KITSCHMINYOF Stephan
GALLET Benjamin

M2-INFO

INFO002 : Rapport

La cryptologie au sein des communications sans fil

Sommaire

Introduction

Objectif

Ce rapport a pour objectif d'étudier et de comprendre quels sont les protocoles cryptographiques qui sont mis en place au sein des communications sans fil.

Définition

Une communication sans fil peut être définie comme un mode qui permet la transmission de données via des ondes radio. La distance parcourue par ces ondes peut être plus ou moins élevée en fonction de la bande de fréquence utilisée.

Ces ondes sont émises par un périphérique (ordinateur, téléphone ou tout objet permettant l'envoi d'ondes électromagnétiques) et vont être captées par un autre périphérique qui pourra traiter les données contenues dans les ondes.

Contenu

Comme le sujet des communications sans fil est très large, nous avons choisi de nous concentrer sur 3 types de communications.

Dans une première partie, nous allons étudier le protocole Wifi. Nous ferons un point sur son fonctionnement puis nous ferons un historique des différents systèmes de sécurité qui ont été mis en place au cours du temps.
Dans une seconde partie nous détaillerons les caractéristiques du Bluetooth.
Enfin, la dernière partie sera dédiée à la technologie RFID. Nous nous intéresserons plus particulièrement à la branche NFC et à son utilisation dans le milieu bancaire.

La sécurité au sein des réseaux wifi

Contexte

Les réseaux wifi utilisent aujourd'hui le protocole 802.11.

Il fait partie d'un ensemble de normes qui ont commencé à être développées en 1997, ce sont les normes 802.x.

Ces normes sont utilisées comme base par les constructeurs de matériels servant à implémenter des infrastructures réseaux pour les liaisons aussi bien filaires que sans fil.

La spécification 802.11 concerne plus particulièrement les réseaux sans fil.

Les différentes étapes de sécurisation

WEP

Le WEP est un protocole qui permet de chiffrer des trames utilisant le protocole 802.11. Il utilise l'algorithme symétrique RC4. C'est-à-dire que la même clé permet de chiffrer et déchiffrer un message en le repassant dans l'algorithme. RC4 utilise des clés de 64 ou 128 bits.

Dans un premier temps, il faut définir la clé secrète qui sera utilisée pour le chiffrement et le déchiffrement. La clé va permettre de générer un nombre pseudo aléatoire qui aura la longueur de la trame à transmettre.
Ce nombre aléatoire, concaténé à la clé secrète, permet de chiffrer les données en faisant un XOR avec la trame en clair.
La clé sert à créer un nombre pseudo-aléatoire d'une longueur égale à la longueur de la trame.

On sait aujourd'hui que le protocole WEP n'est pas sûr. La clé est partagée par tous les membres du réseau donc, sur un réseau très étendu, beaucoup de machines connaissent la clé et la sécurité est alors plus facilement compromise car le fait de connaître la clé permet de déchiffrer l'ensemble des communications du réseau.

La sécurité pose aussi problème sur la longueur des clés. 24 bits de la clé servent pour l'initialisation du générateur pseudo-aléatoire. Ensuite, le reste des bits (40 ou 104), en fonction de la taille de la clé, permettent de chiffrer. Ce système est perméable aux attaques par brute force car la taille des clés en est réduite et le temps pour tester l'ensemble des clés possible est aussi fortement réduit.
Une faille a aussi été trouvée et permet de trouver la clé du réseau en créant entre 100Mo et 1Go de trafic sur le réseau. Cette faille est liée au nombre généré par le générateur aléatoire. Les premier octets des messages chiffrés ne sont pas vraiment aléatoires et en les étudiant, nous pouvons retrouver la clé initiale et donc déchiffrer l'ensemble des données échangées sur le réseau.
Le protocole WEP ne permet donc pas d'assurer une sécurité efficace sur les données.

WPA, WPA2 et WPA3

Le protocole WPA est apparu pour remplacer le protocole WEP. La base de ce protocole visait à améliorer la gestion des clés de chiffrement ainsi que l'authentification des utilisateurs du réseau en ajoutant de la sécurité.
WPA utilise TKIP (Temporal Key Integrity Protocol) pour le chiffrement des données. TKIP est un protocole similaire à WEP. Par exemple, il utilise aussi RC4 mais réduit les erreurs du protocole WEP pour éviter de compromettre la sécurité.
WEP chiffre les paquets avec la clé de base concaténée avec un chiffre pseudo-aléatoire généré pour correspondre à la longueur de la trame qui change à chaque trame. WPA lui va chiffrer de la même manière mais avec la clé de base qui va être régulièrement modifiée. Le chiffre aléatoire va aussi être haché pour éviter de l'envoyer en clair comme le fait WEP.

