Détection d'intrusion dans les réseaux 802.11

Laurent Butti
Expert senior en sécurité des réseaux

France Télécom R&D
Laboratoire Sécurité des Services et Réseaux
38-40 Rue du Général Leclerc
92794 Issy-les-Moulineaux Cedex 9 - France
firstname.lastname@francetelecom.com

Résumé Les réseaux locaux radioélectriques sans-fil 802.11 sont sous les feux des projecteurs depuis de nombreuses années [1]. Les attaques 802.11 actuellement réalisables peuvent avoir des impacts sécurité critiques sur tout système d'information. Cet article présente les méthodes les plus pertinentes pour détecter et qualifier la majorité des attaques. Enfin l'article expose notre retour d'expérience dans le développement et le déploiement d'une solution de détection d'intrusion 802.11 à France Télécom R&D.

1 Introduction

Les réseaux radioélectriques 802.11, plus connus sous l'appellation Wi-Fi[1], font maintenant partie de notre paysage informatique. La prolifération des technologies 802.11 aussi bien au niveau de l'infrastructure (points d'accès) qu'au niveau des postes terminaux (ordinateurs portables[2]) n'est plus à démontrer ! De nombreuses entreprises utilisent ces technologies pour des raisons d'ergonomie (facilité d'accès accrue), pour des raisons de coûts de déploiement (par rapport à un câblage) ou encore par phénomène de mode. Malheureusement cette démocratisation n'est pas sans effets de bord sur la sécurité... Toute arrivée d'une technologie radioélectrique n'est pas anodine et doit être considérée avec rigueur, en particulier en milieu entreprise.

En effet, depuis les premières publications en 2000 à propos des faiblesses conceptuelles des premières révisions de la norme 802.11, de nombreuses attaques critiques sont aisément réalisables sans nécessiter de moyens techniques et financiers importants [2,3,4,5]. Depuis la ratification des nouveaux mécanismes de sécurité [6] et l'avènement du standard Wi-Fi Protected Access [7,8], il est possible de déployer des solutions d'accès 802.11 robustes. Ces solutions reposent sur des briques d'authentification, de chiffrement, de dérivation et de distribution de clés robustes (si bien utilisées).

Malheureusement, une infrastructure d'accès robuste peut être mise à mal par la présence d'un point d'accès ouvert (de manière volontaire ou par erreur)

[1] dénomination par la Wi-Fi Alliance.
[2] la politique Intel avec sa marque Centrino a fortement contribué à cet engouement.

directement inter-connecté au réseau filaire de l'architecture réseau, ou encore par du déni de service sur la voie radioélectrique. Les outils classiques de supervision réseau (journaux d'activité sur les noeuds d'accès tels que les pare-feux, routeurs, points d'accès) montrent ici leurs limites car il est extrêmement difficile pour un opérateur d'infrastructure 802.11 (entreprise, hot spot[3]) d'avoir une vision claire des incidents de sécurité sur la voie radioélectrique (déni de service, faux points d'accès...).

L'arrivée des technologies de détection d'intrusion dans les réseaux 802.11 est partie de ce constat : comment observer ces réseaux et s'assurer qu'ils sont conformes à un référentiel imposé par la politique de sécurité du site protégé ? Ils doivent par exemple s'assurer que les points d'accès illégitimes[4] ne sont pas inter-connectés aux réseaux filaires de l'entreprise.

C'est dans ce sens que les technologies de détection d'intrusion dans les réseaux 802.11 ont un spectre plus large que les technologies de détection d'intrusion dans les réseaux filaires. Ces logiciels doivent être capables à la fois de détecter des attaques (grâce essentiellement à des signatures) mais ils doivent aussi qualifier les points d'accès environnants (lorsque cela est possible) pour évaluer s'ils sont inter-connectés ou non au réseau filaire de l'architecture protégée. Les procédures réalisées manuellement par des campagnes d'audits réguliers sont alors (en partie) automatisées.

Les logiciels de détection d'intrusion dans les réseaux 802.11 peuvent représenter un intérêt certain à quiconque souhaitant valider son niveau de risque par rapport aux technologies radioélectriques 802.11. En effet, il ne faut pas oublier que la moindre faille peut ouvrir une brèche importante à cause de la diffusion radioélectrique qui est intrinsèquement difficilement maîtrisable (et donc accessible à des personnes « anonymes »).

Dans la première partie, l'article décrira les principales attaques 802.11 ainsi que les techniques de détection associées. Ces techniques ne sont pas toutes implantées dans les logiciels de détection d'intrusion du commerce (les plus connus étant AirDefense, AirEspace, AirMagnet et Aruba Networks). Cet article peut donc servir de base à quiconque souhaitant développer une méthodologie d'évaluation de ces logiciels.

Ensuite, l'article présentera un retour d'expérience de déploiement d'une solution de détection d'intrusion 802.11 utilisée à France Télécom R&D.

2 Avant propos

2.1 Pré-requis

Nous supposons dans cet article que le lecteur a des connaissances dans le domaine des réseaux 802.11 [1]. Si cela n'est pas le cas, nous l'invitons à se focaliser sur les catégories de trames 802.11 ainsi que les différents en-têtes pour une meilleure compréhension des paragraphes suivants. En effet, les techniques

[3] zone d'accès publique à Internet.

[4] ceux qui ne sont pas reconnus comme appartenant au site protégé.

de détection présentées dans cette proposition d'article ne sont pas complètement explicitées, chacune d'entre-elles pouvant faire l'objet d'un article complet...

2.2 Sécurité ?

La sécurité informatique nécessite des efforts importants en terme de politique de sécurité et de moyens pour pouvoir l'appliquer. Il est donc nécessaire de concevoir des architectures réseau les plus robustes possibles, en particulier pour les réseaux 802.11.

La détection d'intrusion 802.11 vient en complément d'une architecture déjà rendue robuste par un déploiement adéquat et réfléchit. Elle ne peut pas résoudre toutes les problématiques et certainement pas lorsqu'aucun effort en sécurité n'a été réalisé : elle fait partie d'une approche globale du traitement des problématiques de sécurité, et en particulier sur les problématiques inhérentes aux réseaux radioélectriques.

3 Architectures de détection d'intrusion 802.11

La détection d'intrusion 802.11 opère essentiellement au niveau de la couche 2 du modèle de l'OSI car le but d'une telle application est de détecter des évènements sur la voie radioélectrique (signatures d'attaques, comportement de stations, présence de points d'accès illégitimes). Les attaques applicatives qui transitent sur le réseau 802.11 ne font généralement pas partie des objectifs de détection de tels logiciels.

Les deux principales raisons sont :

- ces attaques qui seraient visibles sur la voie radioélectrique le seraient aussi sur la voie filaire, par conséquent, un logiciel de détection d'intrusion classique bien positionné[5] peut être dévolu à cette tâche ;
- le trafic peut être chiffré au niveau de la voie radioélectrique rendant inutile l'analyse de la charge applicative.

En effet, même si cela est techniquement réalisable, il ne s'agit pas de greffer des fonctionnalités de détection d'intrusion classique pour détecter des évènements sécurité sur les couches supérieures lorsque le trafic sur la voie radioélectrique est en clair : ce n'est évidemment pas le but de ce type d'outil.

Actuellement, deux types d'architecture pour la détection d'intrusion 802.11 sont possibles :

- intégrée — la détection d'intrusion est réalisée au niveau des équipements d'accès (le plus souvent les points d'accès) ;
- sur-couche — la détection d'intrusion est réalisée sur des équipements dédiés (déploiement de sondes spécifiques dédiées à l'écoute de la voie radioélectrique).

La première approche impose des contraintes fortes sur les possibilités de détection si l'on n'accepte pas de perte de qualité de service pour l'accès des

[5] typiquement derrière un nuage de points d'accès.

clients sur les points d'accès. En effet, il est difficile en pratique pour un même point d'accès (qu'il soit de type « thin[6] access point » ou « fat[7] access point ») d'opérer le service de fourniture d'accès (qui impose l'utilisation d'un canal radioélectrique attribué) et d'effectuer des fonctions de détection d'intrusion (qui par définition imposent des sauts de canal[8] pour écouter successivement chacun d'entre eux). La deuxième approche, bien que nécessitant une architecture dédiée, est plus prometteuse en terme de fonctionnalités de détection d'intrusion.

