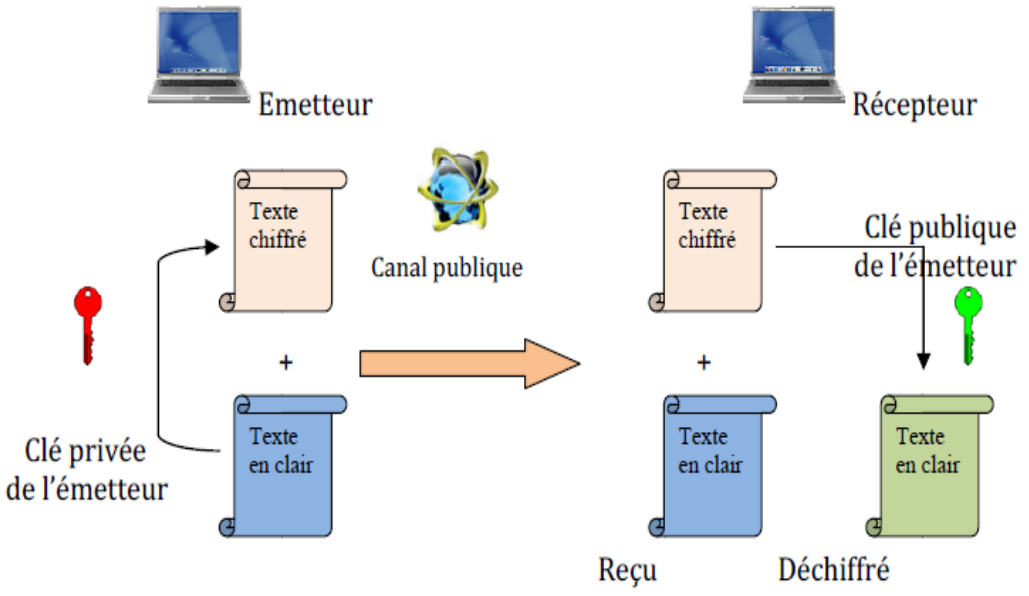


Figure 2.2 : ***Cryptographie asymétrique -- authentification***

D'autre part, pour garantir la confidentialité d'un message [6], il est nécessaire de chiffrer le message émis avec la clé publique du destinataire. La clé privée complémentaire n'est connue que du destinataire

du message. Ce dernier sera donc le seul pouvoir déchiffrer le message. La propriété de confidentialité est ainsi obtenue.

La figure suivante illustre le déroulement de cryptage :

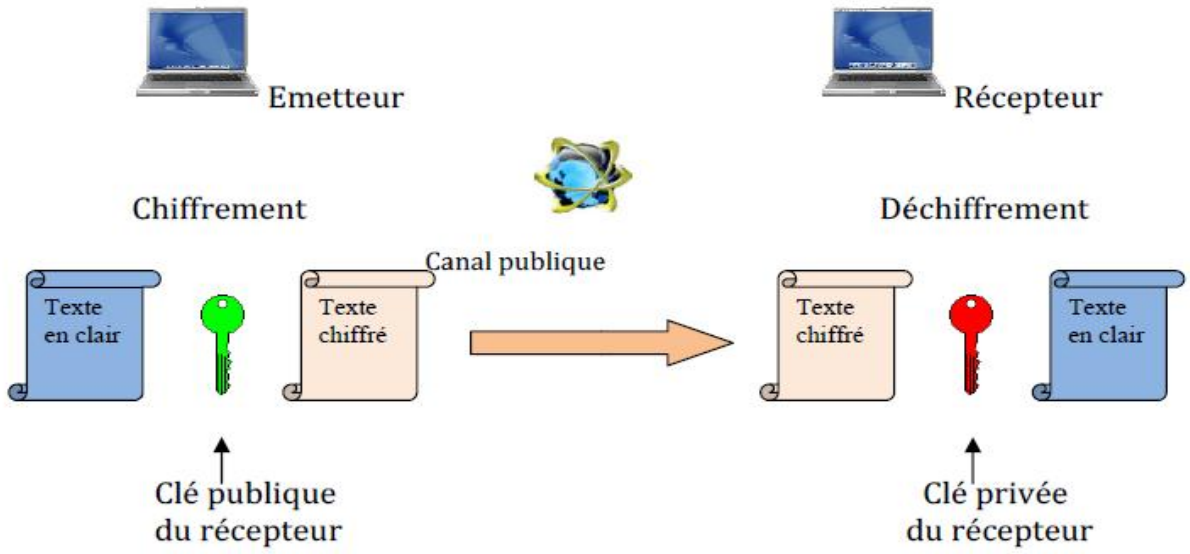


Figure 2.3 ***Cryptographie asymétrique – authentification***

3.2. les fonction de hachage [7]

Une fonction de hachage est une fonction permettant d'obtenir un condensé, applelé aussi empreinte , de longueur fixe à partir d'un texte de longueur arbitraire finie. La foction de hachage doit étre telle qu'elle associe un et un seul condensé à un texte en clair. Cela signifie que la moidre modification du texte entraine la modification de son condonsé. D'aure part **[8],** il doit s'agir d'une fonction facilement calculable et à sens unique afin qu'il soit impossible de retrouver le message original à partie du condénsé. En expédiant un message accompagné de son haché, il est possible de garantir l'intégrité d'un message. C'est-à-dire que le destinataire peut vérifier que le message n'a pas été altéré durant la communication.

3.3. les chaines de hachage

Les chaines de hachage sont basées sur les fonctions de hachage à sens unique**.**Une chaine de hachage de longueur N est construitr en appliquant une fonction de hachage N fois sur une valeur aléatoire appelé X_N. La valeur X_N est appelée valeur racine de la chaine de hachage. On définit une chaine de hachage en utilisant de hachage h par :

$h_i(Y) = h(h_{i-1}(Y))$

$h_0(Y) = X_N$

Où $h_i(Y)$ est le résultat de l'utilisation répétée i fois de la fonction de hachage à lavaleur initiale Y. La valeur finale de hachage de la chaîne de hachage $X_0 = h_N(X_N)$est obtenue en appliquant la fonction de hachage N fois. Le récepteur applique une seule fois la fonction de hachage pour vérifier la valeur dehachage reçue. Puisque la fonction de hachage est à sens unique, seulement l'utilisateur qui a créé la chaîne de hachage peut générer la valeur de hachage qui précède la valeur envoyée. [9]

3.4. La signature numérique

La signature numérique est définie comme des « données ajoutées à un message », ou transformation cryptographique d'un message, permettant à un destinataire de :

1. Authentifier l'auteur d'un document électronique.

2. Garantir son intégrité.

3. Protéger contre la contrefaçon (seul l'expéditeur doit être capable de générerla signature), assuré alors la non-répudiation.

La signature électronique est basée sur l'utilisation conjointe d'une fonction dehachage, et de la cryptographie asymétrique.

Étapes de signature d'un message :

La signature numérique comprend deux étapes :

a) Évaluation du condensé de message : l'émetteur commence par générer un condensé, qui est une représentation réduite et unique du message complet, à l'aide d'une fonction de hachage.

b) Signature du condensé : l'émetteur chiffre ce condensé avec un algorithme asymétrique à l'aide de sa clé privée. Il obtient une signature électronique qu'il appose au message original avant d'émettre l'ensemble, message et signature, sur le réseau.

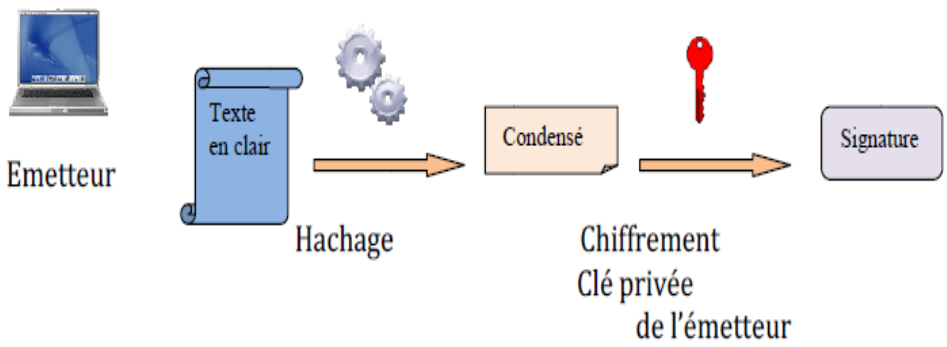


Figure 2.4 : signature d'un message

Etape de vérification de la signature d'un message

La vérification de signature comprend les trois étapes suivantes :

a) Déchiffrement du condensé de message : le condensé est déchiffré à l'aide de la clé publique de l'émetteur.
b) Evaluation du condensé : étant donné que le hachage est un processus unidirectionnel, autrement dit, qu'il est impossible de retrouver le message d'origine à partir du condensé, le destinataire doit réévaluer le condensé en utilisant exactement le même Algorithme de hachage que l'émetteur.
c) Comparaison des condensés. Le condensé déchiffré et le condensé évalué sont comparés. S'ils concordent, la signature est de ce fait vérifiée et le destinataire peut alors avoir la certitude que le message a été envoyé par l'émetteur et n'a pas été altéré. S'ils ne concordent pas, il est possible que le message n'ait pas été signé pas l'émetteur ou que le message ait altéré. Dans les deux cas, me message doit être rejeté. [10]

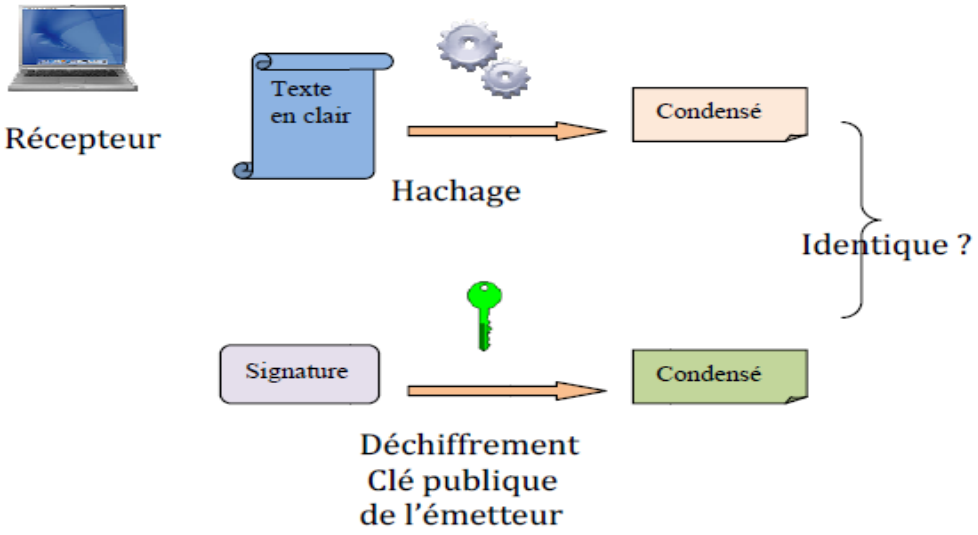


Figure 2.5 : vérification de la signature d'un message

3.5. Certificats électroniques

Un certificat est un élément d'information qui prouve l'identité du propriétaire d'une clé publique. Les certificats sont signés et transmis de façon sécuritaire par un tiers de confiance appelé autorité de certification (Certificate Authority, ou CA). L'autorité de certification est chargée de délivrer les certificats, de leur assigner une date de validité, ainsi que de révoquer éventuellement des certificats avant cette date en cas de compromission de la clé (ou du propriétaire).

La structure des certificats est normalisée par le standard X.509 de l'UIT, qui définit les informations contenues dans le certificat :

- Version
- Numéro de série de l'autorité de certification
- Algorithme de signature du certificat
- Le nom de l'autorité de certification
- Le nom du propriétaire du certificat
- La date de validité du certificat
- Le propriétaire du certificat
- La clé publique du propriétaire

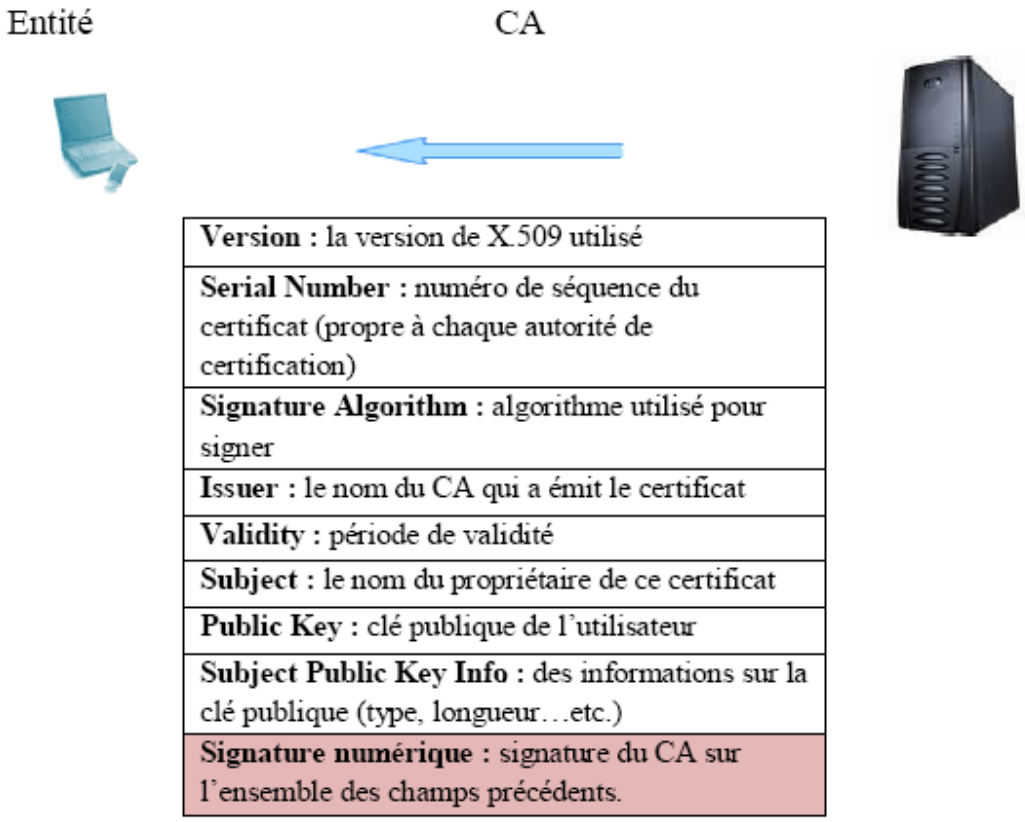


Figure 2.6 : contenue d'un certificat

L'ensemble de ces informations est signé par l'autorité de certification. Cela signifie qu'une fonction de hachage crée une empreinte de ces informations (ou condensé), puis ce condensé est chiffré à l'aide de la clé privé de l'autorité de certification.

Vérification du certificat

Lorsqu'un utilisateur désire communiquer avec un autre, il lui suffit de se procurer le certificat du destinataire. Il est donc possible de vérifier la validité du certificat en appliquant d'une part la fonction de hachage aux informations contenues dans le certificat, en déchiffrant d'autre part la signature de l'autorité de certification avec la clé publique de cette dernière et en comparant ces deux résultats. [10]

3.6. La réputation

Chaque entité réseau encourage la collaboration d'autres entités en utilisant une métrique de coopération appelée réputation. La métrique de réputation est calculée sur la base des données recueillies localement par chaque nœud et peut se baser optionnellement sur l'information fournie par d'autres nœuds du réseau impliqués dans des échanges de messages avec les nœuds surveillés. Une note est attribuée à chaque entité, cette note sera augmentée chaque fois que l'entité participe au routage. Basé sur la réputation, un mécanisme de punition est adopté comme système de dissuasion pour empêcher un comportement égoïste en refusant graduellement les services de communication aux entités qui se conduisent mal. Les nœuds prouvés comme malveillants sont exclu du réseau. Cette décision est prise par une autorité centrale. [11] [12]

4. Attaques et vulnérabilités contre les protocoles de routage

Dans un réseau ad hoc toute les entités peuvent participer au routage, donc il n'y a pas de barrières pour un nœud malicieux de causer des perturbations dans le trafic circulant. L'intérêt de l'attaquant vise essentiellement à compromettre la confidentialité et l'intégrité des informations en transit. Ou de manière plus générale, à perturber le bon fonctionnement du processus de routage pour dominer le réseau.

4.1. Classification des attaques

- **Attaque passive ou active**

Dans les réseaux ad hoc, selon le niveau d'intrusion des actions menées par un attaquant, on distingue généralement deux catégories d'attaques : les attaques passives et les attaques actives.

✓ **Attaque passive**

L'adversaire ne fait que surveiller les canaux de communication. Une écoute se produit lorsqu'un attaquant capture un nœud et étudie le trafic qui le traverse sans en altérer le fonctionnement. Les données analysées aident l'intrus à agir plus tard. Un adversaire passif ne fait que menacer la confidentialité des données.

✓ **Attaque Active**

Une attaque est active lorsqu'un nœud non autorisé altère des informations de routage en transit par des actions de modification, suppression, ou fabrication, ce qui conduit à des perturbations dans le fonctionnement du réseau.

- **Attaque externe ou interne**

En outre, selon le domaine d'appartenance d'un nœud, les attaques actives peuvent elles-mêmes être classées en deux catégories, à savoir les attaques externes et internes. Tandis que les attaques externes sont réalisées par des nœuds qui n'appartiennent pas au domaine du réseau, les attaques internes sont menées par des nœuds compromis qui sont autorisés à participer au fonctionnement du réseau. Etant donné que les attaquants font d'ores et déjà partie du réseau de nœuds autorisés, les attaques internes sont généralement plus pernicieuses et difficiles à détecter que les attaques externes.

- **Attaque individuelle ou attaque distribuée**

En effet, les attaques peuvent être de type individuelles ou par collusion ou appelée également distribuée. Les attaques individuelles sont menées par un seul nœud attaquant. Puisque les capacités de communication et de calcul de l'attaquant sont en général similaires à celles des autres nœuds du réseau, ces attaques demeurent relativement simples, et sont d'autant plus limitées que des mécanismes de sécurité sont mis en œuvre. En revanche, rien n'empêche à des nœuds attaquants de mutualiser leurs informations et leurs ressources, en exploitant les connexions qu'ils ont entre eux. Ces attaques par collusion, issues de plusieurs nœuds répartis à différents endroits dans le réseau, sont généralement plus évoluées et plus dangereuses. Par ailleurs, en raison de l'intervention de plusieurs nœuds intermédiaires, leur détection et l'identification précise de leur origine sont rendues plus complexes.

4.2. Présentation de quelques attaques

✓ **Contrefaçon d'information**

L'attaque la plus directe contre un protocole de routage est de viser les informations échangées entre les nœuds. Les paquets du protocole de routage peuvent être contrefaits, altérés ou rejoués. Ce qui permet aux adversaires de créer des boucles, attirer ou repousser le trafic du réseau,

prolonger ou raccourcir les itinéraires, produire de faux messages d'erreur, diviser le réseau, augmenter la latence, etc. [13]

- ✓ **Suppression des messages**

Un nœud malicieux absorbe les messages en circulation**.** Cette attaque peut paralyser totalement le réseau (perte de connectivité entre les nœuds) si le nombre de messages supprimés est très important.

- ✓ **Replay ou rejeu :** un nœud malicieux réinjecte des messages dans le réseau. Des anciens messages continuent à circuler ce qui occupe de la bande passante et peut même affecter la justesse de la topologie.
- ✓ **Spoofing ou usurpation d'identité** : consiste à se faire passer pour quelqu'un d'autre en utilisant son identité. L'attaquant se présente en utilisant l'identité d'un nœud légitime et peut ainsi communiquer avec les nœuds du réseau sans être rejeté.
- ✓ **Les dénis de services :** denial of services (DoS), apparaissent comme les attaques les plus faciles à réaliser par un attaquant. La criticité de telles attaques dépend fortement du contexte d'utilisation. Les modèles de dénis de services qui suivent se dégagent plus particulièrement dans le cas de réseau sans fil ad hoc :
  - Brouillage du canal radio pour empêcher toute communication.
  - Tentative de débordement des tables de routages des nœuds servant de relais.
  - Non-coopération d'un nœud au bon fonctionnement du réseau dans le but de préserver son énergie. L'égoïsme d'un nœud est une notion propre aux réseaux ad hoc. Un réseau ad hoc s'appuie sur la collaboration sans condition de ses éléments. Un nœud refusant de jouer le jeu peut mettre en péril l'ensemble.
  - Tentative de gaspillage de l'énergie des nœuds. L'attaque consiste à faire en sorte que le nœud soit obligé de rester en état d'activité et ainsi de lui faire consommer toute son énergie. Cette attaque est référencée aussi par l'appellation sleep privation torture attack, un scénario de torture par privation du sommeil.

- ✓ **Brouillage (jamming)**

Le *jamming* est une attaque très connue qui s'en prend à la communication sans fil. En effet, vu la sensibilité du média sans fil au bruit, un noeud peut provoquer un déni de service en émettant des signaux à une certaine fréquence pour interférer avec les fréquences radio employées par les noeuds du réseau. [14]

- ✓ **Attaque du trou noir (sinkhole)**

Dans une attaque *sinkhole*, le nœud malveillant essaye d'attirer vers lui le plus de chemins possibles permettant le contrôle de la plus part des données circulant dans le réseau. Pour ce faire, l'attaquant doit apparaître aux autres comme étant très attractif, en présentant des routes optimales. L'attaquant se place généralement à un endroit stratégique et supprime tous les messages qu'il doit retransmettre ou bien permet la mise en œuvre d'une autre attaque. Créant ainsi une sorte de puits ou « trou noir » dans le réseau. [14]

- ✓ **Trou de ver (Wormhole)**

Dans une attaque *wormhole*, un attaquant reçoit des paquets dans un point du réseau, puis les encapsule vers un autre attaquant pour les réintroduire dans le réseau. Dans ce genre d'attaque, les adversaires coopèrent pour fournir un canal à basse latence pour la communication en utilisant une radio pour communiquer avec une puissance plus élevée et des liens à longue portée. Ceci favorise les nœuds voisins à acheminer leurs données à travers l'attaquant. [15]

✓ Attaque Sybil

Dans cette attaque, le nœud présente des identités multiples aux autres nœuds du réseau, créant ainsi des inconsistances dans les tables de routage des nœuds voisins. Ce qui permet de créer plusieurs routes passant par le nœud malicieux, qui ne sont en réalité qu'un seul chemin. [13]
Après avoir eu une idée globale sur les failles de sécurité menaçant le routage ad hoc, on présente dans ce qui suit l'ensemble des solutions visant l'acheminement sûr des données.

5. Etat de l'art sur le routage sécurisé

5.1. Solutions utilisant la cryptographie

ARAN Sanzgiri et al. ont proposé le protocole sécurisé ARAN (Authenticated Routing for Ad hoc Networks) [16] qui prévoit l'utilisation de la cryptographie à clé publique pour sécuriser la construction des chemins des protocoles réactifs tels que AODV. Il suppose l'existence d'un serveur d'authentification, dont le rôle est de gérer la distribution des certificats pour les nœuds autorisés dans le réseau.
ARAN s'appuie sur deux mécanismes d'authentification. Le premier consiste en une authentification de bout en bout afin qu'un nœud destinataire puisse d'une part authentifier l'origine d'un message de contrôle, et d'autre part vérifier la non modification des données statiques (i.e. l'adresse du nœud source et destinataire) pendant le transit. Le second est une authentification de saut en saut dans lequel chaque nœud sollicité dans un processus de recherche ou de maintenance de chemin utilise sa signature et son certificat pour s'authentifier auprès d'autres nœuds voisins. Une étude comparative entre ARAN et chacun des protocoles AODV et DSR a montré une grande résistance de ce protocole envers les attaques de modification et d'usurpation d'identité.
Mais ARAN s'avère extrêmement coûteux en consommation de ressources à cause du grand nombre des opérations de signature et de vérifications de signature utilisées pour assurer la sécurité.
SAODV Zapata et Asokan ont proposé une extension de sécurité pour le protocole AODV nommée Secure AODV [17]. Contrairement à l'extension ARAN pour laquelle les données variables des messages de contrôle sont retirées, l'idée principale de SAODV consiste à faire usage d'une signature numérique (créée par cryptographie à clé publique) pour protéger les données statiques des messages de contrôle et cela à l'aide d'un algorithme de chiffrement asymétrique (RSA, DSA), puis de recourir à des chaînes de hachage pour protéger l'intégrité de la partie non statique qu'est le compteur de sauts.
Comme pour ARAN, des services d'authentification, d'intégrité et de non-répudiation de bout en bout, entre le nœud source et destination, sont ainsi obtenues. Cependant, l'utilisation des chaînes de hachage pour contrer les manipulations illégales sur le compteur de sauts reste limitée. En outre, dans le cas où plusieurs attaquants sont en collusion, une attaque de type *wormhole* peut être menée. A travers cette attaque, l'attaquant parvient à manipuler le compteur de sauts et à raccourcir la longueur d'un chemin, ceci de manière transparente pour les autres nœuds.
SEAD « Secure Efficient Ad hoc Distance vector routing protocol **»** [18] est un protocole proactif de routage ad hoc sécurisé, basé sur DSDV et permet d'authentifier l'émetteur d'une information de routage. En utilisant les chaînes de hachage à sens unique, SEAD permet d'empêcher l'altération des champs mutables, à savoir le champ métrique nombre de saut et le champ numéro de séquence. En appliquant d'une manière répétitive une fonction de hachage à sens unique, on obtient une chaîne. Les éléments de cette chaîne seront utilisés par les nœuds dans la procédure d'authentification et cela sans utiliser le cryptage à clé publique. Ainsi, il évite les opérations coûteuses dues aux signatures.

Bien que SEAD soit une solution intéressante pour sécuriser le protocole DSDV, il n'est pas suffisant pour empêcher les nœuds malveillants d'agir sur les paquets de données. En effet, la retransmission de ces paquets n'est pas assurée par le protocole de routage et un noeud peut facilement les falsifier, les rejouer, les modifier ou simplement les détruire.

SLSP Papadimitratos et Haas proposent SLSP Secure Link State Protocol [19], un protocole à état de lien dont ils ont modifié les messages de contrôle afin d'en sécuriser le contenu. Ce protocole utilise les signatures numériques ainsi que les chaînes de hachage à sens unique pour garantir l'intégrité des mises à jour de l'état des liens.