Les clés de chiffrement font 256 bits ce qui améliore nettement la sécurité face aux attaques par brute force

Tout comme WEP, WPA utilise le chiffrement par flot. C'est-à-dire que le chiffrement se fait bit à bit et qu'il est effectué au fur et à mesure que les données à chiffrer arrivent.

WPA2 est une version mise à jour de WPA. Le protocole TKIP est remplacé par CCMP.
CCMP améliore encore la sécurité en utilisant par exemple le protocole AES pour le chiffrement qui est considéré comme sûr, contrairement à RC4. Le chiffrement passe donc de chiffrement par flot à chiffrement par bloc.

WPA2 est vulnérable aux attaques de réinstallation de clé. Cette attaque exploite une faiblesse dans WPA2, qui permet aux attaquants de créer un réseau clone et force la victime à s'y connecter.
Les attaquants vont alors pouvoir déchiffrer des données. Cela leur permettra de cracker la clé de chiffrement du réseau normal.

WPA3 est ,quant à lui, en cours de déploiement depuis 2018. Il s'agit d'une version qui corrige les problèmes découverts sur WPA2. Il utilise le même système de chiffrement par bloc que son prédécesseur.

La question de la sécurité pour les protocoles bluetooth

Fonctionnement du protocole bluetooth

Le bluetooth est un standard de communication qui a été créé en 1994 par l'entreprise Ericsson.
Ce protocole utilise des Ultras Hautes Fréquences (UHF) pour son système de communication. Les UHF désignent les ondes qui ont une bande comprise entre 300 Mhz et 3 Ghz. Le bluetooth utilise lui la même fréquence que la Wifi (2.4Ghz) mais n'as pas été conçu pour répondre à la même problématique et en est donc très différent sous bien des aspects.
Ce protocole permet la communication sans fil entre différents périphériques de manière bidirectionnelle. C'est-à-dire que les appareils connectés entre eux peuvent

s’envoyer et recevoir des fichiers via le même canal de communication (la même liaison).
Aujourd’hui, avec la dernière version du protocole bluetooth, la portée théorique maximale s’élève à 350 mètres et chaque paquet envoyé/reçu peut faire jusqu’à 255 octets.

Les sécurités issues de la norme Bluetooth

Pour chiffrer ses communications, le bluetooth utilise E0. Il s’agit d’un algorithme de chiffrement par flot. C’est-à-dire qu’il chiffre les données au fur et à mesure qu’il les reçoit.
Il n’est pas nécessaire d’avoir l’intégralité des données qu’il faut chiffrer avant de commencer à les chiffrer. En effet, les opérations se font bit à bit, ce qui permet d’être relativement efficace et rapide dans les communications en temps réel en chiffrant directement les données quand elles sont reçues.
L’algorithme E0 permet de ne pas avoir de redondance des messages chiffrés. Un même message qui sera traité par l’algorithme deux fois ne sera pas chiffré de la même manière.
Pour chiffrer un message, l’algorithme va utiliser un générateur pseudo-aléatoire qui va générer une clé. Il va ensuite effectuer un XOR entre le message en clair et la clé générée. Le résultat du XOR sera alors notre message chiffré.

Exemple

Imaginons :

- Notre message en clair **m** = 1010 1010 1010
- Notre clé de chiffrement générée par le générateur pseudo-aléatoire **c** = 101 0110 1101

Alors :

- Notre message chiffré **mc** = **m** x **c.**

Si nous effectuons le XOR bit à bit de **m** et **c**, nous obtenons **mc** = 0111 1100 0111

Les failles du Bluetooth

Le bluetooth reste un moyen de communication très vulnérable à cause des outils qu'il utilise pour assurer la sécurité de ses communications ainsi que de ses caractéristiques.
Nous pouvons lister quelques failles qui sont présentes sur différentes versions du protocole bluetooth :