4 Panorama des techniques de détection 802.11

Les principales catégories d'évènements à détecter sont :

1. la découverte des réseaux ;
2. l'injection de trafic ;
3. le déni de service ;
4. l'usurpation d'adresse Medium Access Control (MAC) ;
5. les attaques sur les mécanismes d'authentification ;
6. les attaques sur la confidentialité des trames de données ;
7. les points d'accès illégitimes ;
8. les points d'accès mal configurés ;
9. les faux points d'accès ;
10. le double attachement.

Un logiciel de détection d'intrusion a pour vocation de découvrir des évènements suspicieux. Les techniques et logiciels de détection sont bien évidemment intrinsèquement liés aux problématiques de faux positifs et faux négatifs. Le but de chacun de ces logiciels est donc de réduire au maximum ces deux paramètres en utilisant les méthodes de détection les plus pertinentes.

Cette partie décrit des techniques de détection qui permettent de repérer la majorité des attaques connues dans l'état de l'art. Bien entendu, tout n'est pas détectable ! Il existe des zones d'ombres difficilement couvertes : dans tout développement d'une telle solution il faut trouver le juste milieu entre capacités de détection et contraintes d'exploitation...

4.1 La découverte des réseaux

Elle ne constitue une attaque à proprement parler car les techniques utilisées sont parfaitement valides au sens de la norme 802.11. Cependant, détecter des

[6] dans le sens avec peu de fonctionnalités embarquées, l'essentiel des fonctions étant déportées sur un équipement centralisé souvent appelé « switch wifi ».
[7] dans le sens avec un maximum de fonctionnalités embarquées sur le point d'accès.
[8] les bandes de fréquence attribuées aux technologies 802.11 sont découpées en canaux.

outils dédiés au *WarDriving*[9] tels que `NetStumbler` [9] et `Kismet` [10], est une information intéressante[10] en soi.

Deux techniques de découverte des réseaux sont possibles :
- la découverte de réseaux passive — qui écoute les trames émises sur la voie radioélectrique par les clients et points d'accès environnants ;
- la découverte de réseaux active — qui recherche les points d'accès présents en leur posant une question appropriée à laquelle ils répondront.

La détection des outils de découverte de réseaux passive est en théorie impossible[11] au niveau 802.11 car aucune émission de trame ne devrait intervenir. Cependant, il est possible que certains drivers en mode « `monitor`[12] » continuent d'envoyer des paquets sur la voie radioélectrique ce qui permet alors leur identification.

A contrario, la détection des outils de découverte de réseaux active peut être réalisée par trois moyens différents :
- l'écoute de trames de type « probe request[13] » envoyées contenant un nom de réseau Effective Service Set IDentifier (ESSID) vide[14] — il est possible de détecter dans ce cas des outils qui ne sont pas forcément dédiés au *WarDriving* ;
- les signatures volontairement envoyées par des outils de *WarDriving* tels que `NetStumbler` [22] ;
- un comportement de recherche active de réseaux (cf. point numéro 1) sans jamais s'associer à l'un d'entre eux.

4.2 Injection de trafic

L'injection de trafic est potentiellement très intéressante pour un attaquant, car il a alors l'opportunité d'injecter des paquets dans des communications existantes s'il s'agit de trames de données. S'il est aussi capable d'injecter des trames de management et de contrôle, il est alors possible de « manipuler » la machine à état 802.11, ce qui entraîne alors des risques importants.

Détecter et résister aux attaques par injection de trafic représente un enjeu majeur pour les logiciels de détection d'intrusion 802.11. Ces dernières permettent de faire lever des alertes erronées ou saturer leurs ressources sur des données maîtrisées par l'attaquant...

[9] terme dérivé de *W*arDialing : technique de découverte de réseaux 802.11.
[10] si l'on émet l'hypothèse que la découverte des réseaux est une étape préalable à toute tentative d'intrusion.
[11] il est toutefois envisageable qu'au niveau physique, toute antenne qui entraîne fatalement des perturbations radioélectriques (par induction) soit alors détectable.
[12] mode spécifique qui permet d'écouter la voie radioélectrique, il doit être supporté par le chipset, firmware et driver pour pouvoir être utilisé.
[13] trame de découverte de point d'accès.
[14] le nom de réseau a une taille maximale de 32 octets, un nom de réseau vide est donc sur une longueur de 0 octets.

Certains chipsets, firmwares et drivers supportent l'injection de paquet en mode `raw`[15], les plus populaires étant les :

- chipset Atheros et driver `madwifi` [11] ;
- chipset Prism2/2.5/3 et driver `hostap` [12] ;
- chipset Prism54 et driver `prism54.org` [13].

Selon les équipements utilisés il est possible d'injecter des paquets 802.11 complètement spécifiés par la personne désirant injecter. Tous les champs 802.11 sont alors maîtrisés ce qui peut rendre certaines attaques très puissantes. Ce n'était pas le cas il y a quelques années lorsque seuls les chipsets Prism2/2.5/3 étaient disponibles car certains champs[16] étaient gérés par le firmware.

Des initiatives ont aussi vu le jour pour pouvoir construire et injecter des paquets 802.11 via des développements en langage C :

- `airjack` [14] ;
- `libradiate` [15] ;
- `libwnet` [16] ;
- `libwlan` [17].

Cependant, la plupart de ces développements sont non maintenus et pas complètement fonctionnels. En pratique, il est souvent préférable de créer les paquets 802.11 « à la main » lors du développement d'outils 802.11 en langage C ou d'utiliser un outil de création et d'injection de paquets tel que `scapy` [18] pour des tests 802.11 ne nécessitant pas une rapidité d'injection.

Injection Wired Equivalent Privacy (WEP). Les attaques qui compromettent le protocole WEP sont légion. Il est par exemple possible d'injecter des paquets induisant des retours afin de créer du trafic chiffré avec WEP. Les outils de cassage de secret[17] partagé WEP sont les premiers à bénéficier de cette propriété car cela permet de casser rapidement ce secret partagé même sur un réseau non chargé. Ces attaques exploitent l'absence de techniques anti-rejeu dans le protocole WEP, ce qui implique pour l'attaquant qui injecte des paquets WEP de réutiliser la même sortie de l'algorithme Rivest Cipher 4 (RC4) et donc le même Initialization Vector (IV). D'autres attaques exploitent l'Integrity Check Value (ICV) du protocole WEP qui est seulement un Cyclic Redundancy Check 32 (CRC32).

En pratique, il est très peu probable[18] d'avoir un grand nombre de paquets WEP contenant le même IV dans une fenêtre de temps relativement courte[19]. Le paradoxe des anniversaires permet d'estimer une probabilité de 50% de collision

[15] le mode `raw` est un mode spécifique permettant d'injecter des trames de niveau bas (dans notre cas au niveau 802.11).

[16] parmi ceux-ci le numéro de séquence et la fragmentation.

[17] la norme spécifie une longueur de 40 bits, mais d'autres longueurs sont possibles (104 bits étant la plus courante).

[18] sauf cas particulier, typiquement une carte cliente 802.11 envoyant le premier paquet WEP avec un IV nul à chaque remise à zéro de la carte.

[19] typiquement la minute.

pour environ 4800 paquets WEP... Un calcul rapide permet d'estimer à 30000 secondes le temps nécessaire pour atteindre cette probabilité.

Ceci constitue une signature de l'attaque (détection d'occurrences d'IV pour une même source sur une fenêtre de temps). A noter qu'il a été possible en utilisant cette technique de détecter des cartes ayant un générateur pseudo-aléatoire d'IV qui était biaisé.

