Sécurité Wifi

Guillaume Pillot

Club de Hacking de l'université Laval

15 Mars 2016

Sommaire

Qu'est ce que le WiFi ?

- Réseau sans-fil régi par la norme 802.11
- Couche
  - Physique (1)
    - DSSS(Direct-sequence spread spectrum)
    - FHSS(Frequency Hopping Spread Spectrum)
    - Infrarouges
  - Liaison (2)
    - LLC(Logical Link Control) : 802.2
    - MAC(Media Access Control) : 802.11
  - Normes
    - 802.11b : norme la plus répandue | 6 Mbps | 2,4 GHz | 1999
    - 802.11g : 30 Mbps | 2,4 GHz | 2003
    - 802.11n : 450 Mbps | 2.4/5 GHz | 2009
- Sécurité
  - WEP : Wired Equivalent Privacy
  - WPA : Wi-Fi Protected Access
- Documentation : Lien 1 Lien 2

Hardware

- Modes des cartes WiFi : Managed/Monitor
- Alfa Network AWUS036NHR
- Documentation : Lien

Écoute du réseau avec airodump

- Installer aircrack-ng (déjà installé sur Kali)
- Mettre sa carte WiFi en mode monitor

```
# airmon-ng start wlan0
```

- Lancer l'écoute sur le réseau

```
# airodump-ng wlan0mon
```

- Recommendation : Désactiver le service network-manager (conflit possible lors des attaques)

```
# service network-manager stop
```

- Documentation : Lien

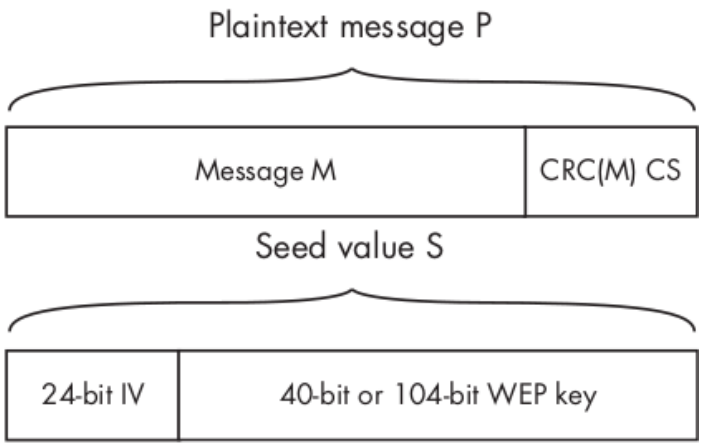
Filtrage par MAC

- Réseau OPN (ouvert)
- Bypasser le filtrage en usurpant l'adresse MAC des clients légitime (macspoofing)
- Récupérer l'adresse MAC d'un client grâce à airodump
- Changer son adresse avec macchanger
- Documentation : Lien

Le WEP

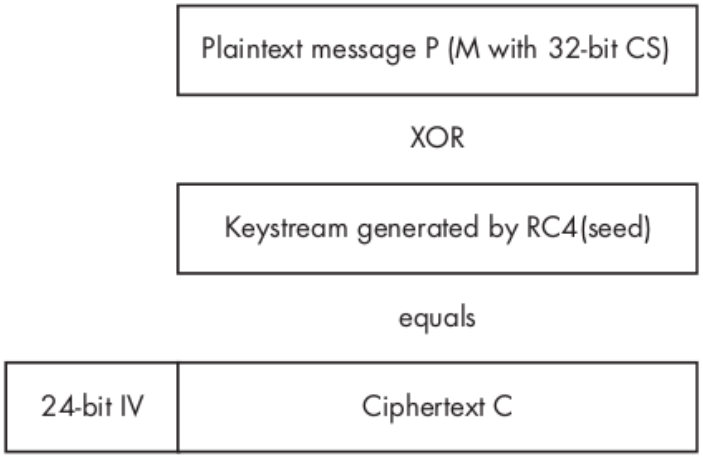
- WEP pour "Wired Equivalent Privacy" ou "Weak Encryption Protocol"
- Sorti en 1999, brisé en 2001 par l'attaque FMS
- Algorithme de chifferement RC4
- Somme de contrôle CRC-32
- Documentation : Lien

Encryption des paquets avec le WEP



- M le message en clair est toujours accompagé de sa somme de contrôle CRC32
- IV pour Initialization Vector permet d'avoir toujours une clé différente pour chaque paquet

Encryption des paquets avec le WEP



- Le message est XORé avec la clé encrypté avec RC4 qui génère le message encrypté C
- L'IV est encapsulé avec C pour permettre de déchiffrer le message avec la clé WEP. L'IV est donc difusé en clair sur le réseau.