L'authentification du message se fait par vérification de la signature avec la clé publique de l'émetteur. Alors que les chaînes de hachage permettent juste de limiter le diamètre de diffusion des messages de mise à jour topologique. En revanche, rien n'empêche un nœud de rejouer la valeur de hachage reçue et/ou d'augmenter le compteur de sauts plus que nécessaire.

Par ailleurs, tout comme pour le protocole SEAD, les paquets de données ne sont pas protégés contre la falsification, le rejeu, la modification ou la destruction. Enfin, SLSP ne permet pas de prendre en compte d'éventuels attaquants complices qui pourraient forger des métriques erronées ou même de créer des tunnels.

5.2. Solutions basées sur la réputation

Cet ensemble de solutions vient remédier au problème du comportement égoïste qui perturbe. Dans le cadre des réseaux ad hoc, la réputation peut être définie comme étant le niveau de participation d'un nœud dans l'opération de retransmission des paquets, tel que vu par d'autres nœuds. La réputation d'un nœud est ensuite utilisée pour prendre des décisions concernant la fiabilité des entités et d'augmenter la confiance au sein du réseau en encourageant la participation au routage.

De façon générale, ces solutions utilisent trois mécanismes distincts :

1- un mécanisme local de surveillance afin d'observer le comportement des nœuds et de déterminer leur degré de réputation selon les actions observées et les échanges entre les nœuds,

2- un mécanisme de sanction ou d'isolation afin de protéger le réseau contre les comportements malveillants, 3- un mécanisme de dissémination des informations collectées.

Watchdog and Pathrater

Les premiers travaux exploitant cette caractéristique pour aborder le problème de non-coopération sont ceux de Marti et al. [20]. Dans leurs travaux, les auteurs traitent le cas du protocole de routage DSR en proposant un système fondé sur deux composants :

Le chien de garde (Watchdog) et **l'évaluateur de chemins** (Pathrater). Le

Watchdog, utilisé localement par chaque nœud, a pour rôle de contrôler que le nœud suivant sur le chemin procède bien à l'opération de retransmission des paquets de données. Lorsqu'une action observée ne correspond pas à un résultat attendu, le nœud observateur comptabilise un échec de retransmission. A partir du moment où le compteur pour un nœud dépasse un seuil fixé, l'information est reportée au Pathrater. Le Pathrater est ensuite utilisé pour sélectionner les chemins les plus fiables entre une source et une destination, en évitant les nœuds qui ont été détectés comme non coopératifs. La faiblesse de cette approche est qu'elle ne permet ni de sanctionner ni d'isoler les nœuds qualifiés de non coopératifs. Ces derniers, bien qu'exclus de la construction des chemins, sont toujours en mesure d'utiliser les ressources des autres nœuds dans le réseau pour leurs propres communications. Ces mécanismes sont également vulnérables, car ils peuvent être détournés par un attaquant. Un nœud malicieux peut facilement faire en sorte qu'un nœud valide soit ajouté à la liste noire, ce dernier sera en conséquence isolé du réseau. [21].

CONFIDANT : D'autres travaux ont traité les problèmes de coopération entre les nœuds par le biais de systèmes de réputation plus complexes. Dans ce registre, Buchegger et Boudec ont

proposé un système de surveillance distribué et collaboratif nommé CONFIDANT (Cooperation of Nodes, Fairness In Dynamic Ad hoc NeTworks)
[22].

L'objectif de CONFIDANT est d'exclure les nœuds qui ne jouent pas leur rôle dans les opérations de routage, que ce soit au niveau du processus d'acheminement des données ou au niveau du processus de découverte des voisins. Il a été conçu comme étant une extension de sécurité des protocoles de routage réactifs à la source tels que DSR. Le système, maintenu par chaque nœud du réseau, définit quatre composants :

- Un moniteur
- Un système de réputation
- Un gestionnaire de confiance
- Un gestionnaire de chemins

Le rôle du moniteur consiste à vérifier, sur la base d'observations directes, le comportement des nœuds à l'égard des opérations de routage. Dès lors que le moniteur détecte un événement suspicieux ou une incohérence, il en informe le système de réputation. Ce dernier a pour rôle de maintenir à jour les valeurs de réputation pour chaque nœud observée. Dans le but de limiter les effets négatifs dus aux imprécisions du mécanisme de détection d'une part, et pour accélérer le mécanisme d'apprentissage des informations servant à évaluer les nœuds d'autre part, les valeurs de réputation sont éventuellement échangées entre les nœuds. Ainsi, en plus de ses observations directes, un nœud intègre les valeurs de réputation des voisins dans le calcul de réputation des autres nœuds. Dans CONFIDANT, seules les valeurs de réputation négatives sont diffusées à travers des messages d'alarme. C'est le rôle du gestionnaire de confiance de prendre la décision d'envoyer ce type de messages, puis de déterminer dans quelle mesure les informations reçues doivent être prises en considération pour le calcul de la valeur de réputation d'un nœud. Finalement, le gestionnaire de chemins évalue les chemins à partir de la topologie du réseau et des informations des autres composants. Il calcule les chemins les plus sûrs en utilisant comme métrique les valeurs de réputation des nœuds, et peut décider de rejeter les demandes de retransmission de paquets pour les nœuds affichant une faible réputation.
CONFIDANT, tout comme autres mécanismes basés sur la réputation, souffre d'un problème lié à la persistance de l'identité d'un nœud : il est suffisant pour un nœud malveillant de changer d'identité réseau pour se débarrasser d'une mauvaise réputation.

5.3. Approches basées sur les IDS

Puisque les systèmes de sécurité basés sur la cryptographie sont parfois coûteux et qu'ils ne permettent pas d'empêcher toutes les catégories d'attaques, d'autres travaux, souvent considérés comme complémentaires à la prévention, ont porté sur la conception de mécanismes de détection d'intrusions.
Un système de détection d'intrusions IDS **[22]** est un processus de contrôle et d'analyse des évènements dans un réseau pour détecter et identifier toute tentative d'attaque. Il permet de détecter les violations de la sécurité dans un système donné et cherche à réduire et réagir contre les éventuelles intrusions. Un IDS comporte trois parties de base : la capture, le traitement, et la réponse.
- Le module de capture responsable de la collecte des activités. Une activité peut inclure un trafic réseau, un comportement d'un utilisateur ou d'une application, des statistiques sur l'usage des ressources, etc.
- Module de traitement d'alertes : responsable du traitement et de l'analyse des activités collectés et repère tout soupçon possible.

- Module de réponse : consiste à répondre à l'attaque et prendre des mesures correctives en se référant au résultat de traitement. [23]

Une architecture de sécurité pour les réseaux ad hoc a été proposée. Les mécanismes de sécurité se basent sur un système de détection d'intrusion distribué et coopératif.

Chaque nœud est équipé d'un IDS local et des agents mobiles autonomes sont mis en œuvre si nécessaire pour collecter les informations stockées sur les autres nœuds.

5.4. Solutions intégrées aux cartes à puces

La solution consiste à joindre un processeur sécurisé (une carte à puce) à chaque nœud. Cette CAP effectuera les traitements sensibles et elle sera totalement séparée du nœud. De cette manière, un nœud ne saura pas le genre de paquets reçus et donc n'a pas de choix à prendre : soit il achemine le paquet sans savoir son contenu ou bien il le bloque et dans ce cas il risque de perdre un message important qui lui est destiné car il ne pourra déchiffrer le message tout seul (la clé symétrique est connue par les CAPs seulement), ou il risque d'être exclu du réseau.

Dans ce type de protocole, le nœud peut se trouver dans deux situations :

1- Le nœud décide d'émettre un paquet. Il le fait passer à sa carte à puce qui vérifie si le destinataire possède une entrée dans sa table de routage (si le chemin existe déjà). Dans ce cas il y a une tentative d'inondation du réseau inutilement (attaque de type dénis de service) et le paquet est ignoré. Autrement, la CAP chiffre le paquet puis rajoute un entête chiffré et l'envoie.

2- Le nœud reçoit un nouveau message. Il n'aura aucune idée sur le contenu du paquet ni sur son type (entête crypté) ce qui va l'inciter à le faire passer à la CAP pour identifier son contenu. A cette étape-là, la carte à puce effectuera les contrôles nécessaires : Cette dernière déchiffre l'entête attachée au paquet avec sa clé symétrique et vérifie si l'adresse de la destination coïncide avec son propre nœud : alors elle passe au nœud le paquet chiffré et la clé de déchiffrement afin qu'il le déchiffre. Sinon, elle chiffre de nouveau l'entête et passe à son nœud le paquet avec l'entête chiffrée ainsi qu'une demande d'inondation vers tous les nœuds se trouvant dans la portée radio.

L'utilisation des CAPs a permis d'allégé le nœud des différents traitements de routage et du contrôle, ils permettent également de surveiller son comportement, et noter sa participation au routage. Mais ce protocole présente un grand inconvénient qui est la grande taille des paquets qui circulent sur le réseau et contenant l'identité de tous les nœuds de la source au destinataire et qui demeurent lourd pour les réseaux de grandes tailles où les routes deviennent très longues, le trafic sera énorme et difficile à gérer.

5.5. Analyse des solutions de routage sécurisé

Notre étude des protocoles les plus connus qui sécurisent le routage ad hoc finit par une comparaison entre les caractéristiques de chacun. Cette comparaison prouve que la sécurité de routage dans les réseaux ad hoc est un problème complexe. Chaque schéma a ses propres besoin et contraintes qui s'imposent pour atteindre la sécurité voulu.

En effet, les schémas d'authentification basés sur la cryptographie symétrique sont peu recommandés en raison de l'absence de réelle confiance mutuelle entre les nœuds du réseau. C'est pourquoi la plupart des solutions se basent sur la cryptographie asymétrique et plus particulièrement, sur l'utilisation d'autorité de certification toujours en ligne. Néanmoins, la présence de cette autorité contredit la propriété des réseaux ad hoc qui ne prévoit l'existence d'aucune infrastructure centralisée. De plus, les protocoles souffrent du surcoût de calcul apporté par la stratégie de clés publiques (asymétriques).

Parmi les moyens classiques qui assurent l'intégrité et l'authentification des messages échangés, on trouve l'utilisation de signatures numériques ou de MACs (Message Authentification Code). La signature numérique s'appuie sur la cryptographie à clé publique qui est robuste mais complexe.

Tel qu'elle est exploitée dans la majorité de protocoles existants (Signature et vérification à chaque saut) la signature dégrade les performances du système.
En ce qui concerne les systèmes de détection d'intrusions, il convient de souligner les limites de certaines solutions. Par exemple, la négligence de la punition des nœuds qui ne coopèrent pas ou ceux qui montrent un comportement égoïste, ou encore les issues supposant l'existence d'une infrastructure dédiée ce qui est en contradiction avec la dynamique qui caractérise les réseaux ad hoc. De plus, les systèmes basés sur l'étude statistique de comportement possèdent toujours des failles : souvent un nœud malicieux change son identité pour éviter d'être rejeté du réseau par le système de détection d'intrusion.
Un autre point important à noter est le coût des solutions dépendent de l'importance des données à sécuriser. Il peut être très peu pratique de protéger des données dont le coût des pertes engendrées dans le cas où ces données sont corrompues est beaucoup moins important que le coût de la mise en place du système de sécurité utilisé pour sécuriser ces données. Ainsi, il faut bien définir la sensibilité de données manipulées et évaluer le degré de sécurité requis pour garantir un bon fonctionnement au moindre coût.

Conclusion

Dans un réseau ad hoc, tous les nœuds doivent participer aux opérations de routage. Ils gèrent entre autre l'établissement des chemins, la dissémination de notifications de ruptures de chemins et la retransmission des données. Etant donné cette caractéristique, il devient relativement facile pour un nœud malveillant de mener divers types d'attaques, rendant ainsi le réseau inopérant. Qu'il s'agisse de nœuds malveillants internes ou bien de nœuds normaux compromis par un attaquant au cours de l'exploitation du réseau. Ces nœuds déviants sont particulièrement difficiles à contenir.
La sécurité du routage dans les réseaux ad hoc est complexe. Nous pouvons constater qu'il n'existe pas de solution complète pour remédier à ce problème. Plusieurs schémas de sécurité ont été proposés chacun de ces schémas à ses propres besoin et contraintes qui s'imposent pour atteindre la sécurité voulue. Les protocoles se basant sur un mécanisme cryptographique requièrent un schéma de distribution et de gestion de clés.
Les protocoles se basant sur la réputation incluent une nouvelle métrique (degré de fiabilité du chemin) pour sélectionner une route vers la destination. Enfin il n'existe pas un schéma résistant à toutes les attaques et les vulnérabilités. En ce qui concerne les problèmes de sécurité, il faudra probablement toujours trouver des systèmes de plus en plus complexes pour faire face à la ténacité et l'ingéniosité des pirates qui chercheront toujours à relever le défi. La solution optimale de sécurité n'existe pas. D'une part, les techniques proposées offrent des solutions partielles et adaptées à quelques failles seulement. Les techniques les plus élaborées sont très coûteuses.
Les réseaux ad hoc constituent de par leur nature, un formidable challenge pour la sécurité informatique. Ce sujet va devenir d'autant plus critique que le développement de tels réseaux va rapidement s'amplifier.

CHAPITRE

3

La détection d'intrusion

Introduction :

Les systèmes informatiques sont aujourd'hui de plus en plus ouverts sur Internet. Il en découle un nombre croissant d'attaques. Une politique de sécurité autour de ces systèmes est donc primordiale. Outre la mise en place d'un pare-feu, de systèmes d'authentification, il est nécessaire pour compléter cette politique, d'avoir des outils de surveillance pour auditer le système d'information et détecter d'éventuelles intrusions.

Au cours de ce chapitre nous verrons comment se protéger efficacement face aux intrusions, en présentant les systèmes de détection d'intrusions.

1. Définitions :

1.1. Intrusion : [41]

Une intrusion est toute utilisation d'un système informatique à des fins autres que celles prévues, généralement dues à l'acquisition des privilèges de façon illégitime.

Une intrusion est définie aussi comme un ensemble d'actions qui essayent de compromettre :

- La confidentialité.
- L'intégrité.
- La disponibilité d'une ressource ou d'un service.

En dépit des différentes formes d'intrusions, elles peuvent être regroupées dans deux classes : **[42]**

- **Les intrusions connues :** Ces intrusions sont des attaques bien définies qui généralement exploitent des failles connues du système cible.
- **Les intrusions inconnues ou anomalies :** Ces intrusions sont considérées comme des déviations du profil normal d'un système. Elles sont détectées dès qu'un comportement anormal du système est observé.

1.2. Détection d'intrusions : [43]

La détection d'intrusion est un autre procédé utilisable par le personnel de sécurité pour protéger l'entreprise contre les attaques. Elle est définie comme étant un ensemble de techniques et de méthodes qui sont employées pour détecter l'activité soupçonneuse sur un système ou un réseau.

1.2.1. Autre définition : [44]

La détection d'intrusion est le processus qui consiste à surveiller les évènements se produisant dans un ordinateur ou dans un réseau informatique, et de les analyser pour découvrir des signes d'intrusions, définies comme des tentatives de compromission de la confidentialité, l'intégrité, la disponibilité et la responsabilité, ou pour dévier des mécanismes de sécurité

1.3. Système de détection d'intrusions (Intrusion Detection System « IDS ») : [45]

IDS signifie Intrusion Detection System (Système de détection d'intrusions). Il s'agit d'un équipement permettant de surveiller l'activité d'un réseau ou d'un hôte donné, afin de détecter toute tentative d'intrusion et éventuellement de réagir à cette tentative.

2. Architecture classique d'un système de détection d'intrusions : [46]

Les systèmes de détection d'intrusions ont un modèle commun, composé de trois modules comme le montre la figure suivante:

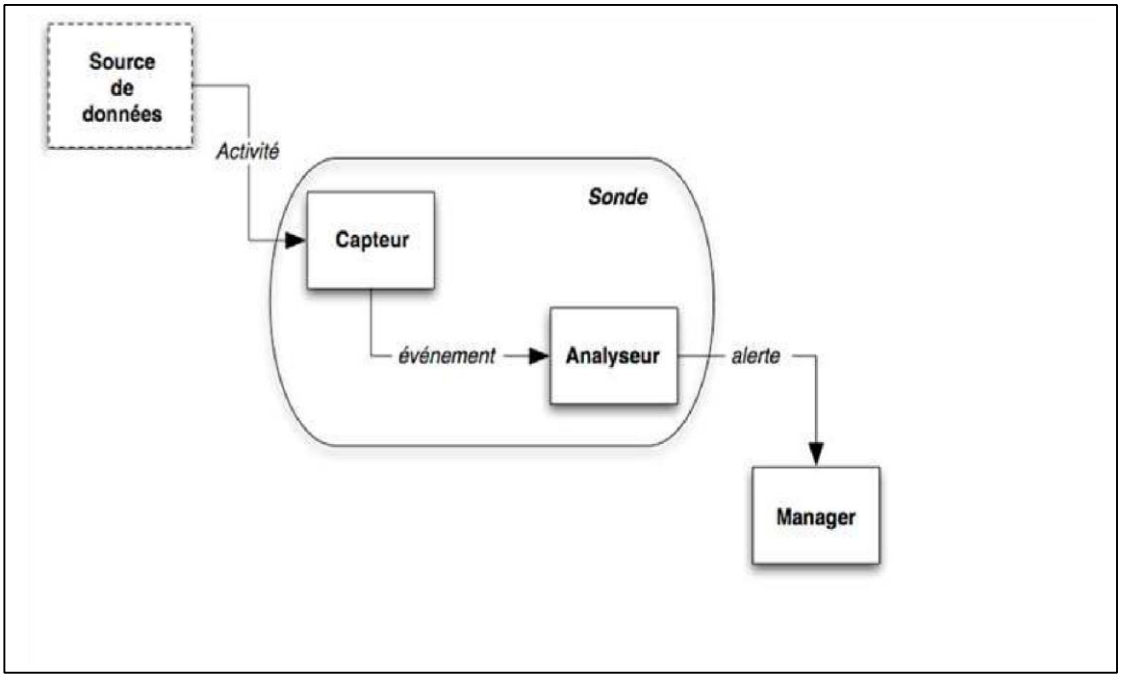


Figure 3.1: Architecture classique d'un IDS (Intrusion Detection System)

Le 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 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. De plus, le capteur réalise généralement une mise en forme des données brutes acquises, afin de présenter à l'analyseur des données utilisant un certain format d'événements.

L'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. Deux grandes approches ont été proposées : l'approche comportementale et l'approche par scénarios et qui seront détaillées par la suite.

Le manager : il est responsable de la mise en place de procédure de réaction à l'intrusion comme le blocage du flux de données suspect, de certains ports, etc.

Debar **[46]** simplifie le système de détection d'intrusions dans un détecteur qui analyse les informations en provenance du système surveillé (voir figure 3.2).

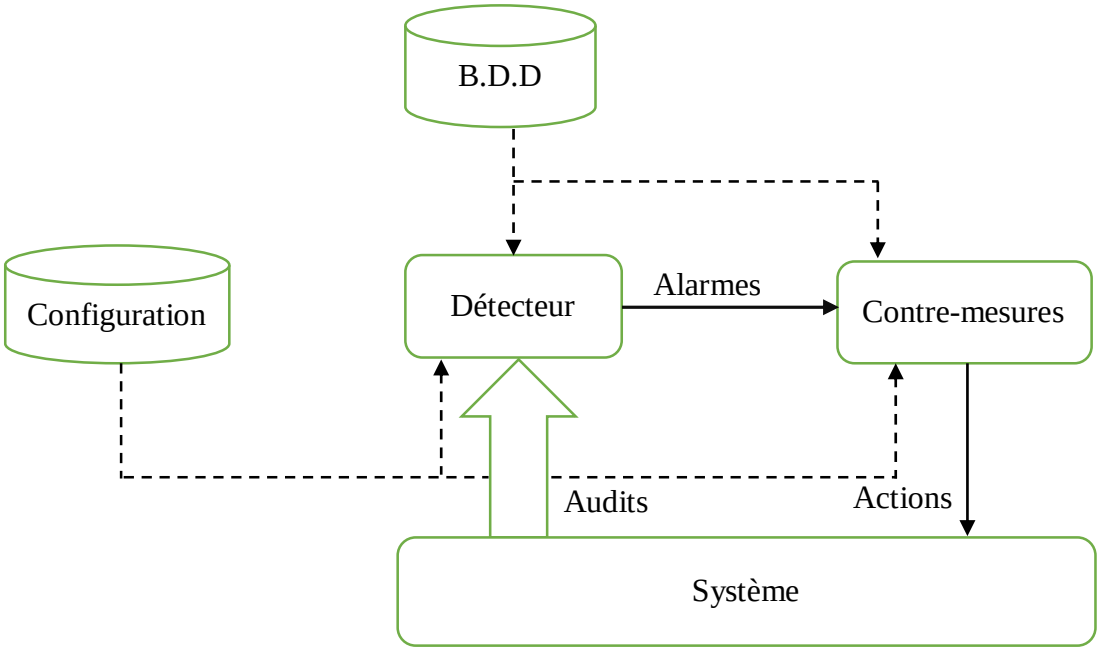


Figure 3.2 : *Modèle simplifié d'un système de détection d'intrusions.*

Le détecteur analyse trois types d'informations : les informations de long terme relatives aux techniques utilisées dans la détection (Base de données de signatures),

les informations de configuration qui déterminent l'état courant du système, et les informations d'audit qui décrivent les événements survenus dans le système.

Philip **[56]** définit trois critères pour évaluer l'efficacité des systèmes de détection d'intrusions.

-L'exactitude ; On parle de l'exactitude quand les systèmes de détection d'intrusions déclarent comme malicieux une activité légale. Ce critère correspond au faux positif.

-La performance ; La performance de système de détection d'intrusions est le taux de traitement des événements. Si ce taux est faible, la détection en temps réel est donc impossible.

-La complétude ; On parle de la complétude quand le système de détection d'intrusions rate la détection d'une attaque. Ce critère est le plus difficile parce qu'il est impossible d'avoir une connaissance globale sur les attaques. Ce critère correspond au faux négatif.

Debbar **[46]** a rajouté également les deux critères suivants :

-La tolérance aux fautes ; Le système de détection d'intrusions doit lui-même résister aux attaques, particulièrement au déni de service. Ceci est important parce que plusieurs systèmes de détection d'intrusions s'exécutent sur des matériels ou logiciels connus vulnérables aux attaques.

-La réaction à temps ; Le système de détection d'intrusions doit s'exécuter et propager les résultats de l'analyse le plus tôt possible, pour permettre à l'officier de sécurité de réagir avant que des graves dommages n'aient lieu. Ceci implique plus qu'un calcul de performances, parce qu'il ne s'agit pas seulement du temps de traitement des événements, mais aussi du temps nécessaire pour la propagation et la réaction à cet événement.

3. Les caractéristiques souhaitées d'un IDS : [47]

Les caractéristiques suivantes sont souhaitables dans un IDS :

- Il doit fonctionner de manière continue avec une présence humaine minimum.
- Il doit être tolérant aux fautes c'est-à-dire qu'il doit être capable de retrouver son état initial de fonctionnement après un crash causé soit par une manipulation accidentelle soit par des activités émanant de personnes malintentionnées.

- Il doit résister à la subversion. L'IDS doit être capable de se contrôler lui-même et de détecter s'il a été modifié par un attaquant.
- Il doit imposer une supervision minimale du système sur lequel il tourne afin de ne pas interférer avec ses opérations normales.
- Il doit être configurable d'après les politiques de sécurité du système qu'il supervise.
- Il doit également être capable de s'adapter aux changements du système et des comportements des utilisateurs au cours du temps.

Lorsque le nombre de systèmes à superviser augmente et donc les attaques potentielles augmentent également, nous pouvons alors attendre de l'IDS les caractéristiques suivantes :

- Il doit être capable de superviser un nombre important de stations tout en fournissant des résultats de manière rapide et précise.
- Il doit fournir "un service minimum de crise" c'est-à-dire que si certains composants de l'IDS cessent de fonctionner, les autres composants doivent être affectés le moins possible par cet état de dégradation.
- Il doit autoriser des reconfigurations dynamiques. Si un grand nombre de stations est supervisé, il devient pratiquement impossible de redémarrer l'IDS sur tous les hôtes lorsqu'on doit effectuer un changement.

4. Classification des systèmes de détection d'intrusions : [46]

Nous pouvons classifier les systèmes de détections d'intrusions selon cinq critères (cités ci-dessous) qui ne sont pas forcément mutuellement exclusif (figure 3.3) :

a) Méthode de détection utilisée.
b) Mode de fonctionnement des mécanismes de détections.
c) Sources de données à analyser.
d) Réponse aux attaques.
e) Fréquence d'utilisation.