Injection Temporal Key Integrity Protocol (TKIP). Grâce aux mécanismes d'intégrité et d'anti-rejeu présents dans TKIP, il n'existe pas à l'heure actuelle de technique connue permettant d'injecter efficacement des paquets valides chiffrés en TKIP. En effet, bien que le Message Integrity Check (MIC) basé sur l'algorithme Michael [19] soit notoirement connu comme étant accessible[20] selon [20], les mécanismes de contre-mesure implantés dans TKIP imposent une remise à zéro des clés de session de chiffrement et d'intégrité si plusieurs tentatives d'injection de paquets ont été détectées[21]. Un effet de bord est la possibilité de réaliser des attaques de déni de service sur TKIP si ces contre-mesures sont activées au niveau des points d'accès et clients.

Dans ce cas précis, comme le MIC est chiffré, il n'est pas possible de vérifier son intégrité au niveau de la voie radioélectrique. Il est donc nécessaire dans ce cas de journaliser des statistiques au niveau des points d'accès déployés sur les vérifications d'intégrité erronées.

Injection Counter CBC-MAC Protocol (CCMP). Grâce aux mécanismes d'intégrité et d'anti-rejeu présent dans CCMP, il n'existe pas à l'heure actuelle de technique connue permettant d'injecter des paquets valides chiffrés et intègres en CCMP.

Injection en clair. Si l'injection de trafic de données utilise une variation des numéros de séquence cohérente, il est (extrêmement) difficile de détecter de l'injection de trafic avec une analyse au niveau 802.11. Nous verrons dans la partie « usurpation d'adresse MAC » que des techniques annexes peuvent donner des éléments de réponse.

Saut de Virtual Local Area Network (VLAN). Certains points d'accès sont vulnérables à des attaques classiques de type saut de VLAN. Il est tout à fait possible de détecter ce type d'attaque lorsque les trames en clair au niveau radioélectrique comportent des entêtes de type 802.1Q [21]. Dans ce cas précis, le logiciel de détection d'intrusion 802.11 devra être capable de reconnaître et dépiler les entêtes LLC et 802.1Q des trames de donnée 802.11.

[20] force réelle de 29 bits.
[21] vérificateur d'intégrité invalide.

4.3 Usurpation d'adresse MAC

Cette partie traite des techniques de détection d'usurpation d'adresse MAC au niveau de trames de données et de management. Les trames de contrôle ne rentrent pas dans le champ des techniques ci-dessous (sauf pour la technique d'analyse de qualité de signal reçue).

Toutes les techniques décrites ci-dessous sont sujettes à des faux positifs. Il est donc nécessaire d'implanter des techniques de corrélation entre plusieurs catégories d'usurpation d'adresse MAC avec des systèmes de poids selon la fiabilité de chacune de ces techniques.

Ces techniques imposent que les deux équipements (légitime et illégitime) communiquent sur la voie radioélectrique à un moment donné (et si possible en même temps) car elles résident sur comparaisons au niveau des trames 802.11 ou de la puissance de signal reçue.

Vérification de la cohérence des numéros de séquence. Cette technique fonctionne sur toutes les trames de signalisation et de donnée. Ces trames contiennent un champ sur 12 bits appelé le Sequence Number. Ce numéro de séquence est utilisé lors du ré-assemblage des paquets fragmentés. Il est difficilement[22] modifiable par un attaquant s'il utilise un driver en mode « `master`[23] ». Par contre il est possible d'injecter du trafic (typiquement en mode « `monitor` ») avec des numéros de séquence préalablement choisis, mais il est difficile d'être parfaitement en adéquation avec les numéros de séquence émis par les points d'accès légitimes. En effet, l'attaquant ne peut empêcher l'équipement légitime d'envoyer des paquets qui seront alors en discordance au niveau de la cohérence des numéros de séquence.

```
No   Source           Destination   Protocol Info
1    Cisco_3e:a0:5a   Broadcast     Beacon frame, SSID: "FTRDWPA"

IEEE 802.11
    Type/Subtype: Beacon frame (8)
    Frame Control: 0x0080 (Normal)
    Duration: 0
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
    Source address: Cisco_3e:a0:5a (00:0d:28:3e:a0:5a)
    BSS Id: Cisco_3e:a0:5a (00:0d:28:3e:a0:5a)
    Fragment number: 0
    Sequence number: 5 <-- ici

No   Source           Destination   Protocol Info
2    Cisco_3e:a0:5a   Broadcast     Beacon frame, SSID: "FTRDWPA"
```

[22] implique des modifications au niveau du driver dans l'hypothèse où le firmware accepte l'injection de ces champs.

[23] ce mode offre la fonctionnalité de point d'accès.

```
IEEE 802.11
    Type/Subtype: Beacon frame (8)
    Frame Control: 0x0080 (Normal)
    Duration: 0
    Destination address: Broadcast (ff:ff:ff:ff:ff:ff)
    Source address: Cisco_3e:a0:5a (00:0d:28:3e:a0:5a)
    BSS Id: Cisco_3e:a0:5a (00:0d:28:3e:a0:5a)
    Fragment number: 0
    Sequence number: 1206 <-- ici
```

L'exemple ci-dessus montre un dé-séquencement des numéros de séquence pour deux trames successives et rapprochées dans le temps. Il s'agit dans ce cas-là de deux trames de « beacon ». Il est donc possible grâce à une analyse des différences sur les numéros de séquence de détecter une usurpation d'adresse MAC [23].

Vérification de la cohérence des étiquettes temporelles. Cette technique n'est utilisable que sur les trames de signalisation de type « beacon[24] » et « probe response[25] » qui sont issues des points d'accès ou des stations en mode Ad-Hoc[26]. Ces trames contiennent un champ sur 64 bits appelé le Basic Service Set (BSS) Timestamp. Ce champ sert à la synchronisation des stations attachées au point d'accès. De la même manière que pour le numéro de séquence, les étiquettes temporelles sont difficilement modifiables par un attaquant s'il utilise un driver en mode « `master` ». Par contre il est possible d'injecter du trafic (typiquement en mode « `monitor` ») avec des étiquettes temporelles préalablement choisies, mais il est difficile d'être parfaitement en adéquation avec les étiquettes temporelles émises par les points d'accès légitimes. En effet, l'attaquant ne peut empêcher l'équipement légitime d'envoyer des paquets qui seront alors en discordance au niveau de la cohérence des étiquettes temporelles.

```
No   Source           Destination   Protocol Info
1    Cisco_3e:a0:5a   Broadcast     Beacon frame, SSID: "FTRDWPA"

IEEE 802.11 wireless LAN management frame
    Fixed parameters (12 bytes)
        Timestamp: 0x000002F91447819C <-- ici
        Beacon Interval: 0.102400 [Seconds]

No   Source           Destination   Protocol Info
```

[24] trame de balise émise par les points d'accès pour signifier leur présence et configuration, elles sont généralement émises régulièrement avec un intervalle appelé « beacon interval ».

[25] trame de réponse aux trames de découverte de réseaux « probe request ».

[26] aussi appelé Independent Basic Service Set (IBSS)

```
2    Cisco_3e:a0:5a   Broadcast     Beacon frame, SSID: "FTRDWPA"

IEEE 802.11 wireless LAN management frame
    Fixed parameters (12 bytes)
        Timestamp: 0x00000000000962EA <-- ici
        Beacon Interval: 0.102400 [Seconds]
```

L'exemple ci-dessus montre un dé-séquencement des étiquettes temporelles pour deux trames successives et rapprochées dans le temps. Il s'agit dans ce cas-là de deux trames de « beacon ». En réalisant la conversion en unités de temps, nous avons aussi une information pertinente (2f91447819c correspond à environ 908 heures et 962ea correspond à environ 0,6 secondes) qui peut nous aider à trouver des points d'accès récemment démarrés[27] (en particulier des points d'accès via des cartes PCMCIA sur des ordinateurs portables). Il est donc possible grâce à une analyse des différences sur les étiquettes temporelles de détecter une usurpation d'adresse MAC [26].

Vérification de la cohérence des paramètres optionnels. Les paramètres optionnels[28] sont présents dans la plupart des trames de signalisation et en particulier dans les trames de type « beacon », « probe request » ou « probe response ».