KSA RC4

- RC4 est divisé en deux parties : le KSA et le PRGA
- KSA : Key-Scheduling Algorithm
- Pour simplifier le fonctionnement de l'algorithme, au lieu d'utiliser un tableau de 256 octets, nous en utiliserons 8. La clé a une taille de 4 octets.
- L'algorithme de key schedule a besoin de deux tableau de 8 octets, celui contenant plusieurs fois la clé K et un tableau contenant tout les chiffres de 0 à 8 (0 à 255 en vrai) :
  S = [0 1 2 3 4 5 6 7]
  K = [1 2 3 6]
  T = [1 2 3 6 1 2 3 6]

KSA RC4

Notre algorithme :

```
j = 0;
for i = 0 to 7 do
     j = (j + S[i] + K[i]) mod 8
     Swap(S[i],S[j]);
end
```

KSA RC4

- **i = 0 :**
  j = 0
  Swap(S[0],S[1])
  S = [1 0 2 3 4 5 6 7] ;
- **i = 1 :**
  j = 3
  Swap(S[1],S[3]) ;
  S = [1 3 2 0 4 5 6 7] ;
- **i = 2 :**
  j = 0
  Swap(S[2],S[0]) ;
  S = [2 3 1 0 4 5 6 7] ;
- **i = 3 :**
  j = 6 ;
  Swap(S[3],S[6])
  S = [2 3 1 6 4 5 0 7] ;
- **i = 4 :**
  j = 3
  Swap(S[4],S[3])
  S = [2 3 1 4 6 5 0 7] ;
- **i = 5 :**
  j = 2
  Swap(S[5],S[2]) ;
  S = [2 3 5 4 6 1 0 7] ;
- **i =6 :**
  j = 5 ;
  Swap(S[6],S[4])
  S = [2 3 5 4 0 1 6 7] ;
- **i = 7 :**
  j = 2 ;
  Swap(S[7],S[2])
  S = [2 3 7 4 0 1 6 5]

PRGA RC4

- PRGA : Pseudo-Random Generation Algorithm
- Soit notre message en clair P d'une taille de 4 octets :
  P = [1 2 2 2]
- Chaque case du tableau est XORé pour produire le cryptogramme C (cyphertext)
- L'algorithme :

```
i, j = 0;
loop 0 à size(P) - 1 {
    i = (i + 1) mod 8;
    j = (j + S[i]) mod 8;
    Swap (S[i], S[j]);
    t = (S[i] + S[j]) mod 8;
    k = S[t];
    Ajoute dans C k XOR P
}
```

PRGA RC4

- 1er octet :
  S = [2 3 7 4 0 1 6 5]
  i = (0 + 1) mod 8 = 1
  j = (0 + S[1]) mod 8 = 3
  Swap(S[1],S[3])
  S = [2 4 7 3 0 1 6 5]
  t = (S[1] + S[3]) mod 8 = 7
  k = S[7] = 5
  5 XOR 1
  101 XOR 001 = 100 = 4
- 2ème octet :
  S = [2 4 7 3 0 1 6 5]
  i = (1 + 1) mod 8 = 2
  j = (2 + S[2]) mod 8 = 1
  Swap(S[2],S[1])
  S = [2 7 4 3 0 1 6 5]
  t = (S[2] + S[1]) mod 8 = 3
  k = S[3] = 3
  3 XOR 2
  011 XOR 010 = 001 = 1
- 3ème octet :
  S = [2 7 4 3 0 1 6 5]
  i = (2 + 1) mod 8 = 3
  j = (1 + S[3]) mod 8 = 4
  Swap(S[3],S[4])
  S = [2 7 4 0 3 1 6 5]
  t = (S[3] + S[4]) mod 8 = 3
  k = S[3] = 0
  0 XOR 2
  000 XOR 010 = 010 = 2
- 4ème octet :
  S = [2 7 4 0 3 1 6 5]
  i = (1 + 3) mod 8 = 4
  j = (4 + S[4]) mod 8 = 7
  Swap(S[4],S[7])
  S = [2 7 4 0 5 1 6 3]
  t = (S[4] + S[7]) mod 8 = 0
  k = S[0] = 2
  2 XOR 2
  010 XOR 010 = 000 = 0
- Soit le cryptogramme C = [4 3 2 0]
- Documentation : Lien
- Livre : The art of exploitation 2nd Edition pages 433 à 436
  0x771 Wired Equivalent Privacy et 0x772 RC4 Stream Cipher

Attaque FMS

- Attaque la plus efficace contre le WEP découverte par Scott Flurher, Itsik Mantin, Adi Shamir (FMS).
- Si suffisamment d'IV faible sont collectés et si on connait les premiers octets de la keystream, on peut déterminer la clé.
- Un paquet 802.11b commence toujours par le SNAP header qui vaut toujours 0xAA. On peut donc obtenir le premier octet de la keystream en XORant 0xAA avec le premier octet encryté du paquet.