- Les clés utilisées pour l'authentification des appareils pour une session sont réutilisables, ce qui est déjà une faille en soit mais à la fin d'une session, ces clés sont rendues publiques donc utilisables par un attaquant.
- Si une session entre 2 appareils dure plus de 23h59, la séquence qui permet de chiffrer se répète car elle repose en partie sur l'horloge qui reviendra à son point de départ et générera donc de nouveau la même séquence qu'au début de la session.
- Le nombre d'authentifications avec un appareil n'est pas limité donc les attaquants peuvent potentiellement, via la technique du brute force (test de toutes les possibilités), se connecter à n'importe quel appareil, que la connexion soit voulue ou non.
- Le générateur pseudo-aléatoire utilisé par l'algorithme E0 n'est pas considéré comme robuste. C'est-à-dire qu'il est éventuellement possible, avec un jeu de données et une certaine méthode, de prédire les prochains bits qui seront générés. Ce qui altère énormément la sécurité des communications car possiblement déchiffrables.

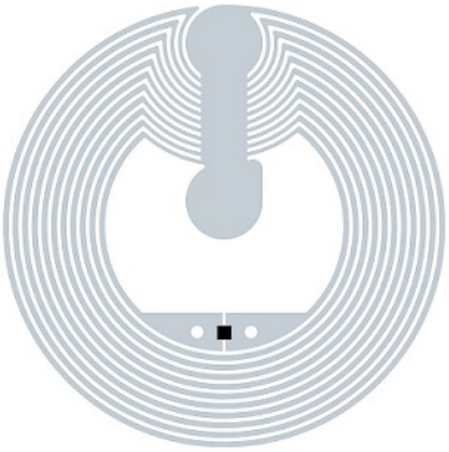
Cela fait du protocole bluetooth, un protocole partiellement sécurisé car les communications sont effectivement chiffrées mais que certaines failles peuvent être exploitées afin de grandement réduire cette sécurité et faciliter l'intrusion des attaquants dans le système.

La norme NFC

Utilité et fonctionnement

La technologie NFC (pour Near Field Communication) est un dérivé de la technologie RFID. Elle permet l'échange de données entre deux objets s'ils sont très proches. Cette technologie est uniquement faite pour le transfert de petites quantités de données car le débit n'atteindra pas des vitesses supérieures à 400 kbit/s.
La technologie NFC peut se comporter en 3 modes différents :

- Le mode peer to peer : C'est le mode utilisé lorsque l'on veut transférer des données (photos, vidéos, …) d'un périphérique à un autre.
- Le mode terminal : Il s'agit d'un mode qui permet à l'appareil de lire les tags NFC qui sont très proches de lui. C'est le mode qu'utilisent les terminaux de paiement sans contact par exemple.
- Le mode émulation de carte : comme son nom l'indique, il permet l'émulation de carte qui utilisent le sans contact (carte bancaire, d'abonnement, …).



Nous pouvons voir ici la représentation d'un tag NFC. Tous les tags NFC ne ressemblent pas à celui-ci mais dans l'ensemble, la structure reste la même.

Nous pouvons remarquer sur le tour (en gris), l'antenne qui va permettre la communication avec les appareils qui scannent les tags. En noir, en peut voir la partie qui contient la mémoire ainsi qu'un microcontrolleur qui permet de gérer les données stockées sur le tag et la transmission/réception des messages.

Au niveau du fonctionnement, quand nous allons approcher notre tag d'un lecteur, le lecteur va émettre de l'énergie par induction. L'antenne va, grâce à cette énergie, alimenter le reste du circuit (microprocesseur et mémoire).
Ensuite le microcontrôleur va faire émettre, via l'antenne, le contenu de sa mémoire à intervalle régulier tant qu'il est alimenté.
Enfin le lecteur va capter le signal et va en extraire puis traiter les données selon son utilisation.
Lorsqu'on éloigne le tag du lecteur, il n'est plus alimenté et cesse donc d'émettre.

Si nous voulons écrire des données dans le tag, le fonctionnement sera similaire hormis le fait que le terminal va lui envoyer les données à écrire. Le tag va ensuite pouvoir mettre les données reçues dans sa mémoire.

La sécurité dans le domaine bancaire

Dans cette partie nous allons nous concentrer sur la partie bancaire dédiée au NFC avec les cartes bancaires et le paiement sans contact. Nativement, la technologie NFC n'embarque pas de sécurité ce qui la rend très vulnérable aux attaques. Cependant certaines structures ont mis en place des systèmes cryptographiques qui permettent au paiement sans contact d'être beaucoup plus robuste aux attaques.