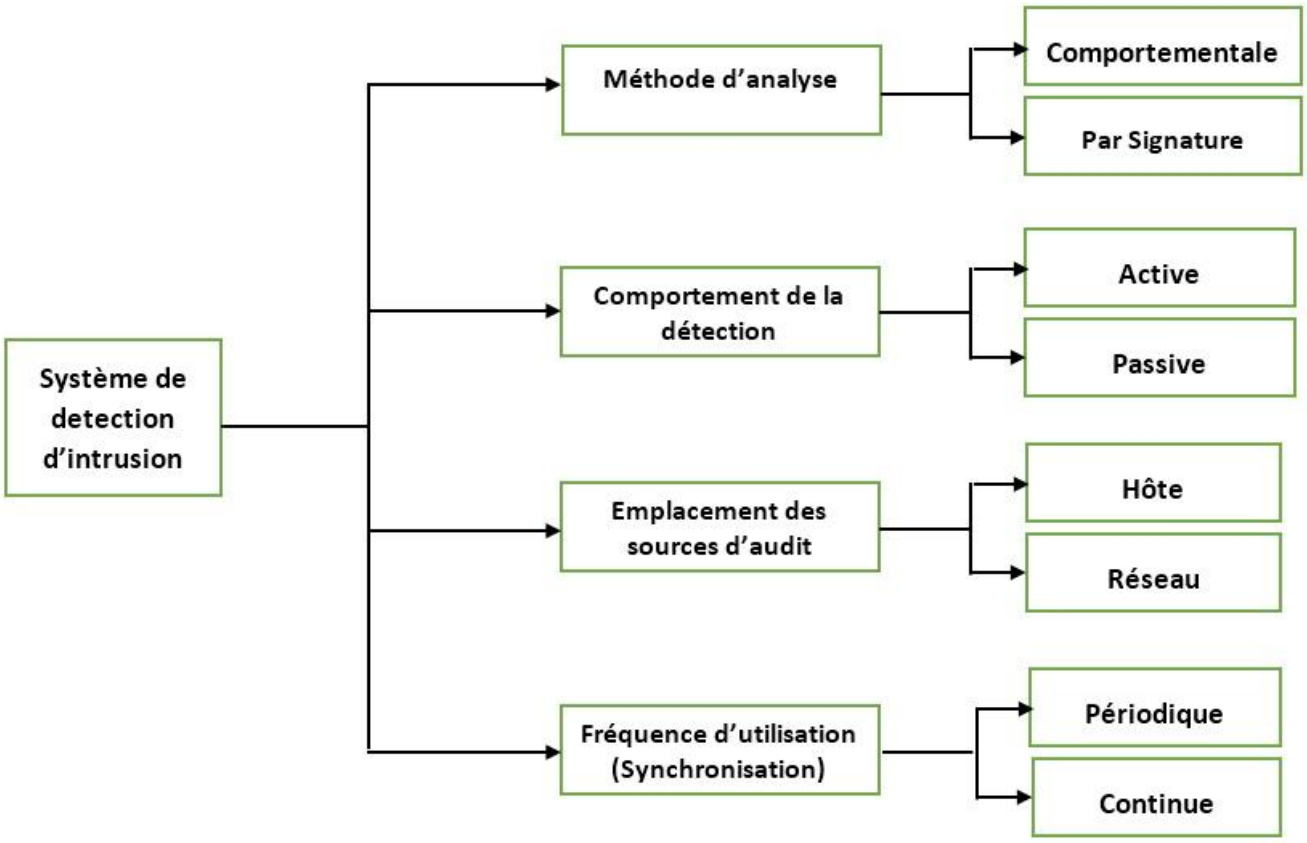


Figure 3.3 : Classification des systèmes de détection d'intrusions

Il existe plusieurs critères qu'on peut utiliser pour classifier les différents systèmes de détection d'intrusion, dont les principaux sont résumés dans la **figure 3.3**

4.1. La méthode d'analyse

La méthode d'analyse définit l'ensemble des techniques utilisées par les systèmes de détection d'intrusion dans le processus de la détection. L'approche est dite d'***abus*** si le détecteur analyse les informations relatives aux attaques, et elle est dite ***comportementale*** si le détecteur analyse les informations relatives au comportement normal du système [46].

Dans la première méthode, l'analyse vise quelque chose de connu qualifié de "mauvais". Cette technique est utilisée par la plupart des systèmes commerciaux. Dans la deuxième méthode, l'analyse cherche les modèles anormaux de l'activité. D'une autre façon, la détection d'abus se base sur les caractéristiques d'une attaque connue pour la détecter. Par contre, la détection d'anomalie se base sur la définition d'un modèle d'utilisation normale pour détecter tout ce qui est anormal. Chaque approche présente des avantages et des inconvénients.

4.1.1 La détection d'abus (par scénario)

La détection d'abus (mauvaise utilisation) considère comme normal tout ce qui n'est pas hostile, et elle adopte la politique suivante : " *si ce n'est pas dangereux, alors c'est normal* ".
Donc, il est impératif de bien connaître les attaques possibles.

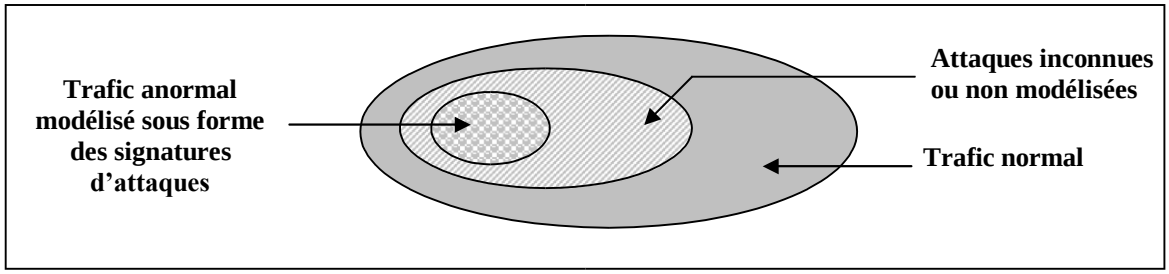


Figure 3.4 : Détection d'abus

Avantages :

- Très efficace pour détecter des attaques sans produire un grand nombre de fausses alarmes.
- Peut rapidement et sûrement diagnostiquer l'utilisation d'un outil spécifique ou une technique d'attaque. Ceci peut aider les responsables de sécurité à donner la priorité aux mesures correctives.

Inconvénients :

- Peut seulement détecter les attaques connues, dont les signatures sont introduites dans le système, donc le système de détection doit être constamment mis à jour avec les signatures des nouvelles attaques (voir la **figure 3.4**).
- Beaucoup de systèmes adoptant cette approche sont conçus pour employer un nombre limité de signatures qui peuvent être définis, ce qui les empêchent de détecter des variantes de ces attaques.

4.1.2 La détection d'anomalie (comportementale)

La détection par anomalie consiste à considérer comme hostile tout ce qui n'est pas normal, au sens où on cherchera plutôt à bien définir ce qui est un comportement normal sur le réseau pour

pouvoir y opposer toute déviation, que l'on considérera comme étant une attaque : " *si ce n'est pas normal, alors c'est dangereux* ". Cette approche comprend donc deux phases :

1. Extraction d'informations sur le milieu, afin de définir la "normalité". Deux techniques sont utilisées : les statistiques et l'intelligence artificielle.

2. Etablir les limites de la "normalité", au-delà desquelles le comportement est nécessairement anormal.

La détection par anomalie revient donc à repérer tout ce qui sort du cadre de la normalité.

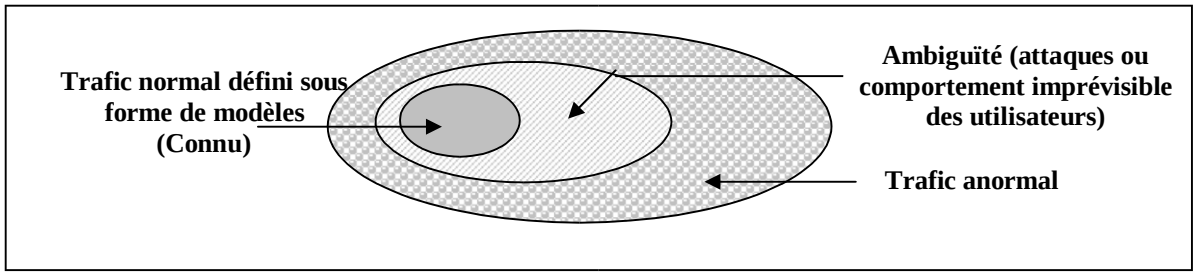


Figure 3.5 : Détection d'anomalie

Avantages :

- Les systèmes de détection d'intrusion basés sur la détection d'anomalie détectent le comportement peu commun, et ils ont ainsi la capacité de détecter des symptômes des attaques connues et inconnues sans la connaissance spécifique des détails.

- Cette approche permet de produire l'information utile pour la définition des signatures pour les systèmes de détection d'intrusion à base de signatures.

Inconvénients :

- Le point noir de cette approche est le grand nombre de fausses alarmes dues aux comportements imprévisibles des utilisateurs du réseau.

- Elle exige souvent l'historique à long terme des évènements enregistrés afin de caractériser les modèles normaux de comportement. Les systèmes basés sur cette approche doivent être dotés d'une certaine intelligence pour raison d'apprentissage automatique en utilisant par exemple les réseaux de neurones.

4.2 Le comportement de la détection (la réponse)

Le comportement de la détection décrit la réponse du système de détection d'intrusion à une attaque. Elle est qualifiée d'active, si le détecteur réagit activement par des actions correctives, ou proactives (changer les règles de filtrage de Firewall, arrêter des connexions *TCP*, ou encore attaquer l'attaquant, etc.). Si le système de détection d'intrusion génère simplement des alarmes (afficher un message sur l'écran, générer un son spécifique, envoi d'un email, archivage dans un fichier ou dans une base de données, etc.), la réponse est qualifiée de passive.

4.2.1 Les réponses actives

Les réponses actives des *IDSs* sont des actions automatisées prises quand certains types d'intrusions sont détectés. Il y a trois catégories de réponses actives :

1. **Rassembler des informations additionnelles :** Il est très important de rassembler des informations additionnelles sur une attaque afin de l'identifier avec précision. Chacun de nous a fait probablement l'équivalant de cela une fois réveillé par un bruit étrange pendant la nuit. La première chose à faire dans une telle situation est d'écouter d'avantage, recherchant l'information additionnelle qui nous permet de décider si on doit agir ou non. Dans le cas des *IDSs*, cela se traduira par l'exigence d'analyse des informations additionnelles, faire des corrélations, ou bien communiquer avec d'autres types d'*IDSs* installés sur le réseau.

2. **Changer l'environnement :** Une autre réponse active doit stopper une attaque en progression et puis bloquer l'accès de l'attaquant. Typiquement, les *IDSs* n'ont pas les capacités de bloquer l'accès d'une personne spécifique, mais ils peuvent uniquement rompre des connexions ou bloquer certains paquets spécifiques en s'appuyant sur les mécanismes des protocoles Internet, cela est dû à la capacité du hacker expert de construire des paquets falsifiés (forging packets). Parmi ces actions on trouve :

- L'envoi des paquets *TCP* de type Reset ou des paquets *ICMP* au système de l'attaquant pour arrêter la connexion.
- La configuration des routeurs et des Firewalls pour bloquer les paquets provenant de l'adresse *IP* de l'attaquant.
- La configuration des routeurs et des Firewalls pour bloquer les paquets selon le numéro de port, le protocole, ou le service utilisé par l'attaquant.

3. **Agir contre l'intrus :** La première option dans la réponse active est d'agir contre l'intrus. En effet, la forme la plus agressive de cette réponse implique le lancement des contre-attaques ou d'essayer d'obtenir activement les informations sur le hôte ou l'emplacement de l'attaquant. Cependant, à cause des ambiguïtés légales au sujet de la responsabilité civile, cette option peut représenter un grand risque qu'une contre-attaque réussie.

La première question concernant le choix de cette option même avec beaucoup d'attention est : « est ce que notre action peut être illégale ? ». Beaucoup d'attaquants emploient de fausses adresses de réseau quand ils attaquent les systèmes, ce qui peut être la cause d'endommagement des sites Internet ou de torts causés aux utilisateurs innocents. En conclusion, il faut prendre ces actions avec plus de prudence.

4.2.2 Les réponses passives

Les réponses passives des *IDSs* fournissent l'information nécessaire aux administrateurs réseau et aux responsables de la sécurité pour les aider à prendre des mesures basées sur cette information. Beaucoup d'*IDSs* se fondent seulement sur des réponses passives dont les principales sont :

✓ **L'alarme** : Les alarmes sont produites par les *IDSs* pour informer les administrateurs réseau quand des attaques sont détectées. La forme la plus commune est d'afficher un message d'alerte contenant des informations détaillées de l'intrusion détectée sur la console du responsable de la sécurité réseau. Une autre option très utile consiste à envoyer ces alertes au téléphone du responsable, on peut aussi envoyer des e-mails, ou générer des alertes sonores.

✓ **SNMP Trap :** Certains *IDSs* sont conçus pour produire des alertes et envoyer les rapports au système de gestion de réseau (network management system). Ils utilisent le protocole *SNMP* (*Sample Network Management Protocol*), qui est un protocole dédié à la gestion du réseau.

✓ **L'archivage** : L'archivage (logging) permet aux analystes de faire des analyses approfondies, et de faire des corrélations avec l'historique dont ils disposent concernant les évènements qui se sont produits auparavant.

4.3 *L'emplacement des sources d'audits*

La manière la plus connue pour classifier les *IDSs* est de les grouper par sources d'informations (sondes). Certains *IDSs* analysent des paquets capturés à partir du réseau, en plaçant des *sniffers* sur les différents segments du réseau local. D'autres *IDSs* analysent des informations produites par le système d'exploitation ou par des applications pour la recherche des signes d'intrusions.

4.3.1 NIDS (Network-Based IDS)

Ces outils analysent le trafic réseau, ils comportent généralement une sonde qui "écoute" sur le segment de réseau à surveiller et un moteur qui réalise l'analyse du trafic afin de détecter les signatures d'attaques. Les *IDSs* réseau à base de signatures sont confrontés actuellement à deux problèmes majeurs qui sont : l'utilisation grandissante du cryptage, et les réseaux commutés. En effet, d'une part, le cryptage rend l'analyse du contenu des paquets presque impossible, d'autre part il est plus difficile "d'écouter" sur les réseaux commutés. La plupart des *NIDS* sont aussi dits *IDS on-line* car ils analysent le flux en temps réel. Pour cette raison, la question des performances est très importante. De tels *IDSs* doivent être de plus en plus performants afin d'analyser les volumes de données de plus en plus importants pouvant transiter sur les réseaux.

4.3.2 HIDS (Host-Based IDS)

Les *IDSs* de ce type analysent le fonctionnement et l'état des machines sur lesquelles ils sont installés afin de détecter les attaques. Pour cela ils ont pour mission l'analyse des journaux système (logs), le contrôle d'accès aux appels systèmes, la vérification d'intégrité des systèmes de fichiers, etc. Ils sont très dépendants de système sur lequel ils sont installés. Il faut donc employer des outils spécifiques en fonction des systèmes déployés. Ces *IDSs* peuvent s'appuyer sur des fonctionnalités d'audit propres ou non au système d'exploitation, pour en vérifier l'intégrité, et générer des alertes. Il faut cependant noter qu'ils sont incapables de détecter les attaques exploitant les faiblesses de la pile TCP/IP du système, typiquement les *Dénis de Service*.

4.3.3 IDS d'application

Similaires aux *HIDSs*, ils sont installés sur un serveur ou une machine pour détecter les attaques relatives à une application donnée. Par exemple un *IDS* installé sur un serveur Oracle pour détecter les intrusions relatives à Oracle.

4.3.4 IDS hybrides

Les IDS hybrides rassemblent les caractéristiques de plusieurs *IDS* 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 ces sondes remontent alors les alertes à une machine qui va centraliser, agréger, et lier les informations d'origines multiples.

4.4 La fréquence d'utilisation (la synchronisation)

La synchronisation se rapporte au temps écoulé entre les évènements qui sont surveillés et l'analyse de ces évènements. Elle est réalisée en : ***temps réel*** *ou* ***différé.***

4.4.1 En temps différé (Périodique)

Dans cette classe, le flux d'informations émanant des points de surveillance vers les détecteurs n'est pas continu. En effet, l'information est traitée dans un mode semblable au principe "*emmagasiner et expédier*". Cette approche est employée surtout dans les *Host-IDSs* qui scrutent les logs du système d'exploitation dans des intervalles de temps réguliers.

4.4.2 En temps réel (Continu)

Les *IDSs* en temps réel traitent des flux continus d'informations à partir des différentes sources d'informations. C'est la technique prédominante de synchronisation pour les *IDSs* réseau, qui recueillent l'information du trafic réseau. Par conséquent Les *IDSs* peuvent prendre des actions pour affecter la progression d'une attaque détectée.

Rebbeca [44] a étudié d'autres critères de classifications moins connus, tels que l'architecture et la stratégie de contrôle :

4.5 L'architecture

L'architecture des *IDS*s se rapporte à la manière d'ajuster leurs composants architecturaux. Les composants architecturaux de base sont l'hôte (Host) où l'*IDS* s'exécute et la cible (Target), qu'il soit hôte ou réseau que l'*IDS* doit protéger. Les principales architectures types sont :

4.5.1 Host Target Colocation (Cohabitation de la cible et l'hôte)

Les premiers *IDS*s fonctionnaient sur les systèmes qu'ils étaient censés protégés. Ceci était dû au fait que la plupart des systèmes étaient des systèmes centraux (mainframe), et le coût élevé d'ordinateurs faisait d'une architecture séparée un mauvais choix.

Ceci présente un problème de point de vue de la sécurité. En effet, n'importe quel attaquant qui réussit une attaque sur le système peut neutraliser l'*IDS*, étant donné que l'emplacement de ce dernier est connu.

4.5.2 Host Target Separation (Séparation entre la cible et l'hôte)

Avec l'arrivée des postes de travail et des ordinateurs individuels, la plupart des architectures des *IDSs* ont orienté les systèmes d'analyse et de commande vers un système séparé, par conséquent, séparant la machine d'*IDS* et la cible. Ceci a amélioré la sécurité des *IDS*s parce qu'il est devenu plus facile de cacher leurs existences.

4.6 *La stratégie de contrôle*

La stratégie de contrôle décrit comment les éléments d'*IDS* sont commandés, et comment les entrées et les sorties d'*IDS* sont contrôlées. Elle peut être : *centralisée, Partiellement distribuée, ou Entièrement distribuée.*

4.6.1 Centralisée

La stratégie centralisée consiste à commander à partir d'une console centrale la surveillance, la détection et l'audit.

Dans cette stratégie, il y a une seule console de commande. A partir de laquelle on communique avec les différents systèmes de surveillance : réseau (*NIDSs*), hôte (*HIDSs*) et applications, et s'il y a des indices d'intrusion, on lance des commandes pour changer les règles de sécurité au niveau des firewalls et routeurs.

4.6.2 Partiellement distribuée

La surveillance et la détection sont commandées à partir d'un nœud local de commande, avec un système hiérarchique d'audit.

Dans ce cas de figure, on trouve dans chaque sous réseau une console qui fournit des rapports à la console de niveau supérieur (entreprise IDS consol). Les actions sont prises à partir de cette dernière. Si l'entreprise comporte des réseaux géographiquement séparés, un système indépendant est mis en œuvre pour la surveillance, la détection, et la réaction. La console principale a comme but de commander les différents *IDS*s de l'entreprise.

4.6.3 Entièrement distribuée

La surveillance et la détection sont réalisées en utilisant une approche basée sur des agents, où les décisions de réponse sont prises au lieu où l'analyse s'effectue (IDSs autonomes).

5. Emplacement des systèmes de détection d'intrusions : [48]

Le choix de l'emplacement est un sujet récurrent lorsqu'on aborde les IDSs. Ce problème reste souvent sans réponse claire car la solution idéale n'existe pas vraiment, tout dépend de l'architecture réseau de l'entreprise, de ses choix au niveau de sa politique de sécurité.

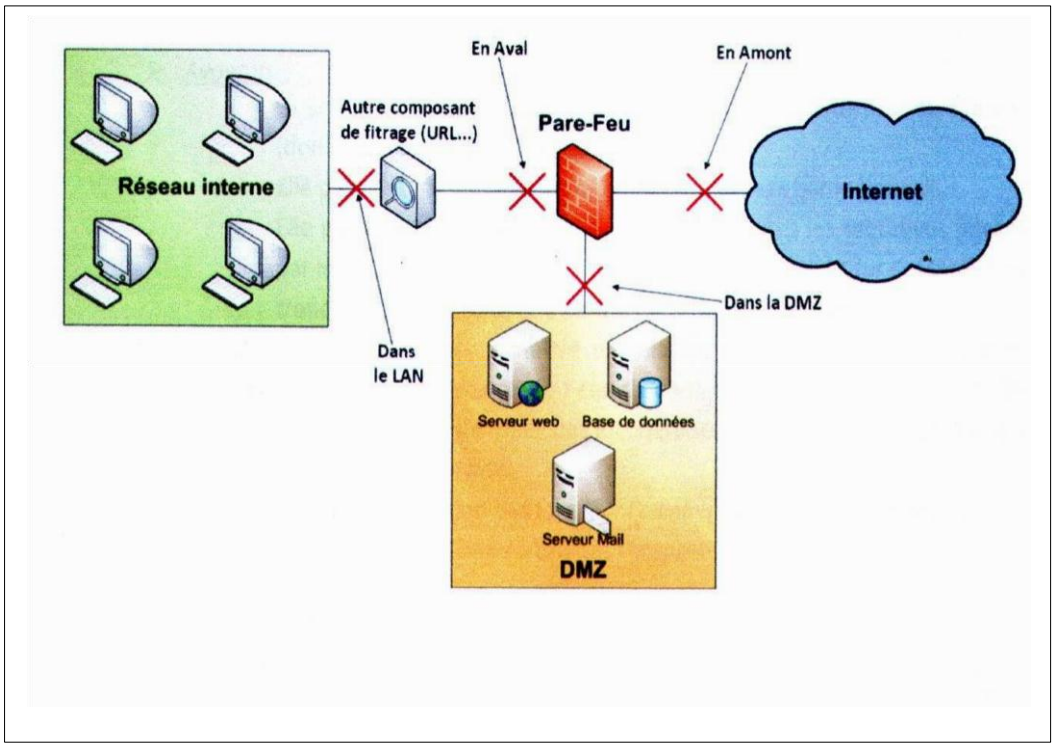


Figure 3.6 : Les emplacements possibles des NIDS (Network IDS)

5.1. En amont : c.-à-d. situé à l'extérieur du périmètre de protection du firewall

✓ **Avantage :**

- Il permet d'avoir une visibilité totale sur le trafic entrant, cela permet bien entendu de repérer les tentatives d'attaques même si celles-ci seront filtrées par le firewall.
- Il aide à analyser la pertinence du pare-feu comme équipement de protection anti-intrusion (en faisant un rapprochement avec un NIDS en aval du firewall).

✓ **Inconvénients :**

- Beaucoup d'alertes seront remontées ce qui rendra les fichiers log ou la base de données des alertes difficilement consultable.
- Le risque engendré par un trafic très important qui pourrait entrainer une perte de fiabilité, car au même temps que le N-IDS analyse un paquet, autres paquets circulant dans le réseau vont passer sans être analysés, et cela empire d'autant plus que le nombre de règles à consulter est grand.
- Etant situé hors du domaine de protection du firewall, la sonde est alors exposée à d'éventuelles attaques pouvant la rendre inefficace.

5.2. En aval : c.-à-d. à l'intérieur du périmètre de protection du firewall

✓ **Avantage :**

- La sonde se place à l'intérieur du périmètre de protection du firewall (donc elle est protégée des attaques).
- Elle peut détecter toutes les attaques non filtrées par le firewall.
- Elle permet de se prémunir des alertes de toutes les tentatives filtrées par le firewall, cela permet à l'IDS de se concentrer sur l'essentiel du trafic entrant (on évite une bonne partie de faux positifs).
- Rend la consultation des logs (par l'administrateur) plus facile, puisque les attaques bénignes ne seront pas recensées.

✓ **Inconvénients :**

- Le risque engendré par un trafic très important qui pourrait entrainer une perte de fiabilité du N-IDS, (une perte de fiabilité minime comparée au premiers cas (c.-à-d. IDS en amont)).

5.3. Dans la DMZ :

Dans cette position, on a les mêmes avantages et inconvénients (qu'en aval). La sonde placée dans cette zone permet de détecter les intrusions qui ont atteint la zone DMZ, et ainsi, surveiller les attaques dirigées vers les différents serveurs de l'entreprise accessibles de l'extérieur.

5.4. Dans le LAN :

Placer une sonde dans le réseau interne de l'entreprise permet d'observer les tentatives d'intrusions parvenues à l'intérieur du réseau, ainsi que les attaques à partir de l'intérieur (qui sont souvent les plus dangereuses). Dans le cas d'entreprises utilisant largement l'outil informatique où la gestion de leur activités ou de réseaux fournissant un accès à des personnes peu soucieuses de la sécurité (réseau d'une université), cette position peut revêtir un intérêt primordial.

6. Imperfection dans les implémentations actuelles des IDS : [42]

Dans la plupart des cas, les systèmes de détection d'intrusions sont faits d'un seul bloc ou module qui se charge de toute l'analyse. Ce système monolithique demande qu'on lui fournisse beaucoup de données d'audit, ce qui utilise beaucoup de ressources de la machine surveillée. L'aspect monolithique pose également des problèmes de mises à jour et constitue un point d'attaque unique pour ceux qui veulent s'introduire dans le système d'information.

D'autres imperfections plus générales sont relevables dans les systèmes de détection d'intrusions actuels :

- Même en implémentant les deux types d'approches, certaines attaques sont indécelables et les systèmes de détection sont eux-mêmes attaquables. Les approches comportementale et par scénarios ont elles-mêmes leurs limites.
- Les groupes de travail sur ce sujet sont relativement fermés et il n'y a pas de méthodologie générique de construction. Aucun standard n'a pour l'instant vu le jour dans ce domaine. Des groupes y travaillent, notamment au sein de la DARPA et de l'IETF.
- Les mises à jour de profils, de signatures d'attaques ou de façon de spécifier des règles sont généralement difficiles. De plus, les systèmes de détection d'intrusions

demandent de plus en plus de compétence à celui qui administre le système de sécurité.

- Les systèmes de détection sont généralement écrits pour un seul environnement et ne s'adapte pas au système surveillé alors que les systèmes d'informations sont, la plupart du temps, hétérogènes et utilisés de plusieurs façons différentes.
- Aucune donnée n'a été pour l'instant publiée pour quantifier la performance d'un système de détection d'intrusions. De plus, pour tester ces systèmes, les attaques sont de plus en plus difficiles à simuler.

7. Méthodes de classification et d'IA pour la détection d'intrusions

Plusieurs algorithmes de classification et d'IA, peuvent être utiles dans la détection d'intrusions. Ces algorithmes génèrent des classificateurs, sous forme d'arbre de décision, de règles ou de réseau de neurones, etc. Une application dans la détection d'intrusions serait d'appliquer ces algorithmes à une quantité suffisante de données d'audit normales ou anormales, pour générer un classificateur capable d'étiqueter comme appartenant à la catégorie normale ou anormale de nouvelles données d'audit. Parmi ces techniques de classification, nous citons :

7.1. réseaux bayésiens [49]

Les réseaux bayésiens sont des outils de raisonnement avec des informations incertaines dans le cadre de la théorie des probabilités. Les réseaux bayésiens, utilisent des graphes acycliques dirigés pour la représentation des relations causales et des probabilités conditionnelles (de chaque nœud dans le contexte de ses parents), pour exprimer l'incertitude sur ces relations.

Valdes **[50]** a proposé une nouvelle approche hybride, pour la détection d'intrusions se basant sur les réseaux bayésiens, en utilisant une forme simplifiée des réseaux bayésiens, appelée réseaux bayésiens naïfs, composée de deux niveaux: une racine qui représente la nature de la session (normal et les différents types d'attaques), et plusieurs nœuds enfants, chacun d'entre eux correspond à un attribut de la connexion.