IV faible

- Un IV faible est de la forme suivante :
  A + nombre d'octets de la clé, N-1, X
  où **A** est le numéro de l'octet de la clé qui doit être attaqué
  N est la taille du tableau soit 256 avec RC4
  X qui vaut n'importe quelle valeur
- Les octets de la clé doivent être attaqué dans l'ordre, on connaît déjà les 3 premiers qui correspondent à l'IV diffuser en clair sur le réseau.

IV faible

- L'attaque s'effectue en utilisant l'algorithme du KSA
- Pour trouver le premier octet de la clé (A=0), on itère 0+3 la boucle et on effectue le calcul suivant :
  1er octet de la keystream - j - S[0+3]
- À condition seulement que : 1) j > 2 car si S[0] et S[1] sont swapé l'attaque ne marchera pas et 2) X est la bonne valeur
- Il y a environ 5% de chance que X est la bonne valeur

Exemple d'attaque

- Pour simplifier l'exemple, nous utiliserons un tableau de 16 octets au lieu de 256 (N=16) et les "octets" ont une taille de 4 bits.
- La clé vaut (1,2,3,4,5) et donc pour trouver le premier octet (A=0), l'IV doit être de cette forme : (3,15,X)

Exemple d'attaque : Attaque sur le premier octet de la clé

- Valeur du premier octet du keystream encryptée : 9
  A = 0
  IV = 3, 15, 2
  Clé = 1, 2, 3, 4, 5
  Seed = 3, 15, 2, 1, 2, 3, 4, 5
  K[] = 3 15 2 X X X X X 3 15 2 X X X X X
  S[] = 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Exemple d'attaque : Attaque sur le premier octet de la clé

- **1er itération du KSA :**
  i = 0
  j = j + S[i] + K[i]
  j = 0 + 0 + 3 = 3
  Swap S[i] and S[j]
  K[] = 3 15 2 X X X X X 3 15 2 X X X X X
  S [] = 3 1 2 0 4 5 6 7 8 9 10 11 12 13 14 15
- **2ème itération du KSA :**
  i = 1
  j = j + S[i] + K[i]
  j = 3 + 1 + 15 = 3
  Swap S[i] and S[j]
  K[] = 3 15 2 X X X X X 3 15 2 X X X X X
  S [] = 3 0 2 1 4 5 6 7 8 9 10 11 12 13 14 15
- **3ème itération du KSA :**
  i = 2
  j = j + S[i] + K[i]
  j = 3 + 1 + 15 = 7
  Swap S[i] and S[j]
  K[] = 3 15 2 X X X X X 3 15 2 X X X X X
  S [] = 3 0 7 1 4 5 6 2 8 9 10 11 12 13 14 15

- Maintenant, pour trouver le premier octet de la clé, il suffit d'effectuer le calcul suivant : 9 - j - S[A+3] = 9 - 7 - 1 = **1**

Exemple d'attaque : Attaque sur le deuxième octet de la clé

- Maintenant que l'on connaît la valeur du premier octet de la clé, on peut attaquer le suivant.
- Valeur du deuxième octet du keystream encryptée : 6
  A = 1
  IV = 4, 15, 9
  Clé = 1, 2, 3, 4, 5
  Seed = 4, 15, 9, 1, 2, 3, 4, 5
  K[] = 4 15 9 X X X X X 4 15 9 X X X X X
  S[] = 0 1 2 3 4 5 6 7 8 9 10 11 12 13 14 15

Exemple d'attaque : Attaque sur le deuxième octet de la clé

- **1er itération du KSA :**
  i = 0
  j = j + S[i] + K[i]
  j = 0 + 0 + 4 = 4
  Swap S[i] and S[j]
  K[] = 4 15 9 1 X X X X 4 15 9 1 X X X X
  S [] = 4 1 2 0 4 5 6 7 8 9 10 11 12 13 14 15
- **2ème itération du KSA :**
  i = 1
  j = j + S[i] + K[i]
  j = 4 + 1 + 15 = 4
  Swap S[i] and S[j]
  K[] = 4 15 9 1 X X X X 4 15 9 1 X X X X
  S [] = 4 0 2 1 4 5 6 7 8 9 10 11 12 13 14 15
- **3ème itération du KSA :**
  i = 2
  j = j + S[i] + K[i]
  j = 4 + 2 + 9 = 15
  Swap S[i] and S[j]
  K[] = 4 15 9 1 X X X X 4 15 9 1 X X X X
  S [] = 4 0 15 1 4 5 6 7 8 9 10 11 12 13 14 2
- **4ème itération du KSA :**
  i = 3
  j = j + S[i] + K[i]
  j = 15 + 3 + 1 = 3
  Swap S[i] and S[j]
  K[] = 4 15 9 1 X X X X 4 15 9 1 X X X X
  S [] = 4 0 15 1 4 5 6 7 8 9 10 11 12 13 14 2