Les premières cartes à introduire un aspect de sécurité sont les cartes MIFARE Classic. Elles utilisent l'algorithme Crypto-1 qui est un algorithme de chiffrement par flux spécialement conçu pour les puces RFID en 1994 par l'entreprise NXP.
L'algorithme Crypto-1 inclut :

- Un registre à décalage de 48 bits qui permet de conserver l'état du chiffrement
- Une fonction non linéaire qui sert à générer la clé de chiffrement.

- Un registre à décalage à rétroaction linéaire de 16 bits qui est utilisé au moment de l'authentification de la carte.

Cet algorithme fonctionne sur le principe de sécurité par l'obscurité, c'est-à-dire qu'aucune information n'à été divulguée sur le fonctionnement de cet algorithme, ce qui permet d'assurer une certaine sécurité face aux attaquants. Mais cette sécurité supplémentaire n'a fonctionné que pendant un temps car Crypto-1 est obsolète depuis 2009. En effet, une méthode a permis d'inverser le chiffrement.
L'algorithme n'est donc plus sûr et n'est plus apte à être utilisé dans ces conditions.

Afin de résoudre les problèmes liés aux puces MIFARE Classic, les constructeurs ont développé la puce MIFARE DESFire. Ces puces reposent sur les algorithmes AES et 3DES qui sont des algorithmes de chiffrement symétrique et que nous allons expliquer.

AES

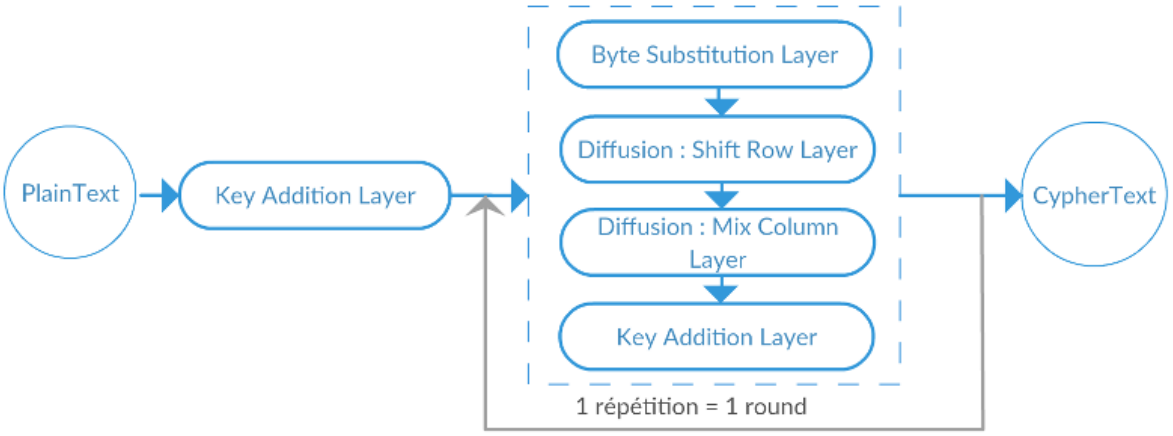
AES est un algorithme de chiffrement symétrique. C'est-à-dire que nous allons utiliser la même clé pour chiffrer et déchiffrer un message.
AES utilise du chiffrement par bloc. L'algorithme va découper le message à chiffrer/déchiffrer en bloc. Chaque bloc a une taille fixe qui est de 16 octets.

Le principe de l'algorithme :
L'algorithme prend en entrée le texte non chiffré (plaintext) et donne en sortie le texte chiffré (cyphertext).

Le plaintext sera découpé en blocs pour pouvoir être chiffré.
Chaque bloc donné en entrée sera chiffré en un bloc de même taille en sortie.
Un bloc sera donc une suite de 16 octets, par exemple le bloc 0 sera représenté comme ci-dessous (en remplaçant les On par l'octet correspondant du message en hexadécimal) :
B0 = O1 O2 O3 O4 O5 O6 O7 O8 O9 O10 O11 O12 O13 O14 O15 O16

Pour le chiffrement, il faut aussi choisir une clé de chiffrement. Cette clé correspond à une suite de bits et fait généralement 128 ou 256 bits. La clé permet d'une part de chiffrer/déchiffrer le message mais elle permet aussi de connaître le nombre de fois où le message devra passer dans l'algorithme avant d'être chiffré/déchiffré. Par exemple avec une clé de 128 bits, le message devra tourner 10 fois dans l'algorithme alors qu'avec une clé de 256 bits, ce sera 14 fois.