Les réseaux bayésiens naïfs ont plusieurs avantages dus, en particulier, à leur construction qui est très simple. Par ailleurs, l'inférence (classification) est assurée

de façon linéaire (alors que l'inférence dans les réseaux bayésiens qui ont une structure générale est connue comme un problème NP complet. En plus, la construction des réseaux bayésiens naïfs est incrémentale, dans le sens qu'elle peut être facilement mise à jour (notamment, il est toujours possible de prendre en considération de nouvelles classes). Cependant, les réseaux bayésiens naïfs travaillent sous une hypothèse d'indépendance très forte entre les attributs dans le contexte de la nature de la session. Une telle hypothèse n'est pas toujours valable et peut entacher les résultats.

7.2. Arbre de décision [51] [52]

Les arbres de décision, représentent l'une des techniques les plus connues et les plus utilisées en classification. Leur succès est notamment dû à leur aptitude à traiter des problèmes complexes de classification. En effet, ils offrent une représentation facile à comprendre et à interpréter, ainsi qu'une capacité à produire des règles logiques de classification.

Un arbre de décision est composé de:

- ***Nœuds de décision :*** contenant chacun un test sur un attribut.
- ***Branches*** : correspondant généralement à l'une des valeurs possibles de l'attribut sélectionné.
- ***Nœuds feuilles :*** comprenant les objets qui appartiennent à la même classe.

L'utilisation des arbres de décision dans les problèmes de classification se fait en deux principales étapes:

1. Etape de construction de l'arbre : Cette étape se base sur un ensemble d'apprentissage, et consiste à sélectionner, pour chaque nœud de décision, l'attribut test 'approprié'. Elle consiste aussi à définir la classe libellant chaque feuille.

2. Etape de classification : Pour trouver la classe de l'objet, il suffit de suivre le chemin en partant de la racine de l'arbre de décision jusqu'aux feuilles en effectuant les différents tests à chaque nœud selon les valeurs des attributs de l'individu à classer.

Plusieurs IDS ont été proposés à base des arbres de décision, en utilisant la base KDD dans l'étape de construction de l'arbre.

Benfarhat **[49]** a proposé des IDS comportementaux à base des réseaux bayésiens naïfs et des arbres de décision. L'étude expérimentale est effectuée sur la base KDD.

7.3. Réseaux de neurones [53]

Les réseaux neuronaux sont utilisés pour leur rapidité de traitement et leur relative résistance aux informations incomplètes ou déformées. Ils sont utilisés de deux manières différentes. Ils sont d'abord utilisés comme filtres pour filtrer et sélectionner les parties suspectes dans les données d'audit. Celles-ci sont ensuite analysées par un système expert.

On peut ainsi réduire les fausses alarmes, et on peut même augmenter la sensibilité du système expert car il ne travaille que sur des données suspectes. Puis ils sont utilisés de façon à prendre seuls la décision de classer une séquence d'événements comme malveillante.

8. Les systèmes de détection d'intrusions actuels [54]

Le premier système de détection d'intrusions a été proposé en 1980 par *James ANDERSON [57]*. Il en existe maintenant beaucoup d'autres, commerciaux ou non. La majorité de ses systèmes se basent sur les deux approches, comportementale et par scénarios.

Stefan AXELSSON donne un modèle d'architecture de base pour un système de détection d'intrusions : un module s'occupe de la collecte d'informations d'audit, ces données étant stockées quelque part, un module de traitement des données qui interagit avec ces données de l'audit et les données en cours de traitement, ainsi qu'avec les données de référence (signatures, profils) et de configuration entrées par l'administrateur du système de sécurité. En cas de détection, le module de traitement remonte une alarme vers l'administrateur du système de sécurité ou vers un module. Une réponse sera ensuite apportée sur le système surveillé par l'entité alertée. Les imperfections de ce type de systèmes monolithiques et même des systèmes de détection d'intrusions en général sont à prendre en compte. *Stefano MARTINO* [58] souligne que si un certain nombre de techniques ont été développées jusque-là, pour les systèmes de détection d'intrusions, la plupart analysent des évènements au niveau local et se contentent de remonter une alarme sans agir. Ils détectent de plus, les activités dangereuses d'un utilisateur sans se préoccuper du code dangereux.

9. Imperfection des systèmes de détections d'intrusions [55]

Les systèmes de détections d'intrusions de nos jours présentent quelques imperfections, dans la plupart des cas, ces systèmes sont faits d'un seul bloc ou module qui se charge de toute l'analyse. Ces systèmes monolithiques exigent beaucoup de données d'audit, de ce fait ils utilisent beaucoup de ressources de la machine surveillée. L'aspect monolithique pose également des problèmes de mise à jour et constitue un point d'attaque unique pour ceux qui veulent s'introduire dans le système d'informations. D'autres imperfections plus générales sont relevables dans les systèmes de détection d'intrusions actuels :

- Même en implémentant les deux types d'approches, certaines attaques sont indécelables et les systèmes de détection sont eux-mêmes attaquables. Les approches comportementale et par scénarios ont elle-même leurs limites.
- Les groupes de travail sur ce sujet sont relativement fermés et il n'y a pas de méthodologie générique de construction. Aucun standard n'a pour l'instant vu le jour dans ce domaine. Des groupes y travaillent, notamment au sein de la DARPA et de l'IETF.
- Les mises à jour de profils, de signatures d'attaques ou de façon de spécifier des règles sont généralement difficiles. De plus, les systèmes de détection d'intrusions demandent de plus en plus de compétence à celui qui administre le système de sécurité.
- Les systèmes de détection, sont généralement écrits pour un seul environnement et ne s'adapte pas au système surveillé, alors que les systèmes d'informations sont la plupart du temps, hétérogènes et utilisés de plusieurs façons différentes.
- Aucune donnée n'a été pour l'instant publiée, pour quantifier la performance d'un système de détection d'intrusions. De plus, pour tester ces systèmes, les attaques sont de plus en plus difficiles à simuler.

Conclusion :

Nous avons présenté tout au long de ce chapitre, une étude des systèmes de détection d'intrusions. Afin de remplir ses objectifs, diverses méthodes de détection d'intrusions ont été proposées, elles sont basées principalement sur deux approches : l'approche comportementale et l'approche par scénarios. Parmi ces méthodes nous soulignons la classification. Plusieurs travaux dans le cadre de la détection ont été menés sur les algorithmes de classifications. Cependant, ces algorithmes possèdent des limites qui peuvent entacher la détection d'intrusions : un arbre de décision présente un très gros défaut dans le cas où des instances de l'ensemble de test ne satisfont aucune règle de la base d'apprentissage.

CHAPITRE 4

Les protocoles de routage dans les réseaux mobiles ad-hoc

Introduction :

Ces dernières années plusieurs protocoles de routage pour les réseaux ad hoc ont étés développés, ces protocoles essayent de maximiser les performances en minimisant le délai de livraison des paquets, l'utilisation de la bande passante et la consommation d'énergie. Dans ce chapitre nous allons présenter une classification des protocoles de rouage existants et présenter en détail le fonctionnement de quelques protocoles toute en indiquant leurs avantages et leurs inconvénients.

1. Classification des protocoles de routage Ad Hoc [59]

Selon la manière de création et de maintenance des routes on peut les classer les protocoles de routage pour les réseaux ad hoc comme suit : protocoles Table-driven (pro-actifs), protocoles On-demand (réactifs) et protocoles hybrides.

1.1 Les protocoles Table-driven (pro-actifs)

Le principe des protocoles Table-driven est de maintenir à jour des tables de routage qui Indiquent les routes vers chaque destination du réseau. Les protocoles de routage Table-driven sont basées sur deux méthodes utilisées dans les réseaux filaires, la méthode État de Lien (Link State) et la méthode Vecteur de Distance (Distance Vector).

La méthode Link State : Dans la méthode Link State chaque nœud diffuse périodiquement (par inondation) l'état des liens avec ses voisins a tous les nœuds du réseau, chaque nœud maintien alors une vue globale de la topologie du réseau ce qui lui permet de calculer les routes pour atteindre chaque destination. On inonde aussi le réseau quand il y a un changement dans l'état des liens. Cette méthode permet de trouver rapidement des alternatives pour transmettre les paquets en cas de coupure d'une route, on peut aussi utiliser simultanément plusieurs routes pour atteindre la même destination. Le problème avec cette méthode est que la quantité d'informations à stocker et diffuser peut devenir considérable si le réseau contient un grand nombre de nœuds.

La méthode Distance Vector : Dans la méthode Distance Vector chaque nœud transmet à ses voisins la distance (nombre de nœuds) qui le sépare de chaque destination dans le réseau et le nœud voisin à utiliser pour atteindre cette destination. En se basant sur les informations reçues depuis tous ses voisins, chaque nœud calcul le chemin le plus court vers n'importe quelle destination dans le réseau. Si la distance séparant deux nœuds change on répète le processus de calcul. La méthode Distance Vector évite l'inondation, mais elle est moins précise que la méthode Link State il est aussi difficile de trouver des routes alternatives en cas de coupure d'une route.

Les liens entre les nœuds dans les réseaux ad hoc changent rapidement. Les deux méthodes précédentes vont engendrer énormément de paquets de contrôle (inondation des états des liens, et transmission des vecteurs de distance) ce qui les rend inadaptés pour les réseaux ad hoc.

Les protocoles Table-driven calculent les routes à l'avance ils disposent donc des routes immédiatement vers les destinations du réseau. Le problème avec ces protocoles c'est qu'ils chargent le réseau avec les paquets de mise à jour des tables de routage même si le réseau n'est pas utilisé. Parmi les protocoles Table-driven les plus connus on peut citer : DSDV, GSR et FSR.

1.2 Les protocoles on-demand (réactifs)

A l'opposé des protocoles Table-driven les protocoles On-demand créent et maintiennent les routes selon les besoins, si un nœud veut envoyer un paquet à une destination à laquelle il ne

connaît aucune route, il lance un procédure de découverte de route globale qui va inonder le réseau avec un paquet de requêtes et lui fournir les informations nécessaires pour atteindre cette destination. Si la route devient invalide une autre procédure de découverte de route est lancée.

Les protocoles On-demand réduise la charge des paquets de contrôle comparés aux protocoles Table-driven, surtout si le réseau est très dynamique. Le problème avec ces protocoles c'est qu'ils ont un délai initial avant de commencer la transmission des paquets provoquer par la procédure de découverte de route, aussi la redécouverte de route en cas de coupure génère une charge supplémentaire. Parmi les protocoles On-demand on peut citer :
DSR et AODV.

1.3 Les protocoles Hybrides

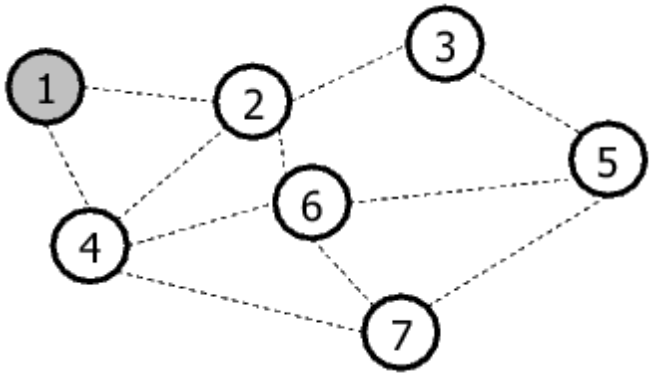
Les protocoles hybrides essaient de combiner les deux approches précédentes pour bénéficier de leurs avantages, ils utilisent un protocole Table-driven, pour connaître les voisins les plus proches, dans le but de réduire le délai et un protocole On-demand au- delà de cette zone prédéfinie dans le but de réduire la charge des paquets de contrôle.

Les protocoles hybrides cumulent aussi les inconvénients des protocoles Table-driven et des protocoles On-demand à savoir les paquets de contrôle périodique, et le délai de découverte de route. Parmi les protocoles hybrides on peut citer le protocole CBRP.

2 Quelques protocoles de routage pour les réseaux ad hoc

2.1 Le protocole DSDV [60]

DSDV (*Destination-Sequenced Distance-Vector*) est un protocole de routage Table-driven.
Chaque nœud dans DSDV garde une table de routage qui donne pour chaque destination accessible dans le réseau :
– Le nœud voisin à utiliser pour atteindre cette destination,
– Un numéro de séquence qui est envoyé par le nœud destinataire et qui permet de distinguer les nouvelles routes des anciennes,
– Le nombre de sauts (nœuds intermédiaires) pour atteindre cette destination.
Périodiquement chaque nœud dans le réseau diffuse par inondation un paquet de mise à jour des tables de routage qui inclue les destinations accessibles et le nombre de sauts exigés pour atteindre chaque destination avec le numéro de séquence lié à chaque route. Des paquets de mise à jour sont aussi diffusés immédiatement s'il y a un changement dans la topologie du réseau afin de propager les informations de routage aussi rapidement que possible.



La figure (4.1) illustre la topologie d'un réseau ad hoc a un instant donné et la table de routage correspondant au nœud (1) dans le protocole DSDV.

A la réception d'un paquet de mise à jour, chaque nœud le compare avec les informations existantes dans sa table de routage. Les routes les plus récentes (qui ont le plus grand numéro de séquence) avec la distance la plus courte sont gardées, les autres sont simplement ignorées.

Destination	*Prochain saut*	*Distance*	*Numéro de séquence*
4	4	1	10
3	2	2	32
6	2	2	88
5	4	3	19
2	2	1	12
7	4	2	57

Tableau 4.1 : A gauche topologie du réseau, à droite table de routage du nœud (1) dans le Protocole DSDV.

DSDV fourni à tout moment des routes valables vers toutes les destinations du réseau, Mais l'inondation des paquets de mise à jour (périodique et en cas de changement de topologie) cause une charge de contrôle importante au réseau.

2.2. Le protocole GSR [59]

Dans le protocole Table-driven GSR (*Global State Routing*) chaque nœud maintient une table de la topologie qui l'informe sur la topologie globale du réseau et lui permet de calculer les routes pour atteindre chaque destination. GSR utilise la méthode Link State des réseaux filaires et l'améliore en supprimant le mécanisme d'inondation des paquets de contrôle.

Un nœud dans GSR maintient :

– Une liste de voisins,
– Une table de topologie qui contient les informations sur les liens du réseau,
– Une table des nœuds suivants qui indiquent le nœud à utiliser pour atteindre chaque destination,
– Une table de distance qui contient la plus courte distance pour chaque destination.

Comme dans la méthode Link State chaque nœud dans GSR construit sa table de topologie basé sur les informations de liens reçus, et l'utilise pour calculer les distances minimales qui le séparent des autres nœuds du réseau. Dans GSR la table de topologie entière de chaque nœud est échangé périodiquement uniquement avec les voisins au lieu de la diffusé par inondation dans tout le réseau.

GSR réduit la charge des paquets de contrôle en évitant l'inondation et assure plus de précision, concernant les données de routage. Le problème de GSR est la taille de ses paquets de mise à jour (Table de topologie) qui peut devenir considérable si le réseau contient un grand nombre de nœuds.

2.3 Le protocole FSR [61]

Une caractéristique observée dans l'œil de poisson (Fish eye) est qu'il distingue les Choses en détail au centre, et que sa précision se dégrade en s'éloignant du point central.

FSR (*Fisheye State Routing*) est un protocole Table-diriven, il minimise la charge des paquets de mise à jour des tables de routage du protocole GSR en utilisant la technique de l'œil de poisson. Les paquets de mise à jour dans FSR, ne contiennent pas l'information sur tous les nœuds du réseau, il échange les informations sur les nœuds les plus proches plus fréquemment qu'il le fait sur les nœuds les plus lointains, il réduit ainsi la taille des paquets de mise à jour.

Un nœud dans FSR a donc des informations précises sur les nœuds proches (figure 4.2 :) Zone 1), la précision des informations diminue quand la distance augmente (Figure 4.2: Zone 2 et 3).

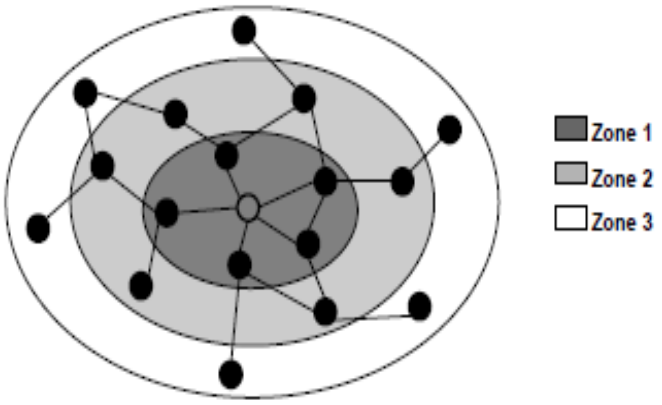


Figure 4.2 : Précision des informations d'un nœud dans FSR
en utilisant la technique œil de poisson.

Malgré qu'un nœud dans le réseau n'ait pas des informations précises sur les nœuds éloignés (surtout si le réseau est très dynamique), les paquets peuvent être transmis correctement car l'information sur les routes devient de plus en plus précise quand les paquets se rapprochent de leurs destinations.

2.4 Le protocole AODV [62]

Le protocole AODV (*Ad hoc On-demand Distance Vector*) appartient à la famille des protocoles On-Demand, il est basé sur deux mécanismes, la *découverte de route* et les 26 maintenances *de route*. La découverte de route permet de trouver une route pour atteindre une destination et cela en inondant un paquet de requête dans tout le réseau. La maintenance de route permet de détecter et signaler les coupures de routes provoquées éventuellement par la mobilité des nœuds. AODV n'utilise pas des mises à jour périodiques, les routes sont découvertes et maintenues selon les besoins.

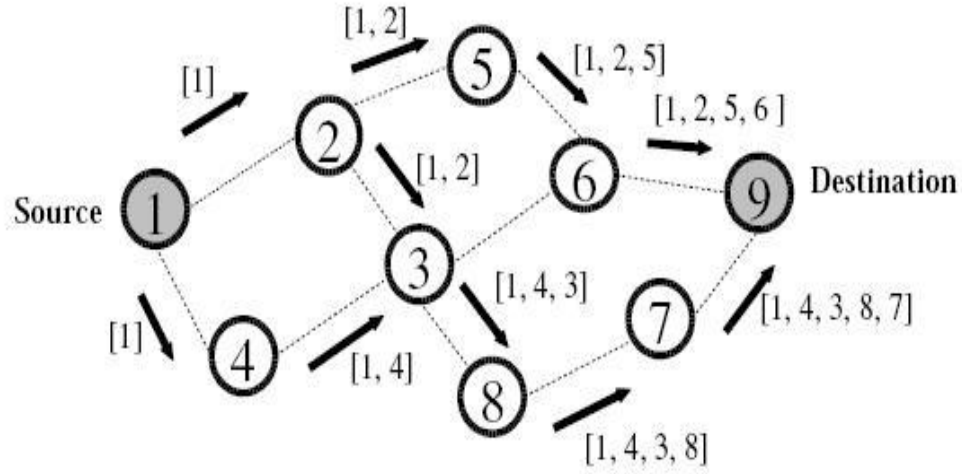
Chaque nœud intermédiaire qui se trouve dans la route entre un nœud source et un nœud destination doit garder une table de routage qui contient :

– L'adresse de la destination.
– Le nœud suivant à utiliser pour atteindre la destination.
– La distance en nombre de nœud: C'est le nombre de nœud nécessaire pour atteindre la destination.
– Le numéro de séquence destination: Il permet de distinguer les nouvelles routes des anciennes.
– Le temps d'expiration de l'entrée de la table: C'est le temps au bout duquel l'entrée est valide.

AODV utilise trois types de messages pour créer et maintenir les routes, le RREQ (Route Request) pour demander une route, le RREP (Route Reply) pour répondre à une requête de demande de route, et le RERR (Route Error) pour signaler une coupure de route.

2.5 Le protocole DSR [63]

DSR (*Dynamic Source Routing*) est un protocole On-demand semblable au protocole AODV, il utilise une technique appelé « Source Routing » dans laquelle l'émetteur (la source) indique la route complète par laquelle un paquet doit passer pour atteindre sa destination, cette route est insérée dans l'entête du paquet. Les nœuds intermédiaires entre le nœud source et le nœud destination n'ont pas besoins de maintenir à jour les informations sur la route traversée puisque la route complète est insérée dans l'entête du paquet.
Si un nœud dans DSR veut communiquer avec une destination à laquelle il ne possède pas de route, il inonde le réseau avec un paquet de requête (RREQ) similaire à celui d'AODV. Chaque nœud qui reçoit la requête et qui ne possède pas de route à la destination demandée insère son adresse dans le paquet RREQ et le diffuse à ses voisins. La réponse à la requête (RREP) est retournée par la destination ou par un autre nœud qui possède une route à la destination (figure 4.3).
(Introduction à la requête)



(Renvoie de la réponse)
Figure 4.3 : Découverte de route dans DSR.

Les routes dans AODV sont construites en traversant la route inverse ver la source (de la destination à la source), dans DSR les routes sont construites quand la requête traverse le réseau vers la destination (de la source à la destination).

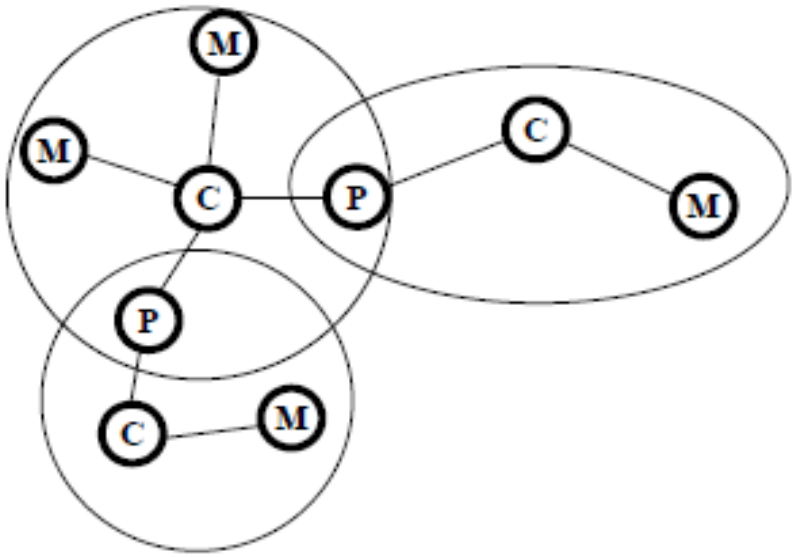
Si un nœud reçoit un paquet de données, et le lien à utiliser pour retransmettre ce paquet est coupé (coupure de route) le nœud envoie un message d'erreur de route (RERR) semblable à celui de AODV au nœud source. Le nœud source va lancer une autre requête de découverte de route pour atteindre la destination.

Le protocole DSR comme le protocole AODV utilise l'inondation pour découvrir les routes ce qui généré un trafic de contrôle énorme quand le réseau est très utilisé. Aussi la taille des paquets de données dans DSR devient très grande quand le nombre de nœuds dans réseau est grand, puisque les paquets doivent porter les adresses de chaque nœud dans la route traversée. DSR a aussi un délai avant de commencer la transmission des paquets provoqués par la procédure de découverte de route.

2.6 Le protocole CBRP [64]

Le protocole de routage CBRP (*Cluster Based Routing Protocol*) est un protocole Hybride. Il utilise deux niveaux hiérarchiques pour effectuer le routage. Les nœuds dans CBRP sont rassemblés en groupes appelés « Clusters », avec un chef au centre de chaque groupe appelé « Clusterhead ». Les clusters sont reliés entre eux par des nœuds passerelles qui se trouvent à l'extrémité des clusters. CBRP utilise un protocole Table-driven pour maintenir les membres du cluster, et un protocole On-demand pour découvrir les nœuds en dehors du cluster.

Le diamètre d'un cluster dans CBRP est de deux sauts, un exemple d'assemblage en cluster est illustré dans la figure suivante :



C : Chef M : Membre P : Passerelle.

Figure 4.4 : exemple d'assemblage en clusters dans CBRP.

Le principe de formation des clusters dans CBRP est le suivant :

1. Les nœuds s'échangent des messages «Hellos» pour connaître leurs voisinages.
2. Les nœuds vont élire le nœud avec le plus petit identifiant chef de cluster.
3. Un nœud qui n'a pas de chef de cluster comme voisin devient chef de cluster.
4. Le chef de cluster prend tous ses voisins comme membres de son cluster.

Pour éviter des changements fréquents dans les structures des clusters, les chefs sont maintenus le plus longtemps possible. Un chef de cluster n'est pas changé même si un nouveau membre (non chef) de son cluster a un identifiant plus petit. Si par la suite d'un changement de topologie deux chefs de clusters ont un lien direct entre eux, il faut reconstruire les clusters et le nœud possédant l'identifiant le plus faible est élu comme chef.

Chaque nœud dans CBRP maintient une table de voisinage qui indique le statut (membre, ou chef) de ses voisins. Un chef de cluster maintient aussi une liste des nœuds membres de son cluster, et une liste des chefs de clusters voisins avec les adresses des nœuds passerelles pour les atteindre.

Quand un nœud source dans CBRP veut envoyer un paquet de données à un nœud destination, il transmet au chef de son cluster un paquet de requête qui contient l'adresse de la destination désirée, le chef de cluster va insérer son adresse dans le paquet et le diffuse aux chefs

de clusters voisins. Quand un chef de cluster reçoit un paquet de requête de route, il vérifie dans sa liste des nœuds membre si le nœud destination appartient à son cluster. Si c'est le cas il répond au nœud source en inversant la route enregistré dans le paquet de requête, sinon il enregistre son adresse dans le paquet de requête et le diffuse aux chefs de clusters voisins. Si le nœud source ne reçoit pas de réponse au cours d'une certaine période de temps, il envoie une autre requête à son chef.

CBR utilise la technique « Source Routing ». Comme dans DSR la route complète vers la destination est insérée dans l'entête du paquet. Le routage par source permet à CBRP de faire des raccourcissements de routes s'il y a un changement de topologie. Quand un nœud reçoit un paquet il essaye de l'envoyer au nœud voisin le plus lointain dans la route, ce qui raccourci la route utilisée. Si un nœud dans CBRP reçoit un paquet de données et le lien ver le nœud suivant est coupé (coupure de route) il envoie un message d'erreur à la source et essaye de réparer la route localement. Il vérifie s'il peut atteindre le nœud suivant dans la route, ou le nœud qui vient après le nœud suivant par un autre voisin, si c'est le cas le paquet est envoyé sur la route réparée.