- Pour trouver le deuxième octet de la clé :
  6 - j - S[4] = 6 - 3 - 1 = **2**
- On effectue les mêmes opérations pour les octets suivant la clé
- Bien sûr, pour l'exemple, les valeurs de X sont tout de suite les bonnes
- Livre : The art of exploitation 2nd Edition pages 439 à 449
  0x785 Fluhrer, Mantin, and Shamir Attack

Fake Authentication

- Pour cracker une clé WEP, la première chose à faire est d'écouter le trafic pour trouver le point d'accès WEP
  RAPPEL : La carte doit être en mode monitor (voir section airodump)

  ```
  # airodump-ng wlan0mon -w out -bssid C8:BE:19:71:BF:0A -c 11
  ```

- L'option -w out est importante, le fichier out collecte tout le trafic du point d'accès dont les IV faibles qui nous permettront de cracker la clé à la dernière étape de l'attaque. Vous devez donc garder le terminal ouvert lors de l'attaque.

Fake Authentication

- Pour capturer des IVs faibles, il faut du trafic sur le réseau. Dans notre cas, aucune machine n'est connectée.
- L'attaque "fake authentification" d'aireplay-ng permet de nous associer avec le point d'accès et par la suite de générer du trafic

```
# aireplay-ng -1 6000 -o 1 -a C8:BE:19:71:BF:0A -e hackable1
wlan0mon
```

- Documentation : Lien 1 Lien 2

Attaque Chop-Chop

- Cette attaque va générer plusieurs keystream dans un fichier .xor qui va nous permettre de forger un paquet ARP par la suite
- Le but de l'attaque est de décrypter un paquet de la taille d'un paquet ARP

```
# aireplay-ng -4 wlan0mon -a C8:BE:19:71:BF:0A -h 24:EC:99:21:8D:DA
```

- Une autre attaque similaire est l'attaque par fragmentation, si l'attaque Chop-Chop ne fonctionne pas essayer l'autre attaque.
- Documentation : Lien 1 Lien 2

Forger son paquet ARP

- Maintenant que nous avons notre fichier .xor, on va forger un paquet ARP que l'on injectera plusieurs milliers de fois sur le réseau

  ```
  # packetforge-ng -0 -a C8:BE:19:71:BF:0A -h 24:EC:99:21:8D:DA -k
  255.255.255.255 -l 255.255.255.255 -y replay_dec-0318-165321.xor -w
  arp-request
  ```

  Pour l'option -k et -l, nous utiliserons l'adresse de broadcast, le paquet forgé se nomme arp-request.
- Rappel : Le protocole ARP pour Address Resolution Protocol permet de faire le lien entre l'adresse MAC d'une machine et son adresse IP sur le réseau
- Documentation : Lien

Injection du paquet

- Nous allons injecter notre paquet ARP plusieurs milliers fois dans le réseau
- L'attaque permet de générer des IVs différentes et donc d'obtenir très rapidement les IVs faibles dont nous avons besoin pour obtenir la clé par la suite

  ```
  # aireplay-ng -2 -r arp-request wlan0mon
  ```

  Le nombre de #datas va augmenter très rapidement, il faut collecter environ 30000/40000 #datas, ce qui prend environ 2/3 minutes.
- Documentation : Lien

Crack de la clé WEP

- Maintenant que nous avons nos IVs faibles, il suffit de faire tourner l'algorithme qui va obtenir les octets de la clé.
- Vous pouvez fermer airodump et la fake authentication
- La commande pour cracker la clé est très simple

```
# aircrack-ng out-01.cap
```

- Si vous avez capturé suffisamment d'IVs faibles, vous obtiendrez la clé
- Tutoriel complet : Lien
- Documentation complète sur les attaques WEP : Lien

Wifite

- Wifite est un outil qui automatise l'attaque
- Il est très simple a utiliser

```
# wifite -wep
```

- Il suffit de suivre les instructions et d'attendre quelques minutes pour obtenir la clé.
- Pourquoi ne pas l'avoir utilisé dès le début ?
- Parce qu'il faut toujours comprendre ce que l'on fait !
- Documentation : Lien

Le WPA