Pour chiffrer un bloc, l'algorithme va passer par 4 étapes. Il va ensuite répéter ces étapes autant de fois que la clé de chiffrement l'indique



- Byte Substitution

Le but de cette fonction est d'effectuer une substitution octet par octet

- Shift Row

Le but est d'introduire de la confusion dans les données en modifiant l'ordre des bits. Pour cela, un simple décalage (Shift) est fait sur chaque ligne (Row) du bloc de données.

- Mix Column

Son but est de faire intervenir de multiples bits dans le codage d'un seul.
Cela permet de rapidement lier tous les éléments d'un bloc entre eux, ce qui rend le déchiffrement plus complexe si l'on ne connaît pas la clé et améliore la sécurité de l'algorithme face aux attaques.

- Key Addition

Les clés qui sont utilisées par AES font 128 ou 256 bits. Comme énoncé précédemment, les clés permettent de savoir combien de répétitions doit faire l'algorithme pour chiffrer/déchiffrer un message.
Pour éviter certaines attaques, des clés intermédiaires sont créées pour chaque tour de l'algorithme. Ces clés sont issue de la clé initiale
Cela permet d'avoir une clé complexe lors du chiffrement et la sécurité est ainsi améliorée.
Le but de AddKey est de sommer le bloc de données courant avec la clé du tour courant.

Ces 4 étapes sont réversibles, c'est pourquoi nous pouvons déchiffrer le texte en le faisant passer dans l'algorithme une seconde fois avec la même clé qui a servi à chiffrer le message.

AES est aujourd'hui encore un algorithme de chiffrement sûr car il n'a pas encore été trouvé de moyen permettant de casser cet algorithme. La méthode la plus efficace connue à ce jour étant l'attaque par brute force.

Triple DES

DES est un algorithme de chiffrement symétrique, le prédécesseur d'AES. Il utilise un chiffrement par bloc et chaque bloc fait 8 octets. Un des octets permet de vérifier l'intégrité de la clé et ne contient donc aucun chiffrement de données.

De la même manière que l'algorithme AES, DES va appliquer des fonctions sur les blocs de données pour les chiffrer et répéter ces fonctions un certain nombre de fois.

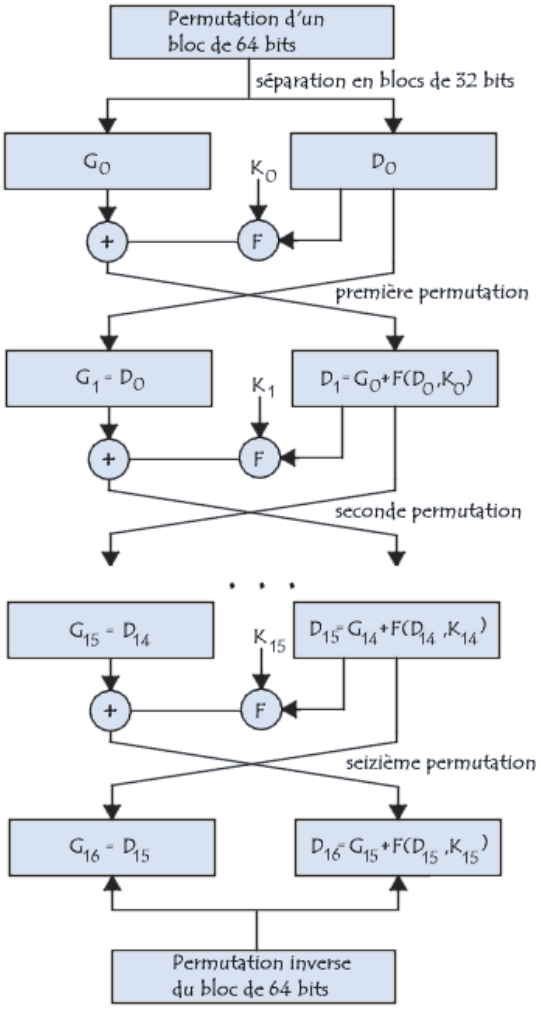


Schéma de fonctionnement DES

Dans une première étape, l'algorithme va récupérer les données en entrée afin de les découper en bloc de 64 bits. Une première permutation est alors effectuée selon une matrice définie. Chaque bit du bloc de base va alors se retrouver à la position qui lui est affectée dans la matrice de permutation. (Si nous prenons un exemple, le bit 58 du bloc se retrouvera en position 1)