CBRP diffuse les requêtes de route seulement au chef de cluster. Il réduit ainsi la charge provoquée par l'inondation des requêtes de route. Mais la formation et la maintenance des clusters engendrent une charge supplémentaire au réseau. Aussi les chefs de clusters sont responsables de l'acheminement des données ce qui peut provoquer des goulets d'étranglement.

2.7 Le protocole ARA [65]

ARA *(Ant-Colony-Based Routing Algorithm)* est un protocole On-demand. Il se base sur les techniques d'optimisation par colonie de fourmis (ACO). ARA utilise des agents fourmis pour découvrir et maintenir les routes entre les nœuds du réseau.

L'idée de base des algorithmes d'optimisation par colonie de fourmis est inspirée du comportement des fourmis réel quand elles recherchent la nourriture. Les fourmis réussissent toujours à trouver le chemin le plus court pour atteindre la nourriture.

La figure 4.5 illustre un scénario avec deux chemins possibles pour atteindre la nourriture. Quand les premières fourmis arrivent à l'intersection des deux chemins ils vont choisir aléatoirement le chemin à prendre. Les fourmis en parcourant le chemin vont déposer une matière appelée phéromone, cette matière indique aux autres fourmis le chemin à utiliser.

La phéromone s'évapore avec le temps. La quantité de phéromone déposée sur le deuxième chemin va devenir plus importante que celle déposée sur le premier chemin puisque les fourmis parcourent le deuxième chemin plus vite. Avec le temps toutes les fourmis vont utiliser le deuxième chemin qui est le chemin le plus court.

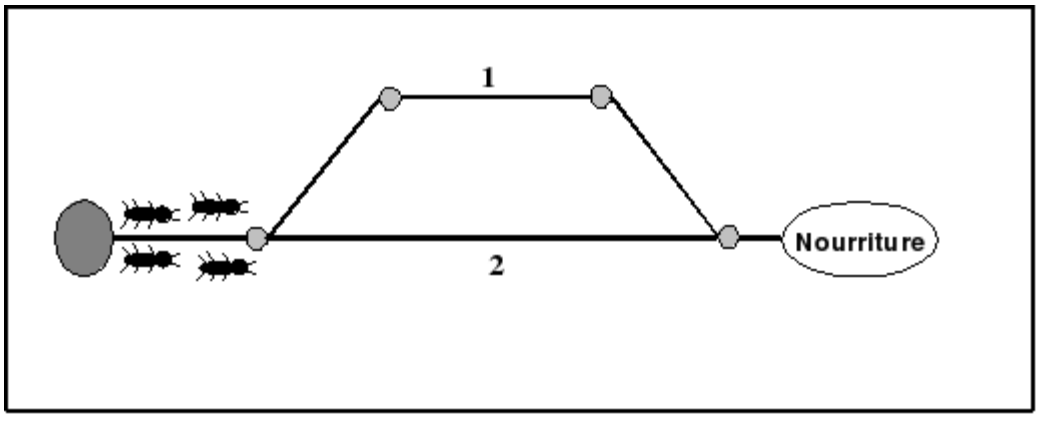


Figure 4.5 : Recherche de nourriture par une colonie de fourmis

Ce comportement est utilisé dans ARA pour trouver la route la plus courte entre un nœud source et un nœud destination dans le réseau**.**

ARA se compose de trois phases la découverte de routes, la maintenance de routes et le traitement de la coupure de routes.

2.8 Le protocole Ant-AODV

Le protocole Ant-AODV est une combinaison du protocole AODV et des capacités de découverte de routes des agents fourmis. Chaque nœud dans ce protocole maintient une table de routage qui indique pour chaque destination le nœud voisin à utiliser pour atteindre la destination. Cette table est maintenue par les agents fourmis qui se déplacent dans le réseau et découvre les routes entre les nœuds (figure 4.6). Les agents fourmis sont de simples paquets avec un numéro de séquence unique qui permet de les distinguer.

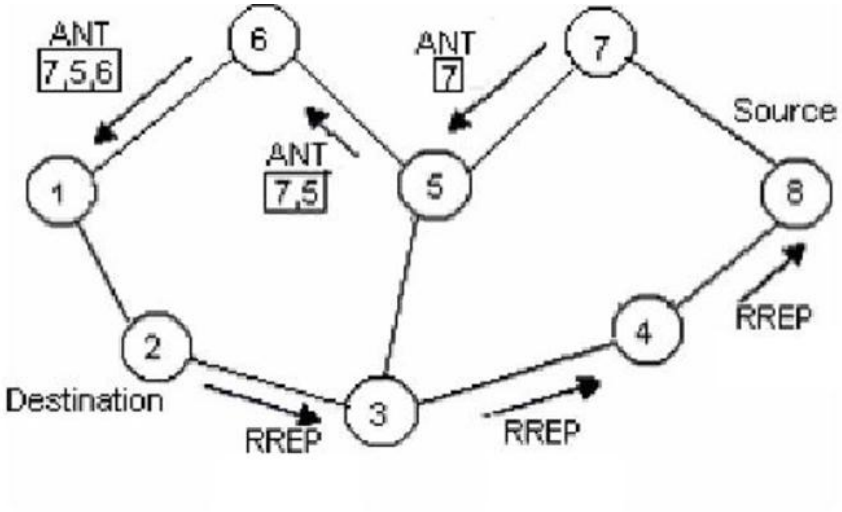


Figure 4.6 : Découverte de route dans Ant-AODV.

Si un nœud veut envoyer des paquets a une destination pour laquelle ils n'a pas de routes récentes dans sa table de routage, il lance une découverte de route en utilisant le protocole AODV. Le nœud inonde le réseau avec la requête de demande route RREQ qui contient son adresse et l'adresse de la destination. La réponse à la requête (RREP) est retournée par la destination ou par un autre nœud qui possède une route à la destination.

L'utilisation des agents fourmis avec AODV permet de fournir plus de routes aux nœuds ce qui par la suite réduit le coup de la découverte de routes, même si un nœud lance une requête pour une destination pour laquelle il n'a pas de routes récentes la probabilité de recevoir une réponse rapidement de ses voisins est grande ce qui a pour résultat la réduction du délai de découverte de routes.

3. Tableaux récapitulatifs [59]

Nous décrivons dans les deux tableaux suivants les différentes classes de protocoles de routage pour les réseaux ad hoc ainsi que les protocoles de routage présentés dans ce chapitre.

Classes	*Caractéristiques*	*Avantages*	*Inconvénients*
Table-Driven	-Calculer les routes à l'avance.	-Transmission immédiate des données.	-Utiliser beaucoup de paquets de contrôles.
On-Demande	-Calculer les routes à la demande.	-Utiliser moins de paquets de contrôles.	-délais initial avant de commencer la transmission des don-nées
Hybride	-Combinaison des deux approches précédentes.	-Bénéficier des avant Ages des deux approches précédentes.	-cumuler les inconvénients des deux approches précédentes.

Tableau 4.2 : Les classes des protocoles de routage pour les réseaux ad hoc.

Protocoles	***Classe***	***Avantages***	***Inconvénients***
DSDV	Table-Driven	- Fournit à tout moment des routes valables ver toutes les destinations du réseau.	- L'inondation des paquets de mise à jour cause une charge de contrôle importante au réseau.
GSR	Table-Driven	- Évite l'inondation en transmettant les paquets de mise à jour seulement aux voisins.	- Taille considérable des paquets de mise à jour.
FSR	Table-Driven	- Minimise la charge des paquets de mise à jour de la table de routages du protocole GSR en utilisant la technique de l'œil de poisson.	- Taille considérable des paquets de mise à jour.
AODV	On-Demand	- Découvre les routes à la demande en inondant le réseau avec un paquet de requêtes.	- Délai initial avant de commencer la transmission des données.
Protocoles	***Classe***	***Avantages***	***Inconvénients***

DSR	On-Demand	- Découvre les routes à la demande en inondant le réseau avec un paquet de requêtes. - Les paquets de données peuvent être redirigés pendant leur transmission.	- Délai initial avant de commencer la transmission des données. - La taille des paquets de données très grande quand le nombre de nœud dans réseau est grand.
CBRP	Hybride	- Diffuse les requête de route seulement aux chefs de cluster ce qui permet de réduire la charge provoqué par l'inondation des requête de route.	- La formation et la maintenance des clusters engendrent une charge supplémentaire.
ARA	On-Demand	- Découvre les routes à la demande en utilisant les techniques d'optimisation par colonie de fourmis. - Permet de trouver rapidement des alternatives en cas des coupures d'une route.	- La phase de découverte de route engendre un trafique considérable.
Ant-AODV	Table-Driven	- Réduit le coup de la découverte de routes en combinant les fourmis et AODV.	- Les fourmis peuvent engendrer un trafic considérable.

Tableau 4.3 : Les protocoles de routage pour les réseaux ad hoc.

Conclusion

Nous avons présenté dans ce chapitre une classification des protocoles de routage pour les réseaux ad hoc. Ces protocoles peuvent se classer en trois grandes catégories. Les protocoles Tables-Driven qui réduisent le délai de livraison de paquets mais utilisent beaucoup de paquets de contrôle, les protocoles On-Demend qui réduisent l'utilisation des paquets de contrôle mais qui ont un délai de livraison des paquets élevé, et les protocoles
Hybrides qui essayent de combiner les deux approches précédentes pour avoir de meilleures performances. Le problème est donc de trouvé un compromis entre le délai de livraison des paquets de données et l'utilisation des paquets de contrôle. Dans le chapitre suivant nous allons présenter la technologie agent ainsi que quelques protocoles de routage basé sur cette technologie.

CHAPITRE 5

Les IDS pour le protocole de routage AODV

INTRODUCTION

Les besoins excessifs de mobilité et de rapidité de déploiement ont mené à l'établissement du schéma de réseau ad hoc. Un réseau ad hoc appelé généralement MANET (Mobile Ad hoc Network) est un ensemble d'unités mobiles autonomes, formant un réseau temporaire sans aucune infrastructure fixe. Pour rester connecté, chaque entité joue le rôle d'un routeur et prend part de la responsabilité d'acheminement des données. Cette manipulation rend le réseau très vulnérable aux attaques.

Ce chapitre est organisé comme suit : Dans la section I on décrit le fonctionnement du protocole de routage réactif AODV, ensuite on aborde les principales vulnérabilités et attaques liées au routage ad hoc et enfin les solutions trouvées aux diverses attaques.

1. PROTOCOLE AODV

Ad hoc On demand Distance Vector (AODV) est un protocole de routage réactif ce qui signifie que les routes sont construites à la demande. Ce protocole est basé sur l'algorithme de routage Distance–Vector (DV) [66]. Il empreinte l'utilisation des numéros de séquence de DSDV afin de maintenir la consistance des informations de routage.

A cause de la mobilité des nœuds dans les réseaux ad hoc, les routes changent fréquemment ce qui fait que les routes maintenues par certains nœuds deviennent invalides. Les numéros de séquence permettent d'utiliser les routes les plus nouvelles ou autrement dit les plus fraiches [67].

Le protocole AODV définit deux types d'opérations : la découverte des routes et la maintenance des routes, en plus du routage nœud-par-nœud, le principe des numéros de séquence et l'échange périodique du DSDV. Il utilise trois types de paquets de routage à savoir : RREQ (Route REQuest), RREP (Route REPly), RERR (Route ERRor).

❖ Découverte des routes :

Avec le protocole AODV, chaque nœud doit maintenir une liste de ses voisins actifs. Cette liste est obtenue par un échange périodique des messages HELLO de chaque nœud avec ses voisins immédiats. Quand un nœud source S veut envoyer des données à un destinataire D et qu'aucune route vers cette destination n'est stockée dans la table de routage de la source, le nœud S initialise une procédure de découverte de routes.

La source S envoie à ses voisins une demande de route RREQ (*Route REQest*) qui contient l'adresse de S, l'identifiant de la requête, un compteur de séquence, l'adresse de D et le compteur de nombre de sauts avec une valeur initiale zéro. La source attendra une période RREP_WAIT_TIMEOUT, si une réponse est reçue alors l'opération de découverte de route est terminée, sinon elle rediffuse le RREQ et attend une période plus grande si aucune réponse n'est reçue, elle continuera la rediffusion du RREQ jusqu'à un nombre maximum de tentatives RREQ_RRTRIES (03 tentatives), si après RREQ_RETRIES tentatives d'établissement de route, il n'y a aucune réponse alors le processus est abandonné et un message d'erreur est signalé à l'application. Après une certaine période d'attente (10 s), l'application demande la route et par **conséquent** l'opération de découverte de route est initiée [68]. Chaque nœud qui reçoit le message RREQ recherche dans sa table de routage locale s'il existe une

route vers le nœud D sinon le nœud qui traite la requête RREQ incrémente le nombre de sauts et la diffuse à nouveau. Lorsque la requête atteint la destination D ou un nœud qui connaît une route vers la destination, une réponse RREP (*Route REPly*) est diffusée sur la même route de réception du RREQ (chemin inverse). La réponse RREP contient l'adresse source, l'adresse de destination, le nombre de sauts, un numéro de séquence de destination et la durée de vie du paquet. La réponse RREP passe par la route inverse vers le nœud source S. Ainsi chaque nœud, sur cette route, enregistre une entrée dans sa table de routage local vers le nœud destination avant de renvoyer le paquet. Une fois la source S reçoit le message, elle commence à envoyer les données vers D. [70]

La Figure ci-dessous donne un exemple de recherche de route dans un réseau avec AODV. Le nœud *Source* veut envoyer du trafic au nœud *Destination*. Il génère un paquet RREQ qu'il diffuse à ses voisins 2, 3 et 4. Ces trois nœuds à leur tour retransmettent le RREQ à leurs voisins. Ce paquet permet, au niveau de chaque nœud, de mettre à jour le saut suivant pour la route construite vers la *Source*. À chaque saut, le nœud calcule le nombre de sauts depuis la *Source*. Le RREQ arrive au niveau au nœud 8 qui voit qu'il est le nœud destination, il génère alors un paquet RREP qui fait le chemin inverse et informe le nœud source du chemin à prendre.

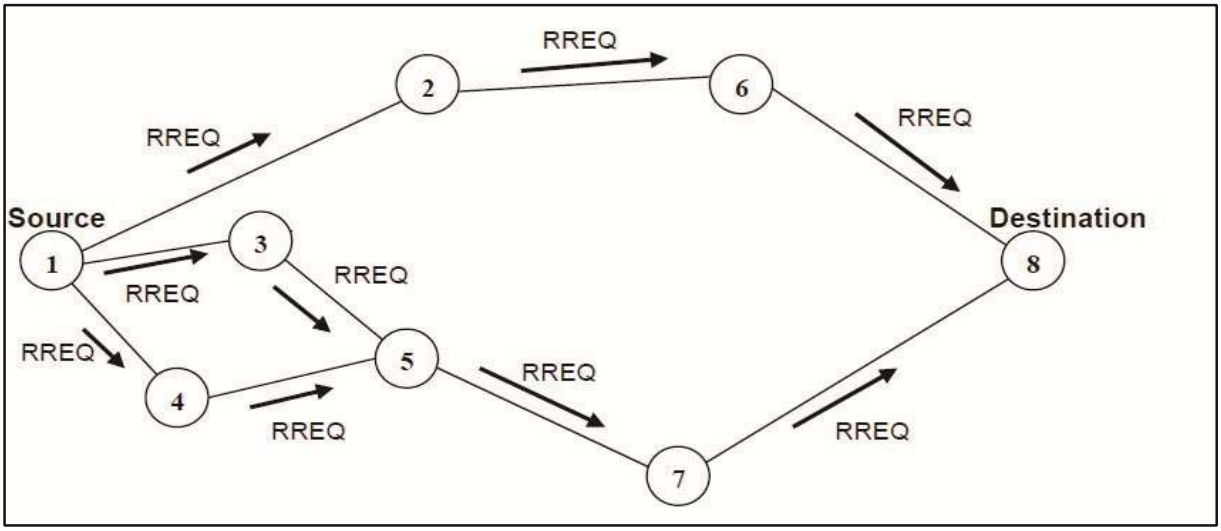


Figure 5.1: Une demande de route

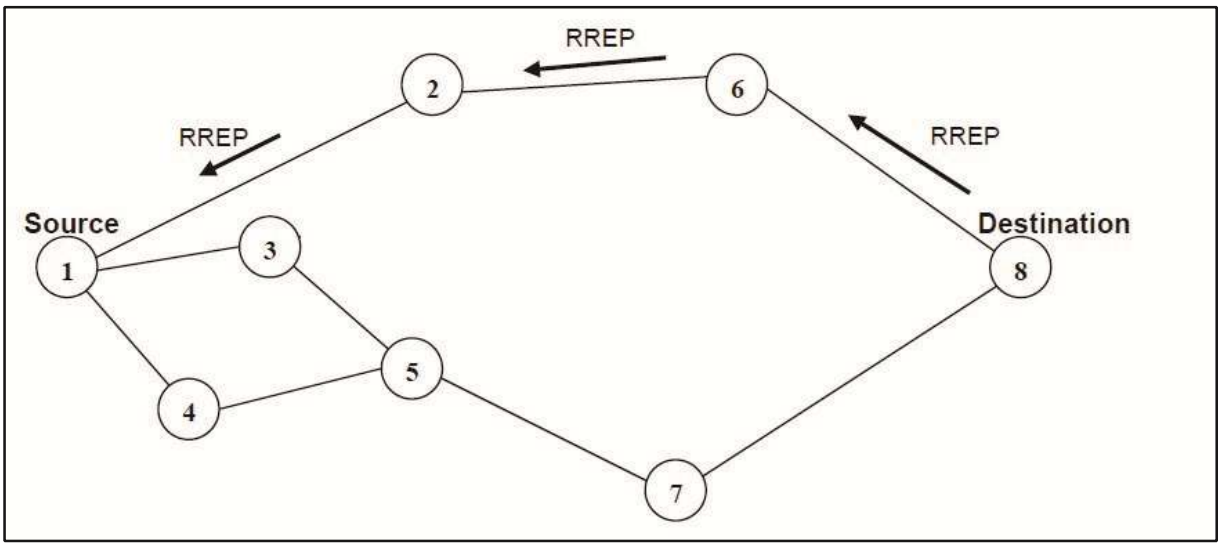
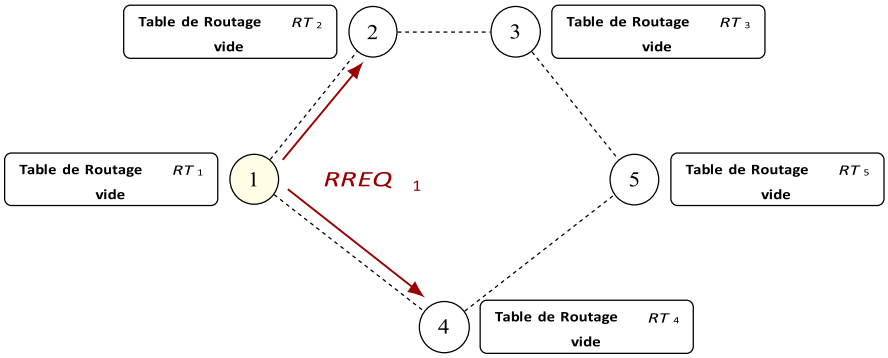


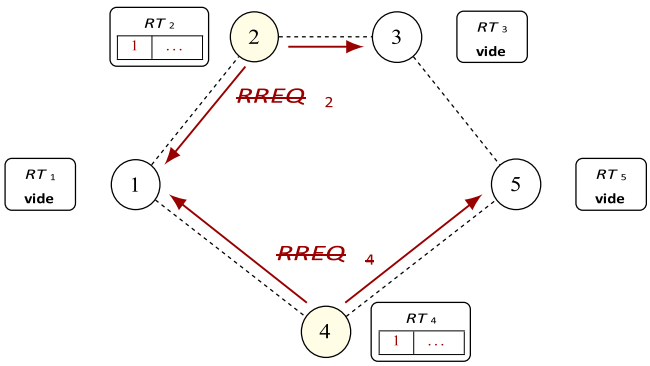
Figure 5.2: Réponse de route

- **Maintenances des routes : [69]**

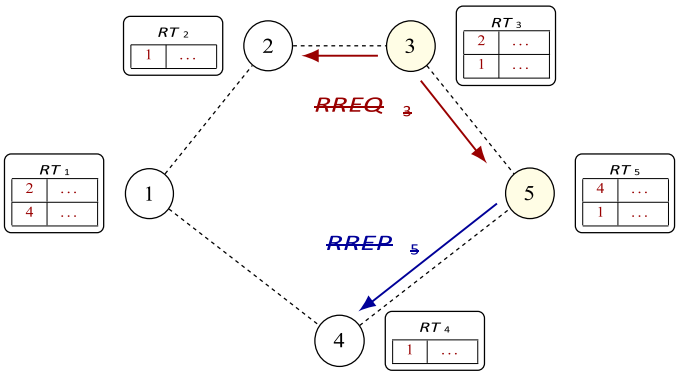
Afin de maintenir les routes, une transmission de messages HELLO est effectuée. Ces messages sont en fait des réponses de route (RREP) diffusés aux voisins avec un nombre de sauts égal à un. Si au bout d'un certain temps, aucun message n'est reçu d'un nœud voisin, le lien en question est considéré défaillant. Alors, un message d'erreur RERR (*Route ERRor*) se propage vers la source et tous les nœuds intermédiaires vont marquer la route comme invalide et au bout d'un certain temps, l'entrée correspondante est effacée de leur table de routage. Le message d'erreur RERR peut être diffusé ou envoyé en *unicast* en fonction du nombre de nœuds à avertir de la rupture de liaison détectée. Ainsi, s'il y en a un seul, le message est envoyé en *unicast* sinon, il est diffusé.



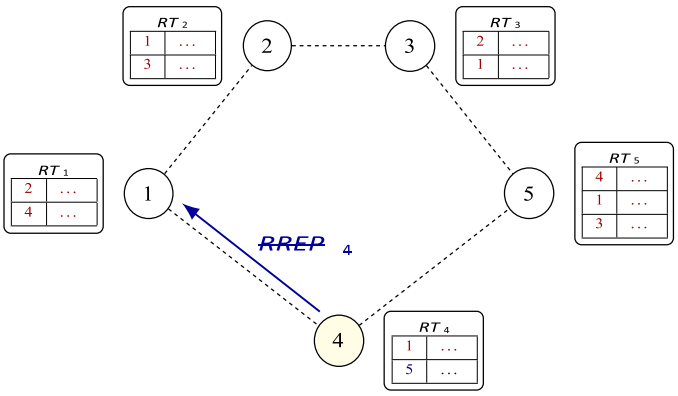
a. Le nœud 1 initialise une demande de route pour obtenir un chemin vers 5



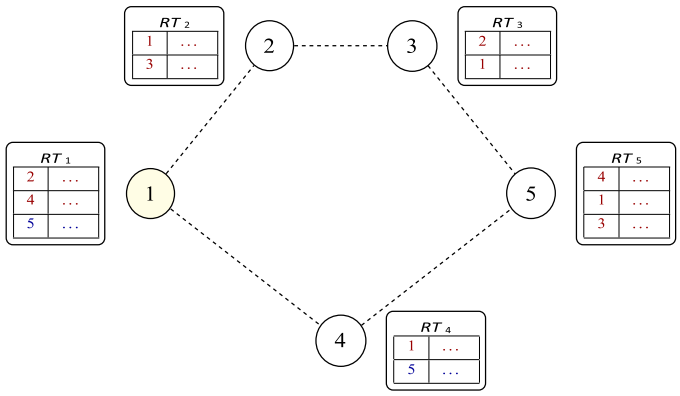
b. 2 et 4 retransmettent la demande de route



c. 5 initialise une réponse de route



d. 4 retransmet la réponse de route



e. Arrivée de la réponse de route à la source 1

FIGURE 5.3 – Exemple d'établissement de route entre 1 et 5 **[70]**

AODV à l'avantage de réduire le nombre de paquets de routage échangés étant donné que les routes sont créées à la demande et utilise le principe du numéro de séquence pour éviter les boucles de routage et garder la route la plus fraîche. Cependant, l'exécution du processus de création de route occasionne des délais importants avant la transmission de données

2. Propriétés d'AODV [71]

2.1 Les mérites d'AODV

Le protocole de routage AODV n'a pas besoin de système administratif central pour contrôler le processus de routage. Les protocoles réactifs comme AODV ont tendance à réduire le contrôle de la circulation des messages généraux au coût de l'augmentation de la latence à trouver de nouveaux itinéraires (routes).

AODV réagit assez rapidement aux changements topologiques dans le réseau et met à jour uniquement les nœuds affectés par ces changements

Les messages HELLO soutenant le maintien de routes sont gamme limitée, donc ils ne provoquent pas une surcharge sur le réseau.

Le protocole de routage AODV enregistre l'endroit de stockage aussi bien que l'énergie. Le nœud de destination répond une seule fois à la première demande et ignore le reste. La table de routage maintient au plus une entrée par destination.

Si un nœud doit choisir entre deux trajets, le trajet récent avec un numéro d'ordre de destination plus grand est toujours choisi. Si entrée de table de routage n'est pas utilisé récemment, l'entrée est expiré. Une route non valide est supprimée : les paquets d'erreur atteignent tous les nœuds en utilisant un lien raté sur sa route vers toute destination.

2.2 Les inconvénients d'AODV

Il est possible qu'une route valable soit expirée. La détermination d'un temps d'expiration raisonnable est difficile, parce que les nœuds sont mobiles et l'envoi des sources de taux peuvent différer largement et peut changer dynamiquement de nœud à nœud.

En outre, AODV peut recueillir qu'une quantité très limitée d'informations de routage, la route d'apprentissage est limitée uniquement à la source d'un routage de paquets transmis. Ceci provoque une inondation de découverte de route plus fréquemment, ce qui peut entraîner d'importantes surcharges réseau. Les inondations incontrôlées génèrent de nombreuses transmissions redondantes qui peuvent provoquer appelé broadcast storm (tempête d'émission)

La performance du protocole AODV sans n'importe quels nœuds se conduisant mal est pauvre dans de plus grands réseaux. La principale différence entre les petits et les grands réseaux est la longueur moyenne des chemins. Un long chemin est plus vulnérable aux ruptures de lien et requiert une charge de contrôle élevé pour son entretien.