Ces paramètres optionnels permettent d'obtenir des informations pertinentes sur les capacités en terme de débit, de support des mécanismes de sécurité comme les Wi-Fi Protected Access Information Element (WPA IE) et Robust Security Network Information Element (RSN IE) ou autres... En mode « `master` » certains de ces champs dépendent à la fois des capacités[29] intrinsèques de la carte, du firmware et du driver, mais aussi de la configuration[30].

```
No   Source           Destination   Protocol Info
1    Cisco_3e:a0:5a   Broadcast     Beacon frame, SSID: "FTRDWPA"

 Tagged parameters (24 bytes)
        SSID parameter set: "FTRDWPA"
        Supported Rates: 1.0(B) 2.0(B) 5.5 11.0
        DS Parameter set: Current Channel: 1
        (TIM) Traffic Indication Map: DTIM 0 of 1 bitmap empty

No   Source           Destination   Protocol Info
2    Cisco_3e:a0:5a   Broadcast     Beacon frame, SSID: "FTRDWPA"
```

[27] bien entendu des modifications au niveau des drivers permettent de démarrer avec un étiquette temporelle élevée.
[28] dénommés « tagged parameters » dans la norme 802.11.
[29] typiquement les débits supportés.
[30] typiquement le nom de réseau.

```
Tagged parameters (88 bytes)
        SSID parameter set: "FTRDWPA"
        Supported Rates: 1.0(B) 2.0(B) 5.5(B) 11.0(B) <-- ici
        DS Parameter set: Current Channel: 1
        (TIM) Traffic Indication Map: DTIM 0 of 2 bitmap empty
        Vendor Specific: WPA <-- ici
        Vendor Specific: Aironet <-- ici
```

L'exemple ci-dessus montre des incohérences sur les paramètres optionnels pour deux trames successives et rapprochées dans le temps. Il s'agit dans ce cas-là de deux trames de « beacon » (la première annonce des capacités classiques et la deuxième annonce des capacités WPA et Aironet). En choisissant judicieusement les paramètres optionnels à vérifier, il est donc possible de détecter une usurpation d'adresse MAC [26].

Vérification de la cohérence de la qualité de signal reçue. Le Received Signal Strength Index (RSSI) est indicateur de la qualité de signal reçue par la carte 802.11 en mode « `monitor` ». Bien qu'il soit non trivial[31] d'estimer avec précision la distance à laquelle se situe l'émetteur, il est tout à fait possible de comparer les variations de RSSI pour des trames émises par un prétendu même émetteur (selon son adresse MAC) et d'en déduire alors des incohérences.

Par conséquent, une analyse des différences sur les RSSI permet de détecter une usurpation d'adresse MAC. Avec un cependant un bémol sur la sensibilité aux faux positifs de cette technique.

```
No   Source           Destination   Protocol Info
1    Cisco_3e:a0:5a   Broadcast     Beacon frame, SSID: "FTRDWPA"

Prism Monitoring Header
    <snip>
    RSSI: 0xfb <-- ici
    <snip>

No   Source          Destination     Protocol Info
2    Cisco_3e:a0:5a  Broadcast       Beacon frame, SSID: "FTRDWPA"

Prism Monitoring Header
    <snip>
    RSSI: 0xbb <-- ici
    <snip>
```

L'exemple ci-dessus montre une différence importante des RSSI pour deux trames successives et rapprochées dans le temps. Il s'agit dans ce cas-là de deux

[31] de nombreux paramètres entrent en ligne de compte tels que la puissance d'émission de la carte, les obstacles et leurs atténuations respectives, les trajectoires multiples, les interférences...

trames de « beacon ». Même si une variation de qualité de signal reçue peut être très importante du fait du comportement difficilement prévisible de la voie radioélectrique, cela constitue quand même une information en soi. En effet, grâce à ce RSSI, il est possible d'implanter des techniques basiques à seuil, ou de manière plus élaborée d'appliquer un modèle de propagation et de résolution en fonction du site protégé. Toute analyse sur les différences de RSSI peut donc détecter et géo-localiser un évènement sur la voie radio pour peu qu'il soit capté par plusieurs sondes.

Note 1. Pour utiliser cette technique il est primordial de connaître ce qui est réellement remonté par le chipset et le firmware. En effet, des opérations de conversions sur les informations de signal (en dB par exemple) peuvent être effectuées de manière complètement différentes selon les interfaces radioélectriques utilisées.

Vérification de la cohérence sur les couches supérieures. Sur les trames de données en clair, il est possible de réaliser une prise d'empreinte passive des systèmes d'exploitation grâce aux caractéristiques de la pile TCP/IP de l'équipement [25]. Si pour une même adresse MAC côté radioélectrique dans une fenêtre de temps définie deux systèmes d'exploitation différents sont présents alors nous pouvons en déduire une usurpation d'adresse MAC.

Corrélation des détections. Les techniques de détection présentées ci-dessus sont toutes sujettes aux faux positifs. Afin de les limiter, il est nécessaire d'implanter une corrélation à la volée sur les alertes émises par ces techniques. Cette corrélation apporte alors un niveau de fiabilité bien plus élevé grâce à un système de poids. Il permet en effet d'affecter une sévérité plus élevée lorsque plusieurs techniques de détection d'usurpation d'adresse MAC ont été déclenchées : corrélation en une seule alerte de type « usurpation d'adresse MAC » d'une sévérité plus élevée.

4.4 Déni de service

Les attaques décrites dans le paragraphe présent sont détectables en utilisant des systèmes à seuil ou par répartition statistiques par type de trame sur un réseau donné.

Un autre paramètre peut être pris en compte, la destination de la trame induisant du déni de service. En effet, si l'adresse MAC destination est l'adresse de `broadcast` alors nous avons détecté un comportement non conforme avec fiabilité[32]. Il faut donc distinguer ces évènements particuliers grâce à une séparation en des signatures spécifiques.

Grâce à des techniques à seuils les attaques suivantes sont détectables[33] :

[32] la plupart des attaques de déni de service utilisent l'adresse de broadcast pour des raisons de simplicité et d'efficacité de l'attaque.

[33] le positionnement des seuils est primordial.

- flot de trames de dé-authentification ou dé-association ;
- flot d'associations au point d'accès ;
- flot de trames avec le champ « Duration Field » élevé ;
- flot de trames « PS-Poll » ;
- trames fragmentées ;
- trames d'échec Extensible Authentication Protocol (EAP) ;
- trames d'échec d'authentification de la méthode EAP ;
- trames d'échec sur le 4-way handshake.

Les seuils sont souvent positionnés de manière empirique. Il existe malheureusement toujours une fenêtre dans laquelle une attaque peut passer inaperçue, mais cette dernière sera a priori moins efficace et donc moins intéressante pour l'attaquant. Toute méthode statistique plus évoluée est aussi utilisable dans ce contexte.

Note 2. Il est bien entendu que cette liste est non exhaustive, il est par exemple possible de réaliser du déni de service au niveau plus bas par saturation des ressources radio ou brouillage...

4.5 Les attaques sur les mécanismes d'authentification

La norme 802.11i rend obligatoire l'utilisation d'une méthode d'authentification mutuelle. Elle peut être réalisée par Pre-Shared Key (PSK) ou par méthode EAP, les plus courantes étant :

- EAP Transport Layer Security (EAP-TLS) [27] ;
- Protected EAP (PEAP) [28] ;
- EAP Tunneled Transport Layer Security (EAP-TTLS) [29] ;
- Lightweight EAP (LEAP) [30].

Recherche exhaustive sur méthode EAP. Certaines méthodes d'authentification EAP reposent sur l'utilisation d'un couple « identifiant et mot de passe ». Elles peuvent donc être sensibles à des attaques par recherche exhaustive ou par dictionnaire « en-ligne ». De part la propagation radioélectrique, ces tentatives d'authentification sont accessibles anonymement à quiconque se situe dans la zone de couverture des points d'accès.

Dans ce cas précis, les serveurs d'authentification peuvent lever des alertes lorsqu'ils détectent de multiples essais erronés. Cela peut aussi être réalisé par un logiciel de détection d'intrusion au niveau 802.11, même si c'est clairement moins approprié.

4.6 Les attaques sur la confidentialité des trames de données

Cassage WEP. Depuis la publication majeure de Fluhrer, Mantin et Shamir, les outils de cassage WEP ont foisonné avec de nombreuses évolutions qui permettent aujourd'hui de récupérer le secret partagé en quelques minutes sur un réseau chargé [31,32].