Initial Permutation							
58	50	42	34	26	18	10	02
60	52	44	36	28	20	12	04
62	54	46	38	30	22	14	06
64	56	48	40	32	24	16	08
57	49	41	33	25	17	09	01
59	51	43	35	27	19	11	03
61	53	45	37	29	21	13	05
63	55	47	39	31	23	15	07

Table de permutation algorithme DES

Ensuite, chaque bloc va être coupé en deux, puis l'étape de chiffrement va commencer.

- Un XOR va être effectué avec la clé de chiffrement sur le demi-bloc de droite
- Il va y avoir une étape de substitution :
    - Le demi-bloc de droite va être découpé en blocs de 6 bits. Il va donc y avoir 8 blocs de 6 bits. Chaque bloc va être substitué par une valeur codée sur 4 bits. Ces valeurs seront prises dans une table de substitution. Le premier et le dernier bit du bloc permettront de désigner la ligne sur laquelle prendre la valeur et les bits restants permettent de trouver la colonne correspondant à la valeur.
    - Chacune des substitutions sur 4 bits sont regroupées pour former un bloc de 32 bits.
- Vient ensuite une étape de permutation de chacun des 32 bits du demi-bloc de droite.

16	7	20	21	29	12	28	17
1	15	23	26	5	18	31	10
2	8	24	14	32	27	3	9
19	13	30	6	22	11	4	25

Table de permutation du bloc droit DES

- Enfin l'algorithme va effectuer un XOR sur le demi-bloc de gauche et de droite.
- Le bloc de gauche va prendre la valeur initiale du bloc de droite.

On recommence ensuite à l'étape 1, 16 fois.

Une fois les 16 itérations effectuées, on va regrouper le demi-bloc de droite et celui de gauche et nous allons faire une dernière permutation qui correspond à la permutation inverse à celle de départ.

Final Permutation							
40	08	48	16	56	24	64	32
39	07	47	15	55	23	63	31
38	06	46	14	54	22	62	30
37	05	45	13	53	21	61	29
36	04	44	12	52	20	60	28
35	03	43	11	51	19	59	27
34	02	42	10	50	18	58	26
33	01	41	09	49	17	57	25

Table de permutation finale DES

Nous obtenons alors la version chiffrée de notre bloc sur 64 bits.

Le principe de Triple DES (ou TDES) est d'enchainer 3 fois l'algorithme DES pour chiffrer un message.
Cela permet d'augmenter significativement la sécurité du chiffrement. L'inconvénient est qu'il faudra 3 fois plus de temps pour chiffrer/déchiffrer un message.

Conclusion

Nous avons pu voir au travers de ce rapport que la cryptologie n'est pas mise en place de la même manière sur les différentes technologies sans fils.
Cela s'explique principalement par le fait que tous les types de communications sans fil n'ont pas besoin de la même dose de sécurité pour protéger les utilisateurs et les données.

Les réseaux wifi ont besoin de beaucoup de sécurité et développent pour cela des moyens d'assurer le chiffrement des données. Les protocoles de sécurité évoluent en même temps que les failles sont trouvées pour rester les plus sécurisés possibles en utilisant des algorithmes sûrs.
Les appareils utilisant par exemple le nfc, n'ont pas vraiment besoin de sécurité car la portée des équipements est si restreinte (quelques centimètres) que les attaquants n'ont que très peu de manières de pirater le système. A la place du chiffrement, des sécurités sont mises en place en cas de problèmes (limite de montant, limite de transactions en sans contact par exemple).

Il existe aussi des cas comme le bluetooth où les communications sont bien chiffrées mais certaines failles existent et ne sont pas corrigées. Le protocole est donc vulnérable aux attaques et le sera encore pendant longtemps.

Nous pouvons donc conclure que la mise en place de protocoles et procédés cryptographiques dans les communications sans fils se font en fonction des cas et des besoins des moyens de communication. Si la sécurité n'est pas importante dans l'utilisation qui en sera faite, aucune sécurité ne sera mise en place. En revanche, si un besoin de sécurité est demandé par les utilisateurs, des contre mesures seront mises en place pour pallier le manque de sécurité.