En outre, comme la taille d'un réseau grandit, diverses mesures de performance commencent à diminuer en raison de l'augmentation du travail administratif, que l'on appelle la charge administrative.

AODV est vulnérable à toutes sortes d'attaques, car il repose sur l'hypothèse que tous les nœuds vont coopérer. Sans cette coopération, aucune route ne peut être établie et aucun paquet ne peut être transmis. Il existe deux principaux types de nœuds non coopératifs : malveillants et égoïstes. Les nœuds malveillants sont soit défectueux et ne peuvent pas suivre le protocole, ou sont intentionnellement malveillant et essaient d'attaquer le réseau.

L'égoïsme est la non coopération dans de certaines opérations de réseau, par exemple : suppression de paquets qui peuvent affecter les performances, mais peuvent économiser la batterie.

3 Vulnérabilités dans AODV [72]

Le protocole Ad-hoc on-demand distance vector est très efficace en tant que service de réseau, mais il a beaucoup de vulnérabilités, signifie que ce protocole peut facilement être attaqué. AODV n'est pas si sécurisé. AODV est conçu pour un réseau idéal signifie pour un réseau n'ayant aucun nœud malveillant. Pour un réseau n'ayant aucun nœud malveillant le protocole AODV est le plus efficace. Mais nous savons tous que rien n'est idéal signifie qu'il y a certains nœuds incommodes partout. Quelques nœuds gourmands sont aussi là dans le nœud qui attaque sur le réseau pour résoudre leur but. Dans AODV ce que nous pouvons faire pendant les messages RREQ ou les messages RREP est comme suit. Les types possibles d'attaques.

- Les numéros de séquence peuvent être modifiés.
- Le nombre de sauts peuvent être modifié. (principale attaque est la formation des boucles dans le réseau « Looping »).
- Modification des routes source (attaque Black hole, des informations erronées sur le chemin).
- Tunneling.
- Spoofing.

3.1. Attaques élémentaires portant sur les demandes de route : [70]

3.1.1. Suppression d'une demande de route :

Un nœud malhonnête pourrait simplement **effacer** la demande de route reçue. En appliquant ce genre de comportement à tout message RREQ reçu, l'attaquant ne participe pas au routage c'est comme s'il ne fait pas partie du réseau. Une autre variante serait d'effacer sélectivement des messages RREQ. Ce comportement peut être comparé à celui d'un nœud égoïste.

3.1.2 Modification d'une demande de route :

À la réception d'une demande de route, le nœud malhonnête **modifie** un ou plusieurs champs qu'il n'est pas supposé modifier avant de retransmettre le message. La modification peut aussi porter sur un champ qu'il a le droit de modifier, mais il ne respecte pas la spécification pour le faire.

Plusieurs champs impliquent des traitements différents lorsqu'ils sont modifiés. Par exemple,
Le champs identifiant de la RREQ associé à l'adresse de la source permet d'identifier de manière unique une demande de route et indique la fraicheur de la demande de route.
Puisqu'un nœud n'accepte que le première copie de RREQ, en augmentant cet identifiant, le nœud malhonnête peut garantir l'acceptation et le traitement de la RREQ modifiée par les autres nœuds.

3.1.3. Fabrication d'une demande de route :

Les attaques décrites précédemment (section 3.1.1 et 3.1.2) sont déclenchées par la réception d'une demande de route. En revanche, les attaques par **fabrication** peuvent être effectuées sans avoir reçu de messages RREQ. Le nœud malhonnête a besoin de collecter certaines informations, en écoutant le trafic par exemple, avant d'injecter le message fabriqué. Par conséquent l'exécution répétitive de ce type d'attaque peut provoquer l'inondation du réseau par des messages de routage inutiles. Le nœud malhonnête peut orienter le trafic vers une seule destination ou faire croire que le trafic part d'une seule source ou les choisir (source/destination) au hasard.

3.1.4. Rushing d'une demande de route :

Dans d'autres cas, le nœud malhonnête peut utiliser la technique du ***rushing*** qui consiste à diminuer le temps de traitement des messages RREQ et les retransmettre plus rapidement afin qu'ils atteignent plus rapidement la destination. Ce qui garantira pour le nœud malhonnête une place sur le chemin.

3.2 Attaques élémentaires portant sur les réponses de route :

3.2.1 Suppression d'une réponse de route :

Ce type d'attaque n'a un sens que si le nœud malhonnête a été choisi sur la **route** reliant la source à la destination. Dans ce cas, la **suppression** de la réponse de route empêche la formation du chemin vers la destination et entraine des messages de contrôle supplémentaires suite à l'initialisation d'un nouveau processus de création de route, ce qui dégrade la qualité de service.

3.2.2 Modification d'une réponse de route :

Comme pour les demandes de route, le nœud malhonnête peut jouer sur le numéro de séquence de la destination et/ou le nombre de saut dans une RREP en augmentant le premier et en diminuant le second. Ces paramètres sont pris en compte lors de la mise à jour du chemin vers la destination : une mise à jour est possible si le numéro de séquence reçu dans la demande de route est plus grand que celui stocké dans la table de routage ou les numéros de séquence sont égaux et le nombre de sauts reçu est plus petit que celui stocké dans la table de routage. Cette intervention permet de garder le chemin qui passe par le nœud malhonnête même si un autre chemin plus court est proposé par un autre nœud.

3.2.3 Fabrication d'une réponse de route :

Certaines attaques peuvent être effectuées sans pour autant être choisi sur le chemin.

C'est le cas des attaques par **fabrication** :

– Fausse réponse : à la réception d'une demande de route, le nœud malhonnête fabrique une réponse de route même s'il n'a pas de chemin valide vers la destination (attaque Blackhole). Dans un autre cas de figure, le nœud malhonnête répond avec une réponse de route même s'il n'est pas supposé le faire lorsque le drapeau D est à 1 (indiquant que seulement la destination doit répondre).

– Réponse active : des réponses de routes sont fabriquées et injectées dans le réseau même sans avoir reçu une demande de route au préalable. Dans ce cas le nœud malveillant peut jouer sur des champs pour produire l'effet désiré. Une variante de cette attaque vise à déborder la table de routage d'une cible en proposant des routes (via RREP) vers des nœuds (nouveaux ou inexistants).

– En écoutant la transmission d'une réponse de route qui ne lui est pas destinée, un nœud malhonnête peut fabriquer et injecter un paquet RREP proposant un chemin plus court et plus frais provoquant la mise à jour du chemin vers la destination qui passe dorénavant par le nœud malveillant.

3.3 Attaques élémentaires portant sur les erreurs de route :

3.3.1 Suppression d'une erreur de route :

Comme c'est le cas pour les RREQ et RREP, en **effaçant** une RERR, un nœud malhonnête peut retarder la détection des liens défaillants. Cependant, l'impact de cette attaque est restreint du fait que les nœuds en amont découvrent le problème et demandent l'établissement de nouvelles routes.

3.3.2 Modification d'une erreur de route :

Un nœud malhonnête peut **modifier** des erreurs de route avant de les retransmettre. Ainsi, il peut supprimer des destinations non-joignables pour faire croire qu'elles le sont encore et ajouter des destinations qui sont joignables et actives pour faire croire qu'elles ne le sont plus et les désactiver.

3.3.3 Fabrication d'une erreur de route :

Un nœud malhonnête peut **fabriquer** un message d'erreur de route et déclarer autant de routes non-joignables causant l'invalidation des entrées correspondantes dans la table de routage des nœuds recevant le message de contrôle.

3.4. Attaques composés :

Une attaque composée est la combinaison de plusieurs attaques élémentaires afin d'atteindre des objectifs plus évolués. Dans la section suivante nous présentons quelques attaques composées.

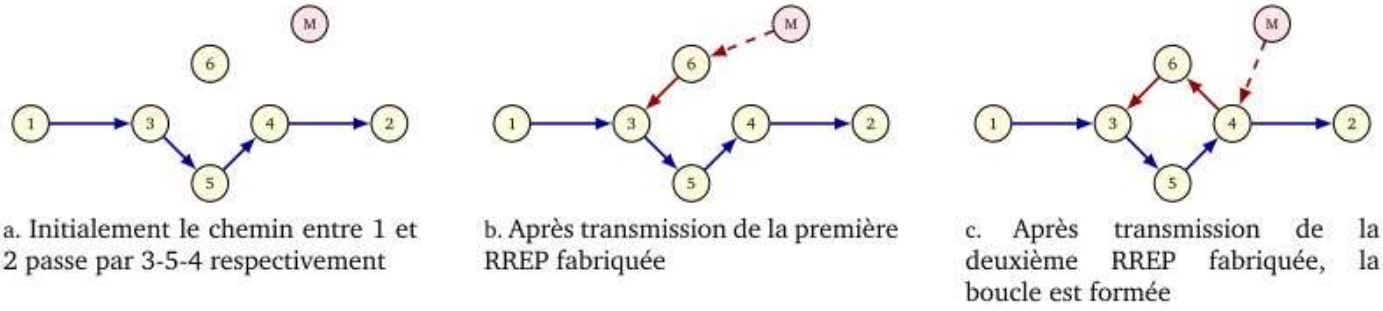
3.4.1 Répétition régulière d'attaques élémentaires :

Cette attaque se base sur une répétition régulière d'une attaque élémentaire pour avoir un impact permanent. Par exemple, l'envoi continu de messages de demande de route RREQ fabriqués à destination d'une cible est efficace pour empêcher celle-ci de recevoir les messages des autres nœuds. De même, l'envoi de réponses de route fabriquées est efficace pour empêcher la cible d'envoyer des messages aux autres nœuds (puisque le nœud malhonnête devient le prochain saut vers les autres nœuds). En combinant ces deux attaques composées, un nœud malhonnête peut isoler sa cible.

3.4.2. Création d'une boucle de routage :

La formation d'une boucle de routage dans une route déjà établie est faite en fabriquant deux réponses de routes. Nous présentons cette attaque à travers l'exemple suivant :

On suppose l'existence d'un chemin entre le nœud 1 et le nœud 2 passant par 3, 5 et 4. Le nœud malicieux M fabrique une première réponse de route où l'adresse source est initialisée à 1, l'adresse destination à 2 et un numéro de séquence de la destination 2 supérieur au numéro de séquence actuel. Il fait croire que ce paquet est envoyé par le nœud 3 à destination du nœud 6 (valeurs à mettre respectivement dans le champ adresse source et destination de l'entête IP). Ceci provoque l'envoi de tout paquet reçu à destination de 2 vers le nœud 3 (voir figure 5.4b). Ensuite, M fabrique une deuxième réponse de route équivalente à la première sauf qu'il fait croire qu'elle est à destination du nœud 4, provenant du nœud 6. Cette action garantie que tout paquet transféré vers 2 à travers 4 est envoyé vers 6, ce qui complète la boucle.



a. Initialement le chemin entre 1 et 2 passe par 3-5-4 respectivement

b. Après transmission de la première RREP fabriquée

c. Après transmission de la deuxième RREP fabriquée, la boucle est formée

Figure 5.4 : Boucle de routage

3.4.3 Création d'un tunnel

L'attaque du trou de ver peut aussi être facilement appliquée sur le protocole de routage AODV. Elle a été présentée comme étant une attaque agrégée. Elle nécessite la coopération de deux ou plusieurs nœuds malhonnêtes qui falsifient la longueur des routes. Ils proposent des routes plus courtes grâce au tunnel qu'ils construisent. Ensuite, le nœud malicieux recevant un message l'encapsule pour le transférer vers l'autre bout du tunnel où il sera décapsulé et rediffusé. De cette manière, les différents sauts formant le tunnel ne sont pas comptabilisés et ainsi le chemin obtenu parait plus court.

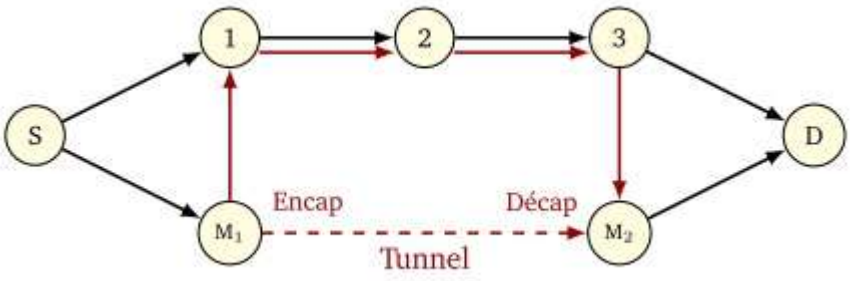


FIGURE 5.5 – Attaque du tunnel

La figure 2.4 montre un exemple de tunnel créé entre les nœuds malicieux M 1 et M 2. D reçoit deux chemins : [S–1–2–3–D] et [S–M 1 –M 2 –D]. Le chemin le plus court est choisi (à savoir [S–M 1 –M 2 –D]). Or ce chemin est en réalité plus long [S–M 1 –1–2–3–M 2 –D].

Ce tableau présente les conséquences des attaques possibles affectant les paquets de contrôle du protocole AODV étudié.

champ	Attaque
Id_source	Fabriquer un message avec l'identité d'un nœud légitime.
Id_destination	L'attaquant crée des routes vers des destinations inexistantes pour consommer l'énergie du réseau (déni de service).
Id_Broadcast	Ce champ permet d'identifier la requête d'une manière unique et de supprimer une demande déjà traitée. L'adversaire incrémente ce champ pour invalider toute les futures requêtes venant d'un nœud légitime. Décrémenter ce champ dans une requête valide empêche la mise à jour de la table de routage des nœuds intermédiaires car la requête sera considérée comme une demande déjà traitée.
Nbr_saut	Quand le nœud malicieux reçoit un paquet, il suffit qu'il positionne le nombre de saut à 0 pour se présenter comme relai à faible coût.
Num de seq	Un numéro de séquence élevé force la mise à jour au niveau des nœuds récepteurs du paquet. En manipulant ce champ l'adversaire peut injecter de fausses informations de topologie.

Tableau 5.1 ATTAQUES CONTRE AODV

4. Etude de l'attaque Blackhole :

L'attaque blackhole **[73]** est un problème sérieux qui doit être résolu dans les réseaux sans fil, elle peut s'effectuer au moment où un nœud source initie un processus de découverte de route en émettant un paquet RREQ, le nœud corrompu en le recevant va répondre par un paquet RREP avec un numéro de séquence non seulement erroné mais également élevé afin d'augmenter ses chances de faire partie de la route. En effet, si son paquet RREP atteint la source le premier par rapport aux réponses des nœuds légitimes, il peut s'intégrer dans la route pour intercepter et contrôler une partie ou la totalité du trafic échangé au sein du réseau, de façon à pouvoir surveiller, bloquer ou même détourner certains flux du trafic. Le trafic absorbé peut être donc soit redirigé vers un autre nœud soit disparaître complètement.

Cette attaque est grave dans la mesure où les nœuds légitimes vont mettre à jour leurs tables de routage avec des informations fausses et le nœud malveillant n'a pas besoin d'intervenir une seconde fois.

La figure ci-dessous illustre cette attaque où la source S veut transmettre des données vers la destination D, elle diffuse une requête RREQ (Route REQuest), le paquet RREQ va être reçu par les nœuds N1, N2, N3.

Supposons le nœud N3 a une route vers la destination dans sa table de routage, le nœud N3 génère un paquet de réponse RREP et mettre à jour sa table de routage par le nombre de sauts et le numéro de séquence de la destination.

Le numéro de séquence de la destination est un entier de 32 bits associé à chaque route et permet l'utilisation des routes les plus fraîches autrement dit les plus nouvelles. Une route est jugée fraîche que si la base du numéro de séquence de la destination est assez élevée [74].

Le nœud N3 va envoyer le paquet vers le nœud M, tant que les nœuds N1 et N2 n'ont pas une route vers la destination D, ils seraient à nouveau diffusé le message de contrôle RREQ. Ainsi le paquet RREQ diffusé par le nœud N3 devrait également être reçu par le nœud M (supposons M est un nœud malicieux). Donc le nœud M va générer un faux message de contrôle RREP et l'envoyer au nœud N3 avec un numéro de séquence de destination très élevé, qui serait ensuite envoyé au nœud S.

Cependant, tant que le numéro de séquence de destination est élevé, la route à partir du nœud N3 sera considéré comme plus fraiche et donc la source serait commencée à envoyer des paquets de données au nœud N3. Le nœud N3 serait envoyer les même paquets au nœud malicieux. Le message de contrôle RREQ à partir du nœud N1, finirait par atteindre le nœud D (nœud de destination), ce qui générerait un message de contrôle RREP et la route du retour. Toutefois, le nœud S a déjà reçu un paquet de réponse RREP avec numéro de séquence supérieur à celui de D, le nœud S ignore les deux véritables messages de contrôle RREP. Pour chaque message de contrôle RREP reçu, la source devrait d'abord vérifier si elle possède une entrée pour la destination dans la table de routage ou non. S'il en trouve un, le nœud source serait de vérifier si le numéro de séquence de destination dans le message de contrôle reçu est plus élevé que celui qu'il a envoyé dans la dernière RREQ ou non. Si le numéro de séquence de destination est plus élevé, la source met à jour sa table de routage avec le nouveau message RREP, sinon le message de contrôle RREP sera rejeté.

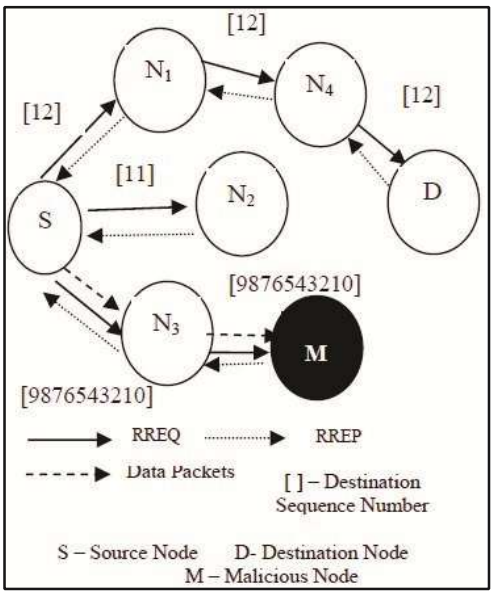


Figure 5.6: Attaque Black Hole

5. Attaques dans les Réseaux Ad hoc Sans fil [75]

L'authentification, la non-répudiation, la disponibilité, l'intégrité, la confidentialité et la vie privée sont les buts pour un réseau ad hoc sécurisé [24]. Dans cette section, nous classifions les attaques contre les protocoles de routage ad hoc et particulièrement contre AODV, selon ces buts de sécurité. La figure 2 récapitule la majeure partie des attaques dans un réseau ad hoc mobile exécutant le protocole de routage AODV. Notez que d'autres attaques existent, ça opère aux différents niveaux d'abstraction, par exemple, au niveau physique ou au niveau d'application. Ci-après, nous nous concentrons exclusivement sur les attaques qui affectent l'acheminement du trafic dans les MANET.

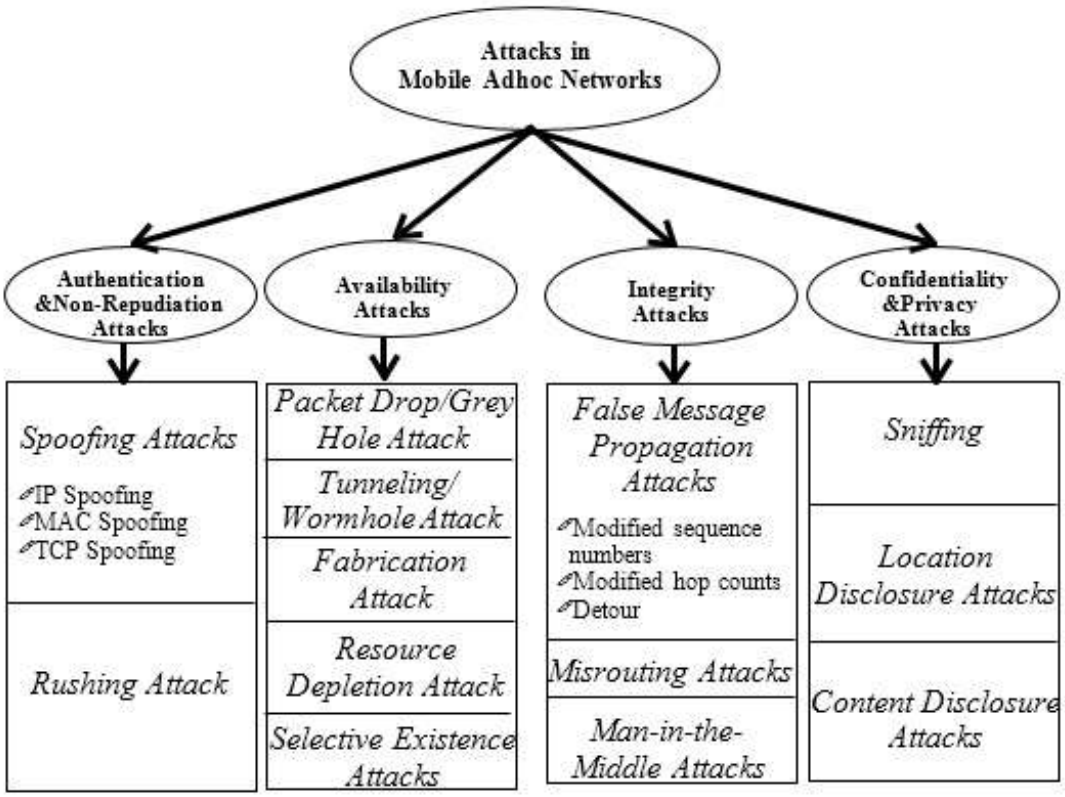


Figure 5.7 Attaques dans AODV MANETs

Les attaques d'authentification et de non-répudiation

L'authentification permet à un nœud de vérifier l'identité d'un nœud de pair avec lequel il communique. La non-répudiation est la capacité de prouver qu'un expéditeur a envoyé un message. La plupart des protocoles de routage ad hoc utilisent des adresses MAC ou IP pour identifier les hôtes du réseau. Par conséquent, l'usurpation d'une de ces deux adresses est la méthode la plus simple de s'attaquer aux objectifs de sécurité de l'authentification et la non-répudiation.

Attaques de Disponibilité

La disponibilité garantit que les services de réseau (par exemple, la bande passante et la connectivité) sont accessibles aux entités autorisées en temps opportun. Les sections suivantes présentent une variété d'attaques par déni de service, qui sont utilisées pour réduire ou refuser totalement l'accès aux services de réseau.

Dropping of packets (L'abandon de paquet). Ces attaques se font en laissant tomber des paquets de données ou contrôlent des paquets. Dans le premier cas, un nœud malveillant, après avoir annoncé un chemin correct à la destination, laisse simplement tomber des paquets de données pour effectuer une attaque par déni de service. Dans le dernier cas, un nœud malveillant laisse tomber des paquets de contrôle envoyés par d'autres nœuds, en lançant en même temps une découverte de route chaque fois qu'il doit envoyer des données. Ceci est également connu sous le nom misbehavior routage (routage mauvaise conduite).

Attaques de Fabrication. Un nœud malveillant peut lancer une attaque par déni de service contre un nœud de destination en usurpant l'identité d'un nœud le long d'un chemin vers cette destination et envoyer continuellement des messages d'erreur d'itinéraire pour le nœud de destination. En recevant les messages d'erreur de route, les nœuds à l'aide de cette route vont supprimer l'entrée de table de routage pour le nœud de destination. En l'absence

de routes alternatives, ou en transmettant en permanence les messages d'erreur, la communication vers le nœud de destination peut être empêchée avec succès.

Attaques d'Épuisement de Ressource. L'utilisation de la bande passante du réseau est une question importante dans un réseau sans fil ad hoc. Un seul nœud ou un groupe de nœuds dans collusion peuvent effectuer une attaque de l'épuisement des ressources. Par exemple, deux nœuds peuvent transférer de grands volumes de données entre eux, en obstruant ainsi le réseau et en réduisant la bande de fréquence de réseau disponible Une attaque d'épuisement de ressource peut aussi être accomplie en utilisant des messages de contrôle. Par exemple, un nœud malveillant peut inonder le réseau avec les demandes de route vers les adresses aléatoires. Cela conduirait à une tempête de RREQ qui épuise la bande passante disponible sur le réseau.

Attaques d'Existence Sélectives. Dans ces attaques, un nœud malveillant se comporte de manière égoïste, en utilisant le réseau pour ses propres besoins, sans participer à l'ensemble du processus de routage. Par exemple, un nœud malveillant ne peut pas envoyer des messages HELLO. Par conséquent, les nœuds dans le réseau ad hoc ne connaissent pas l'existence du nœud malveillant. Toutefois, lorsque le nœud malveillant doit envoyer un paquet de données, il effectue une découverte de route et obtient un itinéraire vers la destination. Ensuite, le nœud malveillant transmet uniquement les paquets en provenance des nœuds voisins que le nœud malveillant doit dépendre à envoyer des données. Lorsque le nœud malveillant n'a plus besoin d'utiliser le réseau, il rebascule sur "mode silencieux" et, en conséquence, les entrées de routage pour le nœud malveillant sont invalidées dans les tables de routage de ses voisins.

Attaques d'Intégrité

L'intégrité garantit que le message n'est pas modifié sur son chemin vers la destination. Dans ce qui suit, une série d'attaques d'intégrité sont discutés.

Attaques de Propagation de Faux Messages. Un cas de ce type d'attaque est la redirection avec les numéros de séquence modifiée. Dans cette attaque, un nœud malveillant annonce un trajet à un nœud avec un nombre d'ordre de destination plus grand que la valeur authentique. En le faisant, il détourne la circulation vers l'attaquant parce que les nœuds choisiront le RREP avec le plus haut nombre d'ordre de destination. Cette attaque est spécifique à AODV. La modification du nombre de sauts a également un effet similaire sur les itinéraires choisis.

Un autre exemple de ce type d'attaques, est l'attaque de détour, où l'attaquant ajoute un certain nombre de nœuds virtuels à une route pendant le processus de découverte de route. En conséquence, le trafic est détourné vers les autres routes qui semblent être plus courte, et le nœud attaquant peut économiser de l'énergie car il n'a pas à transférer les paquets vers cette destination.