Cette attaque nécessite que du trafic WEP soit présent, des techniques sont alors apparues qui permettre « d'induire » du trafic grâce aux techniques d'injection de trafic présentées dans le paragraphe « injection WEP ».

Il est donc possible de détecter du cassage WEP (ou la possibilité de le réaliser) par :

- la détection d'injection de trafic WEP ;
- la détection de paquets WEP permettant de réaliser le cassage (selon les informations décrites dans [2]).

Bien entendu dans tout déploiement 802.11, le protocole WEP n'est pas à recommander. Depuis l'arrivée des standards WPA et WPA2 qui supportent les protocoles TKIP et CCMP, il est possible d'arriver à un niveau de confidentialité et d'intégrité très élevé, même avec des équipements grand public.

Cassage TKIP et CCMP. Il n'existe pas à l'heure actuelle de technique connue permettant de casser les protocoles TKIP et CCMP si la chaîne précédente (authentification, dérivation des clés maîtresses, dérivation des clés de session de chiffrement et d'intégrité) ne présente pas de faiblesse.

4.7 Points d'accès illégitimes

La difficulté ne réside pas dans la détection des points d'accès illégitimes comme cela est décrit dans ce paragraphe, mais plutôt dans la qualification de ces derniers. En effet, un point d'accès illégitime inter-connecté au réseau du site protégé est une alerte critique et nécessite une action rapide.

En pratique plusieurs possibilités sont envisageables :

- point d'accès illégitime non inter-connecté au réseau filaire du site protégé mais présent dans l'enceinte de ce dernier ;
- point d'accès illégitime inter-connecté au réseau filaire du site protégé ;
- point d'accès interférant[34].

Détection du point d'accès. La détection se réalise en capturant toutes les trames issues de points d'accès :

- « beacon, probe response, authentication response, association response, re-association response » pour les trames de management ;
- « data from ds, data to ds » pour les trames de données.

Comparaison des trames reçues avec :

- la liste des noms des réseaux autorisés (ESSID) ;
- la liste des adresses MAC côté radioélectrique des points d'accès autorisés (BSSID).

Des listes « blanches » permettent alors d'identifier rapidement les points d'accès non conformes que l'on appelle couramment les « rogue ap ». Il est maintenant nécessaire de qualifier les points d'accès en essayant de déterminer s'ils sont inter-connectés au réseau filaire du site protégé.

[34] correspond au premier cas qualifié en tant que point d'accès externe non dangereux.

Qualification du point d'accès. Cette qualification est réalisée grâce aux briques suivantes :

1. recherche des adresses MAC dans des bases d'inventaires d'équipement
   (a) sur les adresses MAC source et destination d'une trame de donnée,
   (b) sur l'adresse BSSID +1/-1 ;
2. association automatique et vérification de connectivité
   (a) association sur le nom de réseau du point d'accès à qualifier,
   (b) récupération des informations de connectivité via DHCP,
   (c) envoi d'un paquet vers une (ou plusieurs) adresses IP prédéterminées ;
3. géolocalisation physique de l'émetteur
   (a) récupération des RSSI de l'évènement sur chacune des sondes,
   (b) utilisation d'un modèle de propagation,
   (c) estimation de la position géographique sur le site.

La première technique permet de repérer rapidement la présence d'un point d'accès sur les commutateurs du site protégé bien que cela nécessite d'avoir une base d'inventaire régulièrement à jour.

La deuxième technique est plus active et doit être utilisée avec précautions. Il suffit d'imaginer un attaquant créant des milliers de faux points d'accès à valider... Cette technique n'est réalisable que si l'on a une bonne confiance dans les techniques de détection afin de ne travailler que sur des données fiables et non maîtrisées par l'attaquant : c'est un problème majeur pour tout éditeur de solution de détection d'intrusion 802.11 !

La troisième technique a pour principal avantage d'être indépendante de la trame émise par l'attaquant. En effet, même si un attaquant peut faire varier la puissance d'émission de sa carte 802.11, la qualité de signal reçue en chacune des sondes déployées reste quand même cohérente pour un paquet donné. Si le modèle de résolution des équations est suffisamment évolué et le calibrage suffisamment fin, cette technique n'est pas à négliger.

Note 3. Ces techniques de qualification des points d'accès ne permettent pas de résoudre toutes les problématiques. Un attaquant peut arriver à les contourner avec plus ou moins de facilité. Cependant, elles sont tout à fait pertinentes dans le cadre d'une erreur de configuration ou d'un attaquant de niveau peu élevé.

Points d'accès mal configurés. Les techniques exposées peuvent aussi être appliquées aux points d'accès en liste « blanche » pour valider leur configuration. Cependant, il est plus cohérent de réaliser cette opération au niveau filaire car l'opérateur de l'infrastructure a (normalement) la maîtrise et la connaissance de tous les points d'accès présents sur son site. Dans ce cas, il lui est possible d'automatiser une campagne de test sur les configurations des points d'accès pour les valider régulièrement.

4.8 Les faux points d'accès

Les faux points d'accès sont des points d'accès « simulés » ou « émulés ». Ils sont souvent utilisés pour perturber les outils de *WarDriving* et font généralement partie du cahier de tests lors d'évaluation d'outils de détection d'intrusion 802.11.

Fake AP. Outil en langage PERL qui repose sur les commandes `iwconfig` et `ifconfig` [33]. Les étiquettes temporelles sont remises à zéro à chaque changement de BSS en mode « `master` ». Il est donc aisé de détecter des attaques de type Fake AP en repérant de nombreux points d'accès ayant une étiquette temporelle anormalement très basse.

Pour des raisons de lisibilité seules les informations pertinentes sont présentées dans l'exemple ci-dessous.

```
No   Source           Destination   Protocol Info
1    Cisco_b3:f9:7d   Broadcast     Beacon frame, SSID: "airport"

Prism Monitoring Header
    RSSI: 0xfb <-- ici
IEEE 802.11
    Sequence number: 2871 <-- ici
IEEE 802.11 wireless LAN management frame
        Timestamp: 0x000000000001929A <-- ici
    Tagged parameters (24 bytes)
        SSID parameter set: "airport"
        Supported Rates: 1.0(B) 2.0(B) 5.5 11.0
        DS Parameter set: Current Channel: 1

No   Source           Destination  Protocol Info
3    Cisco_b3:f9:7d   Broadcast    Beacon frame, SSID: "airport"

Prism Monitoring Header
    RSSI: 0xfb <-- ici
IEEE 802.11
    Sequence number: 2872
IEEE 802.11 wireless LAN management frame
    Fixed parameters (12 bytes)
        Timestamp: 0x00000000000323A2 <-- ici
    Tagged parameters (24 bytes)
        SSID parameter set: "airport"
        Supported Rates: 1.0(B) 2.0(B) 5.5 11.0
        DS Parameter set: Current Channel: 1

No   Source           Destination   Protocol Info
7    Cisco_4a:bb:ab   Broadcast     Beacon frame, SSID: "host"
```

```
Prism Monitoring Header
    RSSI: 0xfb <-- ici
IEEE 802.11
    Sequence number: 2873 <-- ici
IEEE 802.11 wireless LAN management frame
    Fixed parameters (12 bytes)
        Timestamp: 0x00000000000192F0 <-- ici
    Tagged parameters (21 bytes)
        SSID parameter set: "host"
        Supported Rates: 1.0(B) 2.0(B) 5.5 11.0
        DS Parameter set: Current Channel: 1

No   Source            Destination   Protocol Info
9    Cisco_4a:bb:ab    Broadcast     Beacon frame, SSID: "host"

Prism Monitoring Header
    RSSI: 0xfb <-- ici
IEEE 802.11
    Sequence number: 2874 <-- ici
IEEE 802.11 wireless LAN management frame
    Fixed parameters (12 bytes)
        Timestamp: 0x000000000003226B <-- ici
    Tagged parameters (21 bytes)
        SSID parameter set: "host"
        Supported Rates: 1.0(B) 2.0(B) 5.5 11.0
        DS Parameter set: Current Channel: 1

No   Source            Destination   Protocol Info
11   Cisco_4a:bb:ab    Broadcast     Beacon frame, SSID: "host"

Prism Monitoring Header
    RSSI: 0xfb <-- ici
IEEE 802.11
    Sequence number: 2875 <-- ici
IEEE 802.11 wireless LAN management frame
    Fixed parameters (12 bytes)
        Timestamp: 0x0000000000004B3A9 <-- ici
    Tagged parameters (21 bytes)
        SSID parameter set: "host"
        Supported Rates: 1.0(B) 2.0(B) 5.5 11.0
        DS Parameter set: Current Channel: 1

No   Source            Destination   Protocol Info
15   Cisco_5b:1a:70    Broadcast     Beacon frame, SSID: "airport"
```

```
Prism Monitoring Header
    RSSI: 0xfb <-- ici
IEEE 802.11
    Sequence number: 2876 <-- ici
IEEE 802.11 wireless LAN management frame
    Fixed parameters (12 bytes)
        Timestamp: 0x0000000000019328 <-- ici
    Tagged parameters (24 bytes)
        SSID parameter set: "airport"
        Supported Rates: 1.0(B) 2.0(B) 5.5 11.0
        DS Parameter set: Current Channel: 1
```

Plusieurs points méritent notre attention...

L'ensemble des étiquettes temporelles reste à un niveau très bas car elles sont remises à zéro à chaque changement de BSS. Il suffit donc de corréler tous les évènements incriminés entre eux de manière à ne lever qu'une alerte de niveau élevé ayant comme signature Fake AP.

Le numéro de séquence est incrémenté à chaque fois de un même après un changement de BSS. Ceci peut être un critère supplémentaire pour détecter ce type d'attaque grâce à une analyse statistique sur la distribution des numéros de séquence.

Le RSSI n'est finalement pas très variant pour des points d'accès sensés être différents !

Raw Fake AP. Outil en langage C qui utilise l'injection de paquet pour envoyer des trames de type « beacon » et « probe response » en mode « `monitor` » [34].

Ce type d'outil est facile à détecter (de nombreux nouveaux points d'accès en même temps et d'un seul coup), mais difficile à gérer pour les outils de détection d'intrusion. En effet, il est nécessaire qu'ils implantent des techniques d'analyse fines pour distinguer des points d'accès émulés des points d'accès réels. Sinon, ils sont potentiellement vulnérables à des attaques visant à les saturer d'évènements inutiles.

4.9 Le double attachement

Le double attachement est l'interconnexion illégitime entre deux réseaux de niveaux de sécurité différents par l'intermédiaire d'un équipement non dévolu à cette fonction.

Cette interconnexion est aujourd'hui de plus en plus courante pour plusieurs raisons : l'apparition en standard de cartes 802.11 dans les ordinateurs portables, certaines configurations laxistes qui autorisent l'association à des points d'accès inconnus et la possibilité de réaliser des points d'accès illégitimes très facilement.

Si l'on exclut le double attachement volontaire[35] qui est détectable mais difficilement repérable (où seules des techniques de géolocalisation sont efficaces),

[35] typiquement l'utilisation d'un routeur 802.11 avec authentification et chiffrement robustes.

on peut considérer que le double attachement par « erreur » implique l'utilisation du mode « ouvert[36] ».

Dans ce cas, il est possible de réaliser une analyse applicative sur :

- le protocole Dynamic Host Control Protocol (DHCP) qui inclut de nombreuses informations pertinentes (nom de la machine, sous-réseau, passerelle...) ;
- le protocole Server Message Block (SMB) qui inclut de nombreuses informations pertinentes (nom de la machine...) ;
- tout autre échange applicatif pouvant donner des informations pertinentes (processus automatiques).

Ces informations peuvent permettre d'identifier avec une certaine finesse que l'ordinateur ayant un nom découvert précédemment est présent aussi sur le réseau filaire en réalisant une corrélation avec les journaux d'activité des serveurs internes DHCP et SMB. Bien entendu, il est préférable d'avoir une attitude préventive face à ces menaces en s'appuyant sur du durcissement de configuration client.

5 Les techniques de prévention d'intrusion

La prévention d'intrusion 802.11 a pour but d'empêcher l'exploitation de brèches réseau induites par les technologies 802.11.

Typiquement, les principales fonctionnalités de ces outils sont basées sur des :

- contre-mesures radioélectriques (utilisation de certaines techniques de déni de service décrites précédemment) pour empêcher l'association de clients sur des points d'accès considérés comme étant illégitimes ;
- contre-mesures au niveau filaire (au niveau des commutateurs) pour empêcher l'utilisation de points d'accès considérés comme étant illégitimes ;
- contre-mesures sur le cassage WEP.

Comme toute technologie « active », il est primordial d'avoir des garde-fous afin d'éviter de se faire manipuler par une personne malveillante. Si en effet, il est possible de placer des points d'accès légitimes dans la liste des points d'accès devant être « isolés » cela peut impliquer du déni de service sur des équipements légitimes. Il faut réaliser un processus de validation manuelle avant le lancement de toute contre-mesure active. Par ailleurs, nous n'aborderons pas les problématiques juridiques dans le cas où le système couperait des utilisation légitimes de points d'accès interférants... Les solutions actuelles ne seront pas plus développées dans ce chapitre, ce sujet pouvant faire l'objet d'un article complet [24].

[36] sans authentification ni chiffrement de la voie radioélectrique, ce mode est souvent utilisé dans les hot spots 802.11, l'authentification et le contrôle d'accès étant réalisés via des technologies de type « portail captif ».

6 Conception d'un logiciel de détection d'intrusion 802.11

Cette partie n'a pas pour but de présenter tous les choix techniques lors de la conception de l'outil développé par France Télécom R&D. Elle décrit brièvement les principales orientations prises tout au long des développements.

La solution développée est de type sur-couche qui intègre :
- les sondes de détection d'intrusion ;
- les mécanismes d'agrégation et de corrélation des alertes issues des sondes ;
- l'interface de visualisation et d'administration.

6.1 Le coeur de détection

Le coeur de détection est un programme en C développé de zéro. Le choix de ce langage est évident : la portabilité et l'efficacité. En effet, traiter des flux réseaux importants sur des équipements embarqués comme des points d'accès Linksys WRT54G(S) imposent naturellement le langage C !

Nous nous sommes efforcés de réaliser un code portable et indépendant de la technologie 802.11 sous-jacente (802.11abgn). Toute interface radioélectrique supportant la `libpcap` en environnement *nix est utilisable. Au niveau du dépilage des en-têtes l'outil supporte à la fois les datalink types `DLT_PRISM_HEADER` et `DLT_IEEE802_11`.

Le code réalisé est actuellement fonctionnel sur les architectures `x86` et `MIPS`. Pour cette dernière, nous avons utilisé l'environnement `OpenWRT` [35] afin de porter l'outil vers les Linksys WRT54G(S). Nous avons utilisé les versions *White Russian* et *Kamikaze* d'`OpenWRT`.

Le diagramme fonctionnel du coeur de détection est le suivant :
- lecture du fichier de règles et des fichiers afférents (liste blanche de BSSIDs, ESSIDs...) ;
- compilation de l'ensemble des règles en une structure arborescente en mémoire qui sera parcourue à chaque réception de paquet 802.11 ;
- initialisation de l'interface de capture via la libpcap (vérification de l'entête de réception grâce aux Data Link Type) ;
- à la réception d'un paquet
  - vérification de la conformité du paquet au standard 802.11 (évitement des trames mal-formées),
  - mise dans un tableau de paquets de la trame ayant passé les tests précédents,
  - mise à jour de ce dernier selon la taille maximum du tableau de paquets,
  - parcours de la structure arborescente et appel des fonctions de comparaisons sur le paquet en fonction des règles décrites dans le fichier de règles,
  - levée ou non d'un évènement de type log sur les règles qui auront été déclenchées par le paquet (ou un ensemble de paquets).