Attaque Misrouting. Dans cette classe d'attaques, un nœud malveillant tente d'envoyer un paquet de données à la mauvaise destination. Par exemple, cela peut être réalisé en envoyant un paquet de données au mauvais saut suivant dans la route vers la destination ou en modifiant l'adresse de destination finale du paquet de données.

Attaques man-in-the-Middle. Un nœud malveillant peut combiner l'attaque de spoofing et l'abandon de paquets pour effectuer une attaque man-in-the-middle. Le nœud malveillant utilise sa place sur la route comme la première étape d'une attaque man-in-the-middle en en refusant acheminer les demandes vers la destination. L'attaquant doit être le seul nœud dans la plage de destination, ou doit être en mesure d'empêcher toute autre demande de route vers la destination. L'attaquant envoie une fausse réponse de route au nœud source et établit une route avec la source. L'attaquant envoie une nouvelle demande de route vers le nœud de destination, établit une route avec la destination, puis laisse tomber le paquet de réponse sur son chemin vers le nœud source. En faisant cela, l'attaquant contrôle la communication entre la source et la destination.

Confidentialité et Attaques de Vie privée

La confidentialité garantit la non-divulgation de renseignements personnels stockés à un nœud à un autre nœud du réseau. La confidentialité garantit que certaines informations sont divulguées uniquement aux entités autorisées.

Attaques de Divulgation d'Emplacement. L'attaque de divulgation d'endroit révèle des informations physiques sur un nœud particulier (par exemple, l'emplacement du commandant d'une troupe).Cette attaque est basée sur le principe que, pour tous les protocoles de routage multi-sauts deux nœuds consécutifs en route doit être géographiquement proches. Cette information peut être recueillie par l'écoute promiscuité ou en utilisant des outils tels que *Traceroute*. En utilisant ce principe de façon récursive, les informations d'emplacement des nœuds dans un réseau ad hoc peuvent être divulguées.

Attaques de divulgation de contenu. L'attaque de divulgation de contenu permet à un utilisateur malveillant d'apprendre le contenu des messages transmis. Les protocoles de cryptage comme WEP, qui sont une partie de la norme 802.11, sont utilisés pour prévenir la divulgation de contenus de message aux nœuds non autorisés. Pour violer la vie privée de la communication, un attaquant doit casser le protocole de cryptage WEP et n'importe quels protocoles de cryptage supplémentaires utilisés.

6. Solutions de l'attaque Black Hole

Plusieurs solutions ont été proposées pour pallier des problèmes de sécurité dans les réseaux sans fil (ad hoc). Dans cette section nous allons présenter les différentes propositions dans la littérature pour éviter l'attaque black hole.

[Deng et al, 2002] **[76]** ont proposé une solution contre l'attaque trou noir en modifiant le protocole AODV. Dans cette méthode chaque nœud intermédiaire doit inclure l'information « next hop » quand il envoie un paquet RREP. Une fois la source a reçu le paquet RREP et avant d'envoyer les paquets de données, il extrait l'adresse du « next hop » et lui envoie une nouvelle demande de route (Further Request) afin de vérifier qu'il possède une route vers le nœud intermédiaire qui a envoyé le message de réponse, et qu'il a aussi une route vers le nœud destination. Le « next hop » répond avec un paquet de réponse de route (Further Reply) qui comprend le résultat de contrôle. La source vérifie les informations des paquets FRREP et agit selon les règles suivantes:

1) Si le « next hop » possède une route vers le nœud intermédiaire et la destination, la source établit la route reçu du nœud intermédiaire et commence l'envoi des données.
2) Si le « next hop » a une route vers la destination, mais n'a pas de route vers le nœud intermédiaire, la source suppose que le nœud intermédiaire est un nœud malicieux. Ensuite, elle initie l'envoi des données via la nouvelle route à travers le next hop et diffuse un message d'alarme dans le réseau afin d'isoler le nœud malveillant.
3) Si le « next hop » n'a pas de routes vers le nœud intermédiaire et la destination, la source lancera un nouveau processus de découverte de route, et envoie également un message d'alarme afin d'isoler le nœud malveillant.

Le mécanisme proposé est efficace dans la détection de l'attaque Blackhole, cependant, l'envoi d'un paquet FRREQ à partir du nœud source vers le « next hop » et l'attente du paquet FRREP du « next-hop » augmente la

charge du routage « overhead » entre la source et le « next hop », surtout quand ce mécanisme est appliqué sur un réseau à grande échelle et la distance entre la source et le nœud malicieux est longue.

[Al-Shurman et al, 2004] **[77]** ont proposé deux solutions conçues pour cibler l'attaque Blackhole dans le protocole AODV. La première solution proposée consiste à trouver plus d'une route vers la destination (au moins trois routes différentes). La source envoie un paquet RREQ au nœud destination en utilisant ces trois routes. La destination, le nœud malicieux et les nœuds intermédiaires vont répondre à ce paquet. Le nœud expéditeur met ses paquets de données dans un tampon jusqu'à ce qu'il reçoit plus d'une réponse RREP ; lorsque la source reçoit des RREP, si les routes à destination ont des nœuds partagés, la source peut reconnaître une voie sûre vers la destination, et les paquets vont être transmis. Si aucuns nœuds partagés ne semblent être dans ces routes redondantes, l'expéditeur attendra une autre RREP jusqu'à ce qu'un chemin avec des nœuds partagés identifié ou le temps d'attente soit expiré.
Cette solution peut garantir à trouver une route sécurisé vers la destination, mais le principal inconvénient est le délai d'attente. Plusieurs paquets RREP doivent être reçues et traitées par la source. En outre, s'il n'y a pas de nœuds partagés entre les routes, les paquets ne seront jamais envoyés.
La seconde solution proposée exploite le numéro de séquence inclus dans l'en-tête de chaque paquet. Le nœud dans cette situation a besoin d'avoir deux tables supplémentaires; la première table comprend les numéros de séquence du dernier paquet envoyé à chaque nœud dans le réseau.
La deuxième table contient le numéro de séquence reçu de chaque expéditeur. Pendant la phase de réponse de route, le nœud intermédiaire ou la destination doivent inclure le numéro de séquence du dernier paquet reçu de la source qui déclenche la demande de route. Une fois la source reçoit ce RREP, il va extraire le dernier numéro de séquence, puis le comparer avec la valeur enregistrée dans sa table. Si elle correspond, la transmission aura lieu. Si ce n'est pas, ce nœud est un nœud malveillant, alors un message d'alarme sera diffusé pour avertir le réseau sur ce nœud. Toutefois, les deux solutions ont le délai de bout en bout comme inconvénient.

Selon la solution proposée par [Tamilselvan et.al, 2007] **[78]**, la source doit attendre d'autres réponses avec des détails sur le prochain saut avant d'envoyer les paquets de données vers la destination. Une fois un nœud a reçu la première RREQ, il fixe un temporisateur « timer » dans le "Timer Expired Table" pour collecter les nouvelles demandes venant des différents nœuds et les stocker dans un ordre séquentiel, la source va stocker le « numéro de séquence », et « le moment d'arrivé du paquet » dans « Collect Route Reply Table » (CRRT). La valeur "timeout" est basée sur le temps d'arrivé du premier RREQ. Elle vérifie d'abord dans la table CRRT afin de déterminer s'il existe un next hop répété dans les réponses de route reçus, dans ce cas il assume que les chemins sont corrects ou la chance de chemins malveillants est limitée. S'il n'y a pas de répétition, une route aléatoire est sélectionnée de la table CRRT. L'inconvénient de la solution proposée est le délai de bout en bout, puisque le nœud source doit attendre d'autres réponses de route avant d'envoyer les données.

[Lalit Himral et.al, 2011] **[79]** ont proposé une méthode pour trouver les routes sécurisées et prévenir les nœuds malveillants dans les MANET en vérifiant s'il existe une différence importante entre le numéro de séquence du nœud source et le nœud intermédiaire qui a envoyé la première RREP. En règle générale, la première réponse sera du nœud malveillant avec un numéro de séquence de destination très élevée, la réponse sera stockée comme la première entrée dans RR-table. Ensuite, comparez le premier numéro de séquence de destination reçu avec le numéro de séquence du nœud source, s'il existe une grande différence entre eux, il est certainement que cette réponse vient du nœud malveillant, par conséquent supprimer immédiatement cette entrée de la table. La méthode proposée offre les avantages suivants : (1) Le nœud malveillant est identifié dans la phase initiale et il est retiré immédiatement. (2) Aucune modification n'est faite dans les autres opérations du protocole AODV. (3) Une

meilleure performance en légère modification. Cependant la méthode ne peut pas trouver de multiples nœuds malicieux.

La solution proposée dans [Payal N et al, 2009] **[80]** modifie le comportement de l'AODV pour inclure un mécanisme permettant de vérifier le numéro de séquence de RREP reçu. Quand le nœud source reçoit un paquet RREP il compare le numéro de séquence du RREP reçu à une valeur de seuil. Le nœud répondant est soupçonné d'être un trou noir si son numéro de séquence est supérieur à la valeur de seuil. Le nœud source ajoute le nœud suspect à sa liste noire, et se propage un message de contrôle appelé une alarme pour faire connaître la liste noire pour ses voisins. Le seuil est la moyenne calculée de la différence entre le numéro de séquence de destination dans la table de routage et le numéro de séquence de destination dans le RREP dans une période de temps. Le principal avantage de ce protocole est que le nœud source annonce le trou noir à ses voisins afin d'être ignoré et supprimé.

Un algorithme présenté dans [Subash et al, 2011] **[81]** pour détecter l'attaque trou noir dans un MANET basé sur un prétraitement appelé Pre_Process_RREP, il est simple ainsi il ne change pas le fonctionnement de l'un des nœuds intermédiaires ou de destination. Il n'a même pas modifié le fonctionnement normal de l'AODV. Le processus continue à accepter les paquets RREP et appelle un processus appelé Compare_Pkts (p1 paquets, p2 paquet) qui compare le numéro de séquence de destination des deux paquets et sélectionne le paquet avec un numéro de destination supérieur si la différence entre les deux numéros n'est pas sensiblement élevée. Le paquet contenant exceptionnellement un numéro de séquence de destination élevé est soupçonnée d'être un nœud malveillant et un message d'alerte contenant l'identification du nœud est généré et diffusé vers les nœuds voisins de sorte qu'il peut être isolé du réseau et peut maintenir une liste de ces nœuds malveillants.

7. Les IDSs conçus dans le cadre d'AODV.

7.1 Specification-Based IDS for AODV [82]

Les premiers IDS basés sur les spécifications dans les MANET sont proposé par Tseng et al. [82]. Ils utilisent les moniteurs de réseau (NM) qui sont supposés couvrir tous les nœuds. Les nœuds se déplaçant hors de la zone de surveillance du réseau actuel sont supposés bouger vers d'autres rangs des moniteurs de réseaux. D'autres hypothèses sont :

- ✓ Les moniteurs de réseau connaissent les adresses IP et MAC de tous les nœuds, et les adresses MAC ne peuvent pas être falsifiés.
- ✓ Les moniteurs de réseau et leurs messages sont sécurisés.
- ✓ Si certains nœuds ne répondent pas aux messages de diffusion, cela ne va pas entraîner de sérieux problèmes.

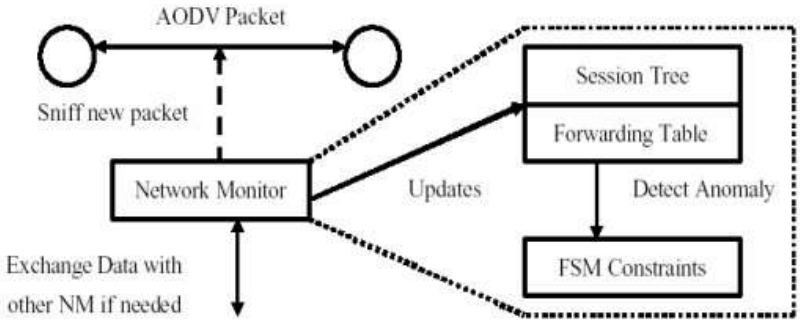


Figure 5.8 : Architecture du moniteur réseau

Les moniteurs réseau utilisent des machines à état fini (FSM) comme caractéristiques des opérations d'AODV, surtout pour le processus de découverte de route et maintient une table de transfert pour chaque nœud surveillé. Chaque message de demande de route (RREQ) et de réponse de route (RREP) dans la gamme du moniteur réseau est surveillé dans un flux de demande-réponse. Lorsqu'un moniteur de réseau a besoin d'informations sur les messages précédents ou d'autres nœuds pas dans sa gamme, il peut demander aux moniteurs de réseau voisin. Dans des conditions de mobilité la communication réseau entre moniteurs augmente puisque les nœuds contrôlés ou/et les paquets bougent souvent de la gamme du nœud de contrôle.

Les auteurs ont également modifié le protocole de routage AODV en ajoutant un nouveau champ : le nœud précédent. Puisque RREQs sont des messages d'émission, il est nécessaire de garder la trace du chemin RREQ. Le nœud précédent est nécessaire pour détecter un certain type d'attaques telles que l'envoi d'un RREP vers un nœud qui n'est pas sur le chemin inverse. [82].

Les futurs travaux comprennent l'expérimentation via la simulation de réseau NS-2, en dressant le portrait du réseau QoS (la Qualité de Service) pour réduire les faux positifs en séparant la perte de paquets, erreur de paquet, et la génération de paquets à travers la définition de seuils raisonnables pour le profil actuel, et l'architecture du raffinage via l'approche un P2P (peer-to-peer)

C'est une approche prometteuse qui peut détecter les attaques connues et inconnues contre les protocoles de routage qui ont clairement défini des spécifications. On prétend que cela découvre la plupart des attaques avec une surcharge minimale en temps réel. Toutefois, certaines des hypothèses retenues dans ce document ne sont pas très réalistes. Par exemple, en supposant que les moniteurs réseau couvre tous les nœuds du réseau et avoir des adresses IP et MAC de tous les nœuds. L'extensibilité est une des caractéristiques importantes sur de nombreuses applications MANET où les nœuds peuvent rejoindre ou quitter indépendamment un réseau et se déplacent fréquemment. Il est irréaliste de supposer que les adresses MAC ne peuvent pas être aisément falsifiables. En outre, l'abandon de certains messages diffusés dans le réseau peut affecter tous les services réseau si le nœud laissant tomber des messages est à un point critique. En outre, les détails de l'architecture ne sont pas traités (telles que les positions des moniteurs réseau de Manet où les changements de topologie arbitrairement).

7.2 IDS à base du protocole AODV (AODV protocol-base IDS) [83]

Bhargava et Agrawal [83] ont proposé une détection d'intrusion et un modèle de réponse (IDRM) pour renforcer la sécurité dans le protocole de routage ad hoc on demand distance vector(AODV).

Figure 5.9 illustre comment le IDRM assure la sécurité dans le protocole AODV. Dans ce schéma, chaque nœud utilise un IDRM qui utilise les informations de voisinage pour détecter les débordements de ses voisins. Lorsque le nombre de mauvaise conduite d'un nœud dépasse un seuil prédéfini, l'information est envoyée à d'autres nœuds dans le cadre de la réponse globale. Les autres nœuds reçoivent cette information, vérifier leur "malcount" local pour ce nœud malveillant et ajouter leurs résultats à l'initiateur de la réponse. Dans le modèle de réponse d'intrusion (IRM), un nœud identifie qu'un autre nœud a été compromis quand ses malcount augmente au-delà de la valeur de seuil qui aurait été compromise nœud. Dans ce cas, il propage ces informations à l'ensemble du réseau en transmettant un type spécial de paquet appelé paquet "MAL". Si un autre nœud aussi suspects que le nœud qui a été détecté comme compromis, il rend compte de ses soupçons au réseau et retransmet un autre type spécial de paquet, appelé "REMAL". Si deux ou plusieurs nœuds rapportent des soupçons concernant un nœud particulier, un autre des paquets spéciaux, appelés paquet " PURGE", est transmis pour isoler le nœud malveillant à partir du réseau. Tous les nœuds qui ont une route par le nœud compromis cherchent de nouvelles routes. Tous les paquets reçus d'un nœud compromis sont abandonnés.

Le Modèle de Détection d'Intrusion prétend capturer les attaques suivantes :

- ✓ Distribution de fausses demandes de route
- ✓ déni de service
- ✓ Destination est compromise
- ✓ Usurpation d'identité
- ✓ Divulgation d'Informations de Routage

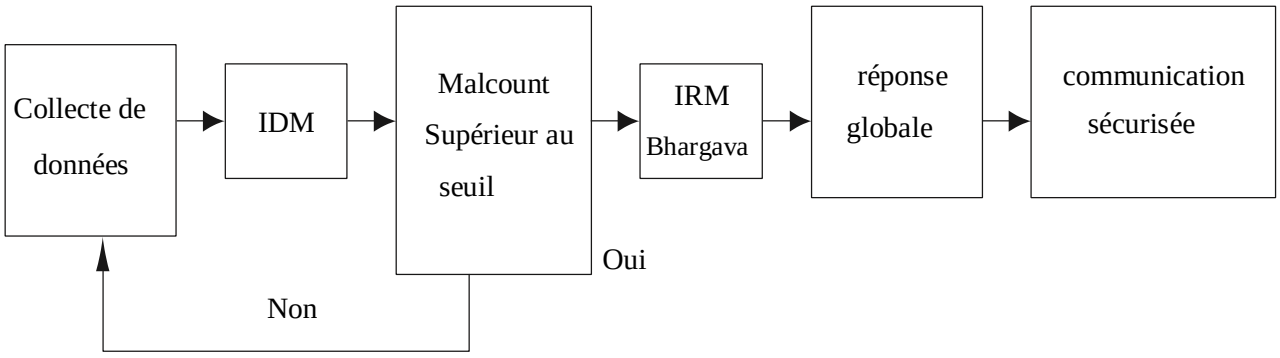


Figure 5.9 : Manipulation des attaques internes

Ils ont proposé un projet de sécurité pour pro-activement prévenir des attaques intérieures. Le modèle de réponse et de détection d'intrusion est présenté pour identifier des attaques et les prévenir. Le résultat de leur implémentation montre que les frais généraux est marginales et a des effets négligeables sur la performance du réseau tout en rendant le protocole robuste.

Le Modèle de Réponse d'Intrusion est un comptoir qui est incrémenté chaque fois qu'un acte de malveillance est rencontré. Lorsque la valeur atteint un seuil prédéfini, le nœud malveillant est isolé. Les auteurs ont fourni des statistiques pour la précision du modèle. Ils travaillent sur la définition plus d'attaques internes et planifier à trouver des solutions pour eux.

De plus, ils prévoient d'introduire la sécurité pour les attaques externes et incorporer ceux avec leur Modèle de Réponse et de Détection d'Intrusion aussi.

7.3 IDS distribué et Coopératif (Distributed and Cooperative IDS) [84]

Le premier IDS pour MANET proposé par Zhang et Lee est un IDS distribué et coopératif. Dans cette architecture, chaque nœud a un agent IDS qui détecte les intrusions localement et collabore avec des nœuds voisins (par le biais des canaux de communication de haute confiance) pour la détection globale à chaque fois que les données disponibles ne sont pas concluantes et une recherche plus large est nécessaire. Quand une intrusion est découverte un agent IDS peut ou déclencher une réponse locale (par ex. en alertant l'utilisateur local) ou une réponse globale (qui coordonne les actions entre les voisins de nœuds).

Puisque les règles experts peuvent détecter uniquement les attaques connues et les règles ne peuvent pas facilement être mis à jour sur un réseau ad hoc sans fil, la détection à base d'anomalie statistique est préférée sur la détection à base de mauvaise usage. On compte sur les données locales pour la détection à base d'anomalie statistique : le mouvement du nœud (la distance, la direction, la vitesse) et le changement de table de routage PCR : Le pourcentage de routes modifiées, PCH : pourcentage de modifications dans pourcentage de modifications dans la somme des sauts tous les itinéraires).

Une multicouche intégrée de détection d'intrusion et réponse est proposé, permettant à de différentes attaques d'être détectées au niveau de la couche plus efficace. Il est censé atteindre un taux de détection plus élevé avec un taux plus faible de faux positifs.

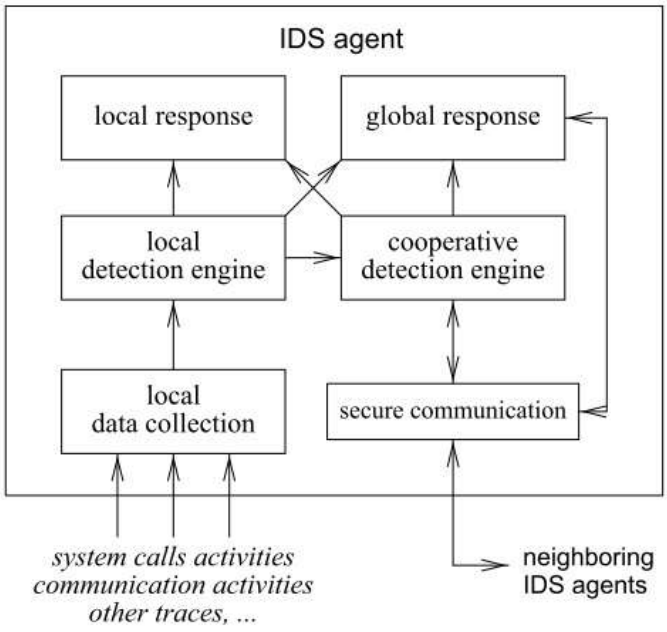


Figure 5.10 : un modèle conceptuel pour un agent IDS

Les algorithmes de classification Ripper et SVM-Light sont utilisés. Dans leur recherche ultérieure [84], ces algorithmes sont évalués sur trois protocoles de routage : AODV, DSR et DSDV en utilisant les taux de détection et de la métrique de taux de fausse alerte. SVM-Light est montré pour avoir de meilleures performances que RIPPER. Il est aussi montré que

les protocoles avec la forte corrélation parmi les changements de différents types d'informations (l'endroit, routage, etc.) ont la meilleure performance, donc les protocoles réactifs (sur demande) sont plus appropriés pour ce système que les protocoles proactifs (déterminés par des tables). En outre, il est indiqué que l'IDS fonctionne mieux avec des protocoles qui comprennent une certaine redondance (tels que la redondance des chemins de DSR). Cependant, l'effet de la mobilité n'est pas discuté.

C'est une des rares approches tenant compte de la mobilité en surveillant les mouvements de nœud. Cela peut diminuer les faux positifs résultant de la mobilité du nœud.

Toutefois, il ne fait que refléter la mobilité locale pas la mobilité du réseau. En outre, chaque nœud doit avoir un GPS (Global Positioning System) pour obtenir ces données de mobilité. Il convient de souligner qu'il peut être appliqué à tous les protocoles de routage, car il utilise les informations de routage minimales. Il permet également l'ajout de nouvelles fonctionnalités pour un protocole spécifique.

7.4 Un outil de détection d'intrusion pour AODV basée sur des réseaux sans fil Ad hoc (An Intrusion Detection Tool for AODV-based Ad hoc Wireless Networks) [85]

Giovanni Vigna et al. ont conçu et mis en place un outil de détection d'intrusion, appelé AODVSTAT, pour détecter les attaques contre le protocole de routage AODV dans les réseaux sans fil ad hoc. AODVSTAT est basé sur la technique d'analyse de transition d'état (STAT), qui a été initialement mise au point pour modéliser les intrusions sur l'hôte sur le réseau et dans un environnement câblé. [85].

Dans STAT, les pénétrations informatiques sont décrites comme les ordres d'actions qu'un attaquant exécute pour compromettre la sécurité d'un système informatique. Les États représentent un instantané de la mémoire versatile, semi-permanente et permanente d'un système. Les États représentent un instantané de la mémoire versatile, semi-permanente et permanente d'un système. Une description d'une attaque a un état de départ sûr, zéro ou plusieurs états intermédiaires, et au moins un État fin compromis. Les états sont caractérisés par des affirmations, qui sont des fonctions avec zéro, un ou plusieurs arguments retournant des valeurs booléennes. En général, ces affirmations décrivent certains aspects de l'état de sécurité du système, telles que la propriété de fichier, identification de l'utilisateur ou caractéristiques du trafic réseau. Les transitions entre états sont annotées avec des actions de signature qui décrivent les actions qui, si omis de l'exécution d'un scénario d'attaque, pourraient empêcher l'attaque de terminer avec succès. Les actions de signature sont exprimées en exerçant une influence sur un modèle d'événement.

Un détecteur AODVSTAT a deux modes de fonctionnement. En mode autonome, un capteur détecte les attaques dans son voisinage immédiat uniquement. En mode distribué, les capteurs échangent périodiquement des messages de mise à jour contenant les détails des nœuds voisins de chaque capteur. Plus précisément, les messages de mise à jour contiennent la liste des paires de MAC/IP connues, le nombre de sauts aux nœuds connus dans le réseau et les informations concernant les attaques locales détectés.

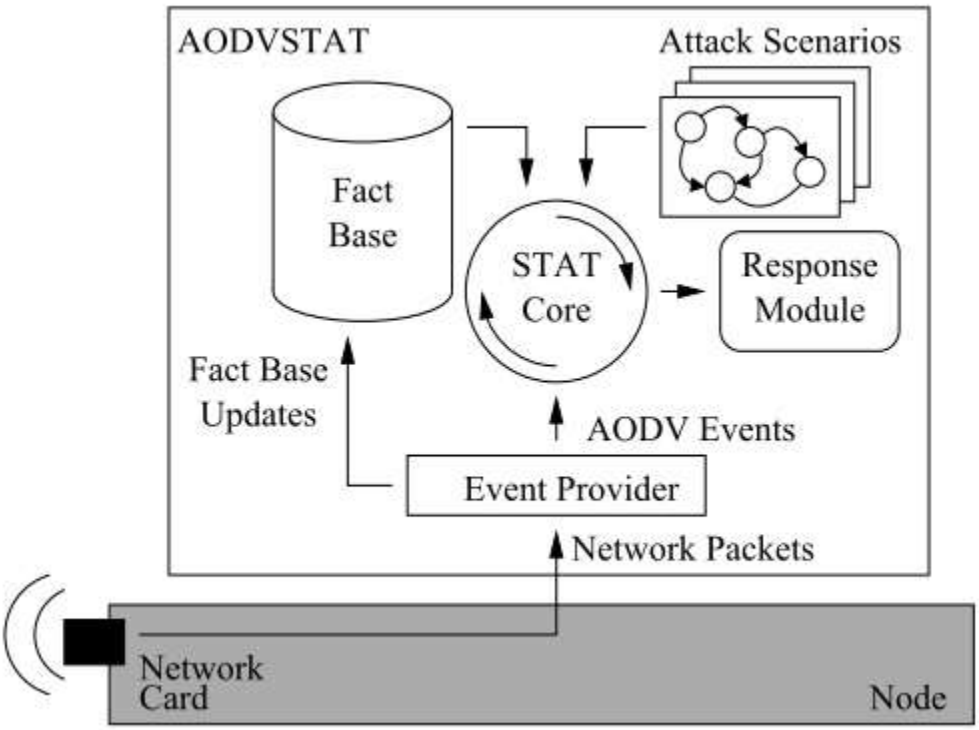


Figure 5.11 : Architecture du Détecteur d'AODVSTAT.

Un capteur AODVSTAT effectue des analyses dynamiques les flux de paquets pour détecter des signes d'intrusion. L'architecture d'un capteur AODVSTAT est illustrée à la Figure 5.11. Le capteur AODVSTAT surveille le réseau environnant à l'aide d'un sniffer de paquets. Les paquets récupérés à partir du réseau sont alors comparées à un certain nombre d'état-transition scénarios d'attaque, chacun décrivant un type spécifique d'attaque. Lorsqu'une correspondance est trouvée, une réponse est initiée, généralement sous la forme d'une alerte de détection d'intrusion.

Ce document passe en revue les diverses questions de sécurité et d'éventuelles attaques dans un réseau sans fil ad hoc et motive la nécessité d'un système de détection d'intrusion. AODVSTAT est un outil de détection d'intrusion qui effectue une détection en temps réel des attaques dans les réseaux mobiles ad hoc exécutant le protocole de routage AODV. Les résultats expérimentaux valident la capacité de AODVSTAT de détecter avec succès à la fois à un saut et de distribuer les attaques contre le protocole de routage AODV (usurpation, abandon de paquets, l'épuisement des ressources attaque, fausses Propagation des numéros de séquence), avec un faible nombre de faux positifs. En outre, l'outil impose une petite surcharge sur les nœuds du réseau, qui est un facteur important pour équipements à ressources restreintes.

7.5 Real-time Intrusion Detection for Ad hoc Networks (RIDAN) [86]

Le système RIDAN [86] est une architecture originale qui a utilisé des techniques de détection d'intrusion à base de connaissance pour découvrir des attaques actives qu'un adversaire peut effectuer sur le tissu de routage des réseaux mobiles ad hoc. En outre, le système est conçu pour prendre des contre-mesures afin de minimiser l'efficacité d'une attaque et maintenir les performances du réseau dans des limites acceptables.

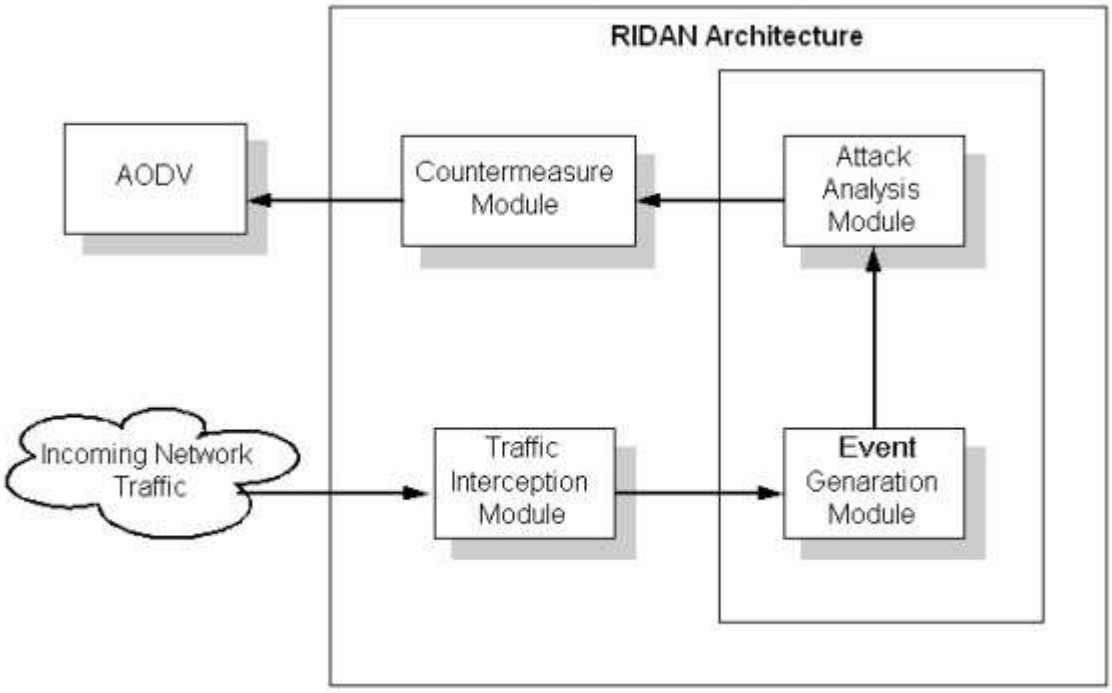


Figure 5.12: Architecture à haut niveau des composantes logiques RIDAN

Dans la figure 5.12 l'architecture de RIDAN est montrée. Le module d'interception de trafic capte le trafic entrant provenant du réseau et choisit qui de ces paquets doivent être traitées ultérieurement. Le module de génération d'événement est responsable d'abréger les informations essentielles requises pour le module d'analyse d'attaque afin de déterminer s'il y a activité malveillante sur le réseau. Le module de génération d'événement et le module d'analyse de l'attaque sont réalisés en utilisant les machines à états finis chronométré. Le dernier volet de l'architecture est le module de contre-mesure qui est chargé de prendre les mesures appropriées pour maintenir les performances du réseau dans les mesures de rendement acceptable. Par conséquent, le composant de détection d'intrusion RIDAN opère entre le trafic de réseau et le protocole de routage ne nécessitant aucune modification au protocole de routage qui est utilisée dans le réseau.

La nouveauté du système réside dans l'utilisation des machines à états finis chronométrées qui permettent la détection en temps réel des attaques actives ; le processus de détection s'appuie sur un état de mauvaise utilisation système de détection. Dans ce cas, chaque nœud doit exécuter l'agent d'IDS.

Ce n'est pas clair dans ce système comment une attaque qui exige plus que les informations d'un bond est découverte.

7.6 Un modèle IDS intégrant différentes techniques (An IDS Model Integrating Different Techniques) [87]

Huang et Lee proposent un modèle ID qui utilise les deux approches de détection basée sur la spécification et axée sur l'anomalie pour détecter les événements intéressants [87]. Un événement (routage) fondamental est défini comme le plus petit ensemble des opérations de routage avec désinvolture connexes tels que la réception/livraison d'un paquet, en modifiant un paramètre de routage. Un événement anormal est défini comme l'événement de base qui ne respecte pas les spécifications du système, telles que la suppression d'une entrée dans la table de routage, modification la suppression d'une entrée dans la table de routage, modification messages de route des messages, etc. Une approche basée sur la spécification

est utilisée pour détecter des événements anormaux qui violent directement les spécifications d'AODV. La détection à base d'Anomalie est utilisée pour découvrir des événements qui ne violent pas de spécifications du protocole de routage et exigent des mesures statistiques.

Dans l'approche axée sur les spécifications automates d'États finis étendus (extended finite state automata) (EFSAs) sont utilisés pour représenter les caractéristiques d'AODV. Les événements qui ne comprennent que des opérations de nœud local sont mis en correspondance avec les transitions de l'automate. Dans les statistiques approche, les caractéristiques sont déterminées à détecter des événements anormaux qui ne peuvent pas être détectées par l'approche basée sur les spécifications, puis un ensemble de règles de détection est généré en utilisant RIPPER classificateur.

L'approche est évaluée à l'aide du simulateur MobiEmu sur certains scénarios (non compris un haut degré de mobilité). Il est démontré que certaines attaques ne sont pas effectivement détectées par cette approche. On en conclut que ces attaques ne peuvent pas être détectées localement. [87].

Les auteurs proposent une taxonomie des attaques qui se décompose d'une attaque dans un certain nombre d'événements de base et également de proposer un modèle pour les détecter. Ils utilisent la détection seulement locale, puisque le nœud local est la seule source de données fiable. Voilà pourquoi il ne peut pas détecter certains types d'attaques qui ne déclenchent pas parce que les événements nécessitant des données provenant d'une autre couche comme l'attaque wormhole ou qu'ils avaient besoin d'autres nœuds tels que network scan [87]. Les auteurs projettent d'enquêter sur la multicouche et la détection globale. Le fait d'extraire des caractéristiques pour détecter automatiquement les attaques inconnues est un autre problème identifié comme les recherches futures.

CONCLUSION

Dans ce chapitre, nous avons passé en revue une présentation globale du protocole de routage AODV ainsi que les nombreuses attaques à son encontre, et enfin les diverses solutions proposées où les auteurs ont donné plusieurs propositions pour la détection et la prévention de ces attaques. Il est à noter que plusieurs travaux ont été fait sur la détection d'intrusion dans les réseaux mobiles ad hoc. Il existe en 2011 sur le site de l'IEEE plus de 1500 publications scientifiques qui traitent du protocole AODV (période de 1997 à 2011), ce qui montre un intérêt certain pour ce protocole et plus généralement sur les protocoles pour réseau Ad hoc **[88]**.

Conclusion générale

Conclusion générale

Les réseaux informatiques basés sur la communication sans fil peuvent être classés en deux catégories : les réseaux avec infrastructure fixe préexistante, et les réseaux sans infrastructure. Dans la première catégorie, le modèle de la communication utilisé est généralement le modèle de la communication cellulaire. Dans ce modèle les unités mobiles sont couvertes par un ensemble de stations de base reliées par un réseau filaire, et qui assurent la connectivité du système. La deuxième catégorie essaie d'étendre les notions de la mobilité à toutes les composantes de l'environnement, toutes les unités du réseau se déplacent librement et aucune administration centralisée n'est disponible. Les réseaux de cette catégorie sont appelés : les réseaux ad hoc.

Etudier la sécurité dans les réseaux ad hoc, sujet d'actualité, était pour nous une occasion de découvrir et d'approfondir nos connaissances à tous les niveaux et bien sûr dans le monde sans fil.

Notre travail a consisté dans un premier temps, à présenter l'environnement ad hoc, ses caractéristiques et ses vulnérabilités. Nous nous sommes particulièrement intéressés à la fonction de routage dans ce type de réseaux pour laquelle nous avons décrit des protocoles de deux types: proactifs et réactifs.

Nous avons par la suite, présenté la sensibilité de ces protocoles aux problèmes de sécurité où nous avons décrit quelques attaques inhérentes au routage dans l'environnement ad hoc que nous avons aussi illustré par des scénarios d'attaques qui peuvent être menés sur le protocole AODV.

Nous sommes ensuite passés, à la présentation de quelques mécanismes qui peuvent être utilisés pour garantir la sécurité du trafic de contrôle échangé entre les nœuds du réseau pour accomplir la fonction de routage. La dernière partie de ce mémoire décrit les différentes IDSs proposés pour pallier au problème de sécurité dans le protocole de routage AODV.

A la fin de ce travail de recherche, nous pouvons dire que la sécurisation des protocoles de routage dans les réseaux ad hoc reste un vrai challenge. Les recherches continuent dans ce domaine afin d'améliorer et d'optimiser de plus en plus les solutions de sécurité existantes afin de rendre les réseaux ad hoc plus fiables, plus performants et plus sécurisés à faible coût pour le grand public

Bibliographie :

[01] : http://www.dicodunet.com/definitions/internet/securite-informatique.htm

[02] : « La sécurité informatique dans la petite entreprise »

Jean-François CARPENTIER © ENI Editions

[03] : http://www.nbs-system.com/blog/introduction-a-la-securite-informatique.html

[04] : http://www.smeb.fr/index.php/ingenierie-informatique-alsace/securite-informatique-reseau-alsace.html

[05] : http://www.secuser.com/faq/securite/

[06] : http://www.privacycommission.be/sites/privacycommission/files/documents/note_securite_des_donnees_a_caractere_personnel.pdf

[07] : http://www.commentcamarche.net/contents/47-introduction-aux-attaques

[08] : http://www.pcastuces.com/pratique/securite/attaques/attaques5.htm

[09] : http://www.authsecu.com/virus-vers-chevaux-de-troie-hoax/virus-vers-chevaux-de-troie-hoax.php

[10] : http://bca.cotesdarmor.fr/html/formation%20droit%20%20et%20securite/2troieetfirewall.htm

[11] : « Sécurité internet, Stratégie et technologie », Solange Ghernaouti-Hélie. Dunod, 2000.

[12] : Sécurité Informatique n°:24 - Sécurité des Systèmes d'Information. Avril 1999 http://dbprog.developpez.com/securite/ids/#LII-C

[13] : « La sécurité des réseaux » Guillaume Desgeorge 2000 http://www.guill.net

[14] : http://coursgratuits.net/securite/ordinateur12.php#

[15] : http://www.vulgarisation-informatique.com/firewall.php

[16] : http://web.mit.edu/rhel-doc/4/RH-DOCS/rhel-sg-fr-4/ch-detection.html

[17] : http://www.cse-cst.gc.ca/its-sti/services/csg-cspc/csg-cspc09l-fra.html

[18] : « NT Réseaux –IDS et IPS », Nicolas Baudoin et Marion Karle, support de cours, Enseignant Etienne Duris 2003-2004.

[19]: Michel Hoffmann. Du "dictionnaire dicodunet "

http://www.dicodunet.com/definitions/internet/securite-informatique.htm

[20]: Valérie Gayraud, Loutfi Nuaymi, Francis Dupont, Sylvain Gombault, and Bruno Tharon, « La Sécurité dans les Réseaux Sans Fil Ad Hoc ».

[21]: Riahla Med Amine, « Conception et mise en œuvre d'un nouveau protocole de routage Multi chemins sécurisé pour les réseaux ad hoc basé sur les colonies de fourmis », Thèse de magister, université de Boumerdes, 2008

[22]: Arnaud Jacques, Véronique Sainson, Mustapha Benjada « Le grand livre de la sécurité ». Livre Edition sécurité informatique. http://www.securiteinfi.com.2004. Page 75-90

[23]: Imed Allal, Mehdi Akou, « Sécurité des réseaux mobile ». Thèse d'ingénieur d'état en informatique, ESI, Ecole Nationale Supérieure d'Informatique de Oued-Smar, Alger. 2009/2010

[24]: Abdesselem BEGHRICHE. "De la Sécurité à la E-Confiance basée sur la Cryptographie à Seuil dans les Réseaux sans fil Ad hoc ".Université de L'Hadj Lakhdar-Batna, 2009.

[25]: Samuel Galice. « Modèle de sécurité dynamique pour les réseaux spontanés ». Thèse de doctorat. Institut National des Sciences Appliquées de Lyon. 2007.

[26]: A. Menezes, P. Van Oorschot, S. Vanstone. "Handbook of Applied Cryptography". Presse CRC, 1996.

[27]: Aliouane Lynda, Nadjib Badache " L'Authentification dans les Réseaux Ad Hoc", CERIST. RIST Vol, 16 n°01. 2006.

[28]: Étude technique réalisée par CGI, " Étude technique Cryptographie à clé publique et signature numérique Principes de fonctionnement ". Septembre 2002.

[29]: Pietro Michiardi, "Coopération dans les réseaux ad hoc : Application de la théorie des jeux et de l'évolution dans le cadre d'observabilité imparfaite". 2006.

[30]: Thibault Cholez, Isabelle Chrisment et Olivier Festor. "Un mécanisme de révocation orienté services pour les réseaux P2P ". 2008.

[31]: Chris Karlof, David Wagner, «Secure routing in wireless sensor networks: attacks and countermeasures», Ad Hoc Networks 1(2003) 293–315, 2003

[32]: Yingshu Li, My T. Thai, Weili Wu, «Wireless Sensor Networks and Applications», Springer Science+Business Media LLC, 2008

[33]: Mohammad Ilyas, Imad Mahgoub, « Handbook of Sensor Networks: Compact Wireless and Wired Sensing Systems», CRC Press LLC, 2005

[34]: Céline Burgod, «Contribution à la sécurisation du routage dans les réseaux ad hoc ». Thèse de doctorat, Spécialité informatique, Université de LIMOGES, le 12 octobre 2009

[35]: Manel Guerrero Zapata and N. Asokan. Securing ad hoc routing protocols. InW. Douglas Maughan and Nitin H. Vaidya, editors, Workshop on Wireless Security, pages 1-10. ACM, 2002.

[36]: Yih-chun Hu ,David B. Johnson, Adrian Perrig «SEAD: Secure Efficient Distance Vector Routing for Mobile Wireless Ad Hoc Networks». Article de recherche publié dans Mobile Computing Systems and Applications, 2002. Proceedings Fourth IEEE Workshop 2002. PAGES 3 – 13

[37]: Panagiotis Papadimitratos and Zygmunt J. Haas. Secure link state routing for mobile ad hoc networks. In SAINT Workshops, pages 379–383. IEEE Computer Society, 2003.

[38]: Sergio Marti, T.J Giuli, Kevin Lai and Mary Baker. « Mitigating Routing Misbehaviour in Mobile Ad Hoc Networks ». International Conference on Mobile Computing and Networking. Proceedings of the 6th annual international conference on Mobile computing and networking. Boston, Massachusetts, United States. Year of Publication: 2000.

[39]: Sonja Buchegger and Jean-Yves Le Boudec. Performance analysis of the condant protocol. In MobiHoc, pages 226{236. ACM, 2002.

[40]: S. Axelsson: Aspect of the modeling and performance of intrusion detection. Technical Report319L, Departement of Computer Engineering Chql;ers University of technologym2000.

[41] : Philipe Biondi : « Architecture expérimentale pour la détection d'intrusions dans un système informatique ». Avril-Septembre 2001

[42] : Bourkache Ghenima, « Un IDS réparti basé sur une société d'agents intelligents », Thèse de magister Université M'hamed Bouguara de BOUMERDES, 2007

[43] : Eric Maiwald, «Sécurité des réseaux», Edition CampusPress, 2001

[44]: Rebecca Bace and Peter Mell. Intrusion Detection Systems. NIST Special Publication on Intrusion Detection Systems.

[45] : Nicolas Baudoin et Marion Karle, « NT Réseaux : IDS et IPS ». Ingenious 2000. 2003-2004.

[46] : Hervé Debar, M.Dacier et A.Wespi, « A revised taxonomy for intrusion detection systems ». Annales des télécommunications. July-August 2000.

[47] : Alexandre Nevski, « Modélisation et la simulation d'un système de détection et de réponse aux intrusions », Laboratoire Telecom de Centre Universitaire Informatique de Genève.

[48] : R.Mouzer et R.kheddam, « Mise en place d'un système de détection et de prévention d'intrusions », thèse de fin d'étude Université Mouloud Mammeri, 2009.

[49] : N. Ben Amor, S. Benferhat et Z. Elouedi, « Réseaux bayésiens naïfs et arbres de décision dans les systèmes de détection d'intrusions ».

[50]: Valdes, A., Skinner K.: Adaptive Model-based Monitoring for Cyber Attack Detection. In proceedings of Recent Advances in Intrusion Detection (RAID 2000), Toulouse, France, 80-92, 2000.

[51]: Quinlan, J. R.: Induction of decision trees. Machine Learning 1, 1-106, 1986.

[52]: Quinlan, J. R.: C4.5, Programs for machine learning. Morgan Kaufmann San Mateo Ca, 1993.

[53]: http://www.chimique.usherbrooke.ca/cours/gch445/neurones-intro.html

[54]: Karima Boudaoud. (2002) « Détection d'intrusions : Une nouvelle approche par système multi agents ». Thèse de doctorat de l'Université de Genève. 2002.

[55] : Introduction et Initiation à la sécurité informatique. « SecuriteInfo.com ».

[56] : A.Phillip, Porras et Alfonso Valdes, « Live traffic analysis of tcp/ip getaways ». Proc. ISOC Symposium on Network and Distributed System Security (NDSS98). (San Diego, CA, March, 98), Internet Society.

[57]: J. Anderson, « Computer security threat monitoring and surveillance ». 1980

[58]:S. Martino, « A mobile agent approach to intrusion detection ». Joint Research Centre-Institute for Systems, Informatics and Safety, Italy, June 1999.

[59]: BOUKHECHEM Nadhir "ROUTAGE DANS LES RESEAUX MOBILES AD HOC PAR UNE APPROCHE A BASE D'AGENTS ". Mémoire magistère 2008

[60]: C. E. Perkins, P. Bhagwat, “Highly dynamic destination-sequenced distance-vector routing (DSDV) for mobile computer”, ACM SIGCOMM-94, pp. 234-244, 1994.

[61]: G. Pei, M. Gerla & T.-W. Chen – « Fisheye state routing: A routing scheme for ad hoc wireless networks », dans Proc. IEEE Int’l Conf. on Communications, (ICC 2000) (New Orleans, LA, USA), vol. 1, 2000, p. 70–74.

[62]: C. E. Perkins, E. M. Royer, Samir R. Das, "Ad-hoc on demand distance Vector (AODV) routing", IETF, Intern- Draft, draft-ietf-manet-aodv-05.txt, March 2000.

[63]: David B.Johnson, A. Maltz, and Josh Broch, ”The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks”, Kluwer Academic Publishers, pp. 153-181, 1996.

[64]: M.Jiang, J. Li, Y.C. Tay, "Cluster Based Routing Protocol", IETF Draft, 1999. http://www.ietf.org/internet- drafts/draft-ietf-manet-cbrp-spec-01.txt.

[65]: K. Sanzgiri, B.t Dahill, B.N. Levine, C. Shields, E. M. Belding-Royer «Authenticated routing for ad hoc networks». 10 th IEEE International Conference on Network Protocols. ICNP 2002. Paris, France. DBLP, http://dblp.uni-trier.de.

[66]: C. E. Perkins, J. T. Malinen, R. Wakikawa, A. Nilsson, and A. J. Tuominen, “Internet connectivity for mobile ad hoc networks,” J. Wireless Commun. Mobile Comput., vol. 2, no. 5, pp. 465–482, Aug. 2002.

[67]: S. Corson and J. Macker, “Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations,” IETF RFC 2501, Jan. 1999.

[68]: N. Tebbane, S.Tebbane, A. Mehaoua, “Simulation et Mesure des performances du protocole de routage AODV,” JTEA'2004, Hamamet, Tunisia 2004.

[69]: mémoire magistère <Protocole pour la sécurité des réseaux sans fil peer to peer> Houda Hafi

[70]: Mohamed Ali Ayachi, « Contributions à la détection des comportements malhonnêtes dans les réseaux ad hoc AODV par analyse de la confiance implicite », thèse de doctorat, université de Rennes 1, 2011.

[71]: International Journal of Modern Engineering Research (IJMER) www.ijmer.com Vol.2, Issue.3, May-June 2012 pp-728-732

[72]: Ad-hoc On-Demand Distance Vector Protocol and Black Hole Attack in AODV By: Rajkumar Singh

[73]: Maha Abdelhaq et al, “Security Routing Mechanism for Black Hole Attack over AODV MANET Routing Protocol”, Australian Journal of Basic and Applied Sciences, 5(10): 1137-1145, 2011 University Kebangsaan Malaysia, 2011.

[74]: K.LakSBmi et al, “Modified AODV Protocol against Blackhole Attacks in MANET”, International Journal of Engineering and Technology Vol.2 (6), 2010, 444-449

[75]: An Intrusion Detection Tool for AODV-based Ad hoc Wireless Networks. Giovanni Vigna, Sumit Gwalani, Kavitha Srinivasan, Elizabeth M. Belding-Royer, Richard A. Kemmerer

[76]: Deng H., Li W. and Agrawal, D.P., "Routing security in wireless ad hoc networks" Communications Magazine, IEEE, October 2002.

[77]: Al-Shurman, M., Yoo, S. and Park, S, "Black hole Attack in Mobile Ad Hoc Networks", ACM Southeast Regional Conference, 2004.

[78]: Tamilselvan, L.; Sankaranarayanan, V., "Prevention of Blackhole Attack in MANET," The 2nd International Conference on Wireless Broadband and Ultra Wideband Communications, Aus Wireless 2007.

[79]: Lalit Himral, Vishal Vig, Nagesh Chand, "Preventing AODV Routing Protocol from Black Hole Attack" International Journal of Engineering Science and Technology (IJEST), May 2011.

[80]: Payal N. Raj1 and Prashant B. Swadas2, "DPRAODV: A Dynamic Learning System against Blackhole Attack in AODV based MANET", IJCSI International Journal of Computer Science Issues, 2009.

[81]: Subash Chandra Mandhata, Dr.Surya Narayan Patro, "A counter measure to Black hole attack on AODV- based Mobile Ad-Hoc Networks" International Journal of Computer & Communication Technology (IJCCT), 2011.

[82]: Chin-Yang Tseng, Poornima Balasubramanyam, Calvin Ko, Rattapon Limprasittiporn, Jeff Rowe, Karl Levitt, "A Specification-based Intrusion Detection System for AODV". Computer Security Laboratory University of California, Davis

[83]: Sonali Bhargava, Dharma P. Agrawal "Security enhancements in AODV protocol for Wireless and hoc networks". Dpt of ECECS, University of Cincinnati

[84]: Y. Zhang, W. Lee and Y. Huang, Intrusion detection techniques for mobile wireless networks, ACM Wireless Networks, 9(5) (2003), 545-556.

[85]: Giovanni Vigna, Sumit Gwalani, Kavitha Srinivasan, Elizabeth M. Belding-Royer, Richard A. Kemmerer; "An Intrusion Detection Tool for AODV-based Ad hoc Wireless Networks". Department of Computer Science University of California, Santa Barbara

[86]: Ioanna Stamouli "Real-time Intrusion Detection for Ad hoc Networks". A dissertation submitted to the University of Dublin, in partial fulfilment of the requirements for the degree of Master of Science in Computer Science. 2003

[87]: Y. Huang, W. Fan, W. Lee and P. Yu, Cross-feature analysis for detecting ad hoc routing anomalies, in Proceedings of the 23rd International Conference on Distributed Computing Systems (ICDCS 03), (2003), 478-487.

[88]: http://fr.wikipedia.org/wiki/Ad-hoc_On-demand_Distance_Vector