L'outil journalise les évènements détectés grâce à la directive de type « log ». La méthode de journalisation actuellement implantée est le protocole `SYSLOG` ce

qui permet une légèreté accrue au niveau de la sonde qui envoie la journalisation des évènements vers un collecteur central.

La volumétrie des évènements levés par une sonde peut-être très élevée même lorsqu'aucune attaque n'est en cours. En deux heures et vingt minutes, le fichier de journalisation a atteint 35 mégaoctets avec 215000 évènements journalisés (soit environ 25 évènements par seconde). L'agrégation est strictement nécessaire pour ce type de technologie !

6.2 Le corrélateur à la volée

Le volume d'évènements généré peut être très conséquent en particulier si le nombre de sondes est important (plus de 10 par exemple). Typiquement, si une sonde a environ 20 points d'accès en visibilité permanente, on est confronté à une volumétrie « normale » de 200 évènements par seconde (selon la performance de la sonde hébergeant le coeur de détection). Par ailleurs, si des attaquants essayent de saturer le système en faisant lever des milliers d'évènements, il est nécessaire d'avoir un mécanisme d'agrégation efficace.

Le logiciel SEC [36] a été choisi pour réaliser l'agrégation et la corrélation à la volée. Il permet d'avoir une solution à moindre coûts et très efficace en terme de stabilité et de performance.

Pour information, les évènements remontés par le coeur de détection sont de ce format :

```
Mar 31 15:11:54 192.168.1.10 airinvaders:
00:0D:28:3E:A0:5A | FF:FF:FF:FF:FF:FF | Info | info |
Authorized AP Whitelist SSID |
rssi=21 | ssid=FTRDWPA | channel=01 |
```

Règle qui a levé l'évènement :

```
when (packet.subtype = beacon or packet.subtype = probe_resp)
     and packet.tags[ssid].content in list("ssid_whitelist")
do log(packet.mac2 + " | " + packet.mac1 + " | " +
"Info | info | Authorized AP Whitelist SSID | rssi=" +
packet.prism_signal + " | ssid=" + packet.tags[ssid].content +
" | channel=" + packet.tags[ds_pset].content +" |", 1) and break
```

Le langage de règles a été conçu avec la contrainte de pouvoir rajouter de nouvelles signatures statiques facilement sans impacter du développement sur le coeur de détection.

Le coeur de détection ne se limite pas à des signatures statiques mais est aussi capable d'implanter des fonctions de détection d'anomalie telles que les techniques de détection d'usurpation d'adresse MAC. Il est possible de réaliser des traitements sur l'ensemble du tableau de paquets ce qui permet alors une évolutivité accrue de l'outil. Les techniques de détection d'anomalie implantées dans l'outil sont basées sur l'analyse des numéros de séquence, des étiquettes temporelles et des paramètres optionnels.

```
when isspoofed(packet, 50)
do log(packet.mac2 + " | " + packet.mac1 + " | " +
"Spoofing | low | Authorized AP MAC Spoofing | rssi=" +
packet.prism_signal + " | spoofing_type=Sequence Number |", 2)
```

Afin de prendre en compte les évènements levés par le coeur de détection, des règles d'agrégation et de corrélation au niveau de SEC doivent être spécifiées.

```
type=Single
ptype=RegExp
pattern=^(\S\S\S\s+\d+\s\d\d:\d\d:\d\d)\s(\S+)
\sairinvaders:\s([0-9a-fA-F:]*)\s\\
|\s([0-9a-fA-F:]*)\s\|\s(Info)\s\|\s(\S+)\s\
|\s(Authorized\sAP\sWhitelist\sSSID))
\s\|\srssi=[0-9\-]+\s\|\sssid=(.*)\s\|\schannel=(\S+)\s\|(.*)
context=!$7_$3_$2_$8_$9
desc=$7_$3_$2_$8_$9
action=create $7_$3_$2_$8_$9 65
(write sec_output_ids $1 | $2 | $3 | $4 | $5 | $$
6 | $7 | ssid=$8 | channel=$9 |$10)

type=Single
ptype=RegExp
pattern=^(\S\S\S\s+\d+\s\d\d:\d\d:\d\d)\s(\S+)
\sairinvaders:\s([0-9a-fA-F:]*)\s\\
|\s([0-9a-fA-F:]*)\s\|\s(Info)\s\|\s(\S+)\s\
|\s(Authorized\sAP\sWhitelist\sSSID))
\s\|\srssi=[0-9\-]+\s\|\sssid=(.*)\s\|\schannel=(\S+)\s\|(.*)
context=$7_$3_$2_$8_$9
desc=$7_$3_$2_$8_$9
action=add $7_$3_$2_$8_$9 $0
```

Les règles ci-dessus sont difficiles à lire à cause des expressions régulières. SEC crée des états et des passages d'états à états en fonction d'évènements réceptionnés. Dans l'exemple ci-dessus, nous agrégeons toutes les alertes reçues de signature « `Authorized AP Whitelist SSID` » grâce à la création de contexte dépendant de plusieurs paramètres (adresses MAC source et destination...) durant une durée de 65 secondes. Ce choix nous permet une agrégation avec volumétrie constante car nous aurons une journalisation agrégée toutes les 65 secondes si l'évènement est toujours d'actualité.

Cette agrégation permet d'alléger les mises en base de données des évènements.

6.3 La mise en base de donnée

SEC journalise les évènements agrégés et corrélés en mode texte. Cette sortie est parcourue par un script dévolu à la mise en base SQL des évènements.

Cette base centrale est ensuite enrichie avec les bases d'inventaires décrites précédemment pour qualifier les points d'accès.

Nous sommes en train de travailler sur les aspects agrégation et corrélation « hors-ligne » pour repérer des attaques réalisées régulièrement en dehors des plages d'agrégation et de corrélation mises en place via SEC.

6.4 Présentation

La base de donnée SQL peut être interrogée via des scripts PHP de présentation pour remonter de manière succincte les informations les plus pertinentes.

7 Les problématiques de la détection d'intrusion 802.11

Lors du développement de l'architecture nous avons été confrontés à de nombreux problèmes auxquels nous avons essayé d'apporter des solutions. En effet, développer une solution comme un éditeur l'aurait fait est la seule manière de comprendre les problématiques inhérentes au domaine de la détection d'intrusion 802.11.

Ce chapitre présente les problématiques principales rencontrées lors de la conception et du développement de la solution.

Performance et taille du coeur de détection Nécessité d'avoir un code optimisé pour obtenir les meilleurs résultats possible pour la capture, l'analyse et la journalisation sur des équipements embarqués qui disposent de peu de mémoire vive et flash.

Nous avons porté nos efforts sur les aspects suivants :
- pas de `malloc()` autre qu'au démarrage de la sonde ;
- compilation des règles en mémoire ;
- structure arborescente des règles ;
- limitation du nombre de règles faisant appel aux traitements dans le tableau de paquets ;
- utilisation de `gprof` et efforts sur les fonctions les plus impactantes ;
- compilation avec l'option `-O3`.

L'optimisation du code et des règles sont les seuls moyens pour supporter les futures normes 802.11 qui offriront des débits encore plus importants.

Les évolutions des règles Il est contraignant de maintenir plusieurs fichiers de règles à plusieurs niveaux, tout impact sur la première liste de règles implique des modifications sur la deuxième. Par conséquent, l'architecture existante impose quelques contraintes en terme de maintenabilité mais en contrepartie les mécanismes d'agrégation et de corrélation sont très efficaces pour réduire la volumétrie des évènements.

Les faux positifs Toute technologie de détection d'intrusion est confrontée à ce problème majeur. Seuls des tests dans des environnements hétérogènes (entreprises, hot spots, conférences...) permettent de mettre à l'épreuve le coeur de détection et les règles implantées.

Les faux négatifs Toute technologie de détection d'intrusion est confrontée à ce problème majeur. Malheureusement, ce paramètre est difficile à évaluer en pratique. En effet, un logiciel de détection d'intrusion 802.11 est obligé de parcourir les différents canaux 802.11 pour détecter des signatures d'attaques connues ou de réaliser de la détection d'anomalie pour la détection d'usurpation d'adresse MAC par exemple. Dans ce contexte, la problématique des faux négatifs prend alors toute son ampleur ! Comment détecter une attaque sur le canal 1 alors que la sonde écoute sur le canal 11 ? En pratique, les évènements que l'on veut détecter en détection d'intrusion 802.11 nécessitent de nombreux paquets (déni de service, points d'accès illégitimes...) et donc seront fatalement détectés tôt ou tard. C'est une des raisons pour lesquelles la notion de faux positifs dans la détection d'intrusion 802.11 n'est pas vraiment une notion par paquet mais plutôt par attaque.

L'évaluation des performances Les critères d'évaluation des performances sont aussi difficiles à évaluer. Les suites d'évaluations en injectant des centaines de paquets 802.11 et en comparant le nombre d'évènements levés par le moteur de détection permettent de donner un ordre de grandeur, mais l'audit est bruité par de nombreux paramètres extérieurs : pertes au niveau de l'émetteur des paquets, pertes au niveau de la voie radioélectrique, pertes au niveau de la capture des paquets, pertes au niveau de l'analyse des paquets...

Nous sommes en cours de définition d'une suite de tests d'outils de détection d'intrusion qui soit la plus efficace possible compte tenu des problématiques énoncées précédemment.

Un indicateur toutefois intéressant est la charge CPU grâce à la classique commande `top` sous *nix si l'on suspecte des problèmes de performance.

L'évolution normative Les normes 802.11 évoluent très vite (802.11e, 802.11k, 802.11r) et impliquent des modifications profondes dans le coeur de détection pour être capable de dépiler ces nouvelles normes. Ce n'est pas un travail négligeable et maintenir un coeur de détection à jour implique d'y passer du temps.

La légalité des contre-mesures radioélectriques Problématique à prendre en compte et qui est certainement non triviale selon les législations des différents pays.

Les coûts de déploiement de solutions sur-couche La solution développée présente un sur-coût non négligeable car il faut déployer une architecture dédiée

à l'écoute de la voie radioélectrique. Bien que les éléments choisis sont peu onéreux (outils Open Source, Linksys WRT54G(S)), le coût humain en terme de déploiement, configuration et exploitation est lui non négligeable...

La qualification des points d'accès Ces techniques ne peuvent prétendre répondre à toutes les problématiques. Elles permettent de faire au mieux dans un cadre où l'on recherche des points d'accès non maîtrisés par des attaquants. En effet, il est toujours a priori possible de contourner ces mécanismes par un attaquant plus évolué. La technique la plus pertinente et prometteuse au regard d'attaques volontaires reste la géolocalisation par analyse des RSSI. Selon les techniques utilisées nous pouvons avoir des résultats intéressants si la couverture du site protégé est bonne. Enfin, il ne faut pas oublier que de nombreux points d'accès sont interférants. Ils ne représentent pas de menaces en soi, mais doivent être qualifiés en tant que tels !

8 Conclusions

La détection d'intrusion 802.11 est un domaine récent et ne peut-être une réponse en soi que si des efforts importants en terme de politique de sécurité du site protégé ont été réalisés : elle ne peut être utile que dans un contexte où la sécurité est prise en compte avec sérieux car elle génère fatalement des alertes qui devront être gérées par une équipe sécurité...

Bien que la majorité des attaques soient détectables par les techniques présentées dans cet article, il existe toujours une fenêtre de vulnérabilité dans laquelle l'attaquant tentera de s'introduire. Cette fenêtre doit être réduite au mieux sans avoir d'impacts en termes de faux positifs !

L'avènement des technologies radioélectriques permettra peut-être de démocratiser ce domaine qui devra alors s'adapter aux futures normes telles que 802.16, 802.20...

9 Remerciements

Je tiens à remercier les principaux contributeurs sur la solution qui a été conçue, développée et déployée à France Télécom R&D : Pierre Ansel, Roland Duffau, Stanislas Francfort, David Houssemand, Aurélien Jacobs, Jérôme Razniewski, Franck Veysset et Benjamin Zorès. Je remercie également le Comité de Programme de m'avoir permis de publier dans ce domaine en pleine effervescence.

Références

1. IEEE, *Local and metropolitan area networks, Specific requirements, Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*, 1997-1999.

2. Scott Fluhrer, Itsik Mantin, Adi Shamir, *Weaknesses in the Key Scheduling Algorithm of RC4*, 2001.
3. William A. Arbaugh et al., *Your 802.11 Network has No Clothes*, 2001.
4. J. Walker, *Unsafe at any key size : an analysis of the* WEP *encapsulation*, 2000.
5. N. Borisov, I. Goldberg, and D. Wagner, *Intercepting Mobile Communications : The Insecurity of 802.11*, 2001.
6. IEEE, *Local and metropolitan area networks, Specific requirements, Part 11 : Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, Amendment 6 : Medium Access Control (MAC) Security Enhancements*, 2004.
7. Wi-Fi Alliance, *Wi-Fi Protected Access*, `http ://www.wi-fi.org/`, 2003.
8. Wi-Fi Alliance, *Wi-Fi Protected Access 2*, `http ://www.wi-fi.org/`, 2004.
9. NetStumbler.com, *NetStumbler*, `http ://www.netstumbler.com/`.
10. Mike Kershaw, *Kismet*, `http ://www.kismetwireless.net/`, 2001-2005.
11. Sam Leffler, *madwifi*, `http ://madwifi.sourceforge.net/`, 2002-2005.
12. Jouni Malinen, *HostAP*, `http ://hostap.epitest.fi/`, 2001-2005.
13. Prism54.org, *prism54*, `http ://www.prism54.org/`, 2005.
14. Michael Lynn, *AirJack*, `http ://sourceforge.net/projects/airjack/`, 2002.
15. Mike Schiffman, *libradiate*, `http ://www.packetfactory.net/libradiate/`, 2001.
16. h1kari, *bsd-airtools*, `http ://www.dachb0den.com/projects/`, 2002.
17. Charles Duntze, Joachim Keinert, Lionel Litty, Laurent Butti, *libwlan*, 2003.
18. Philippe Biondi, *scapy*, `http ://www.secdev.org/`, 2003-2005.
19. Ferguson, *Michael : An Improved MIC for 802.11 WEP*, `http ://www.ieee.org/`, 2002.
20. Harkins, *Attacks against Michael and Countermeasures*, `http ://www.ieee.org/`, 2003.
21. IEEE, *Local and metropolitan area networks, Virtual Bridged Local Area Networks*, 1998.
22. Joshua Wright, *Layer 2 Analysis of WLAN Discovery Applications for Intrusion Detection*, 2002.
23. Joshua Wright, *Detecting Wireless LAN MAC Address Spoofing*, 2003.
24. Joshua Wright, *Weaknesses in Session Containment*, 2005.
25. Michal Zalewski, *p0f*, `http ://lcamtuf.coredump.cx/p0f.shtml`, 2000-2004.
26. Laurent Butti and Franck Veysset, *Design and Implementation of a Wireless IDS*, http ://neg9.org/shmoocon/shmoocon05_cd/, 2005.
27. Microsoft, *PPP EAP-TLS Authentification Protocol*, `http ://www.ietf.org/`, 1999.
28. Microsoft, Cisco, *Protected EAP Version 2*, `http ://www.ietf.org/`, 2004.
29. Funk Software, *Extensible Authentication Protocol Tunneled TLS Authentication Protocol*, `http ://www.ietf.org/`, 2005.
30. Cisco, *Lightweigth EAP*, `http ://www.cisco.com`, 2001.
31. Christophe Devine, *Aircrack*, `http ://www.cr0.net :8040/code/network/`, 2004-2005.

32. The Shmoo Group, *AirSnort*, `http ://airsnort.shmoo.com/`, 2001-2005.
33. Black Alchemy, *FakeAP*, `http ://www.blackalchemy.to/project/fakeap/`, 2002.
34. Laurent Butti, *Raw Fake AP*, `http ://rfakeap.tuxfamily.org/`, 2005.
35. OpenWRT team, *OpenWRT*, `http ://www.openwrt.org/`, 2004-2006.
36. Risto Vaarandi, *Simple Event Correlator*, `http ://www.estpak.ee/ risto/sec/`, 2001-2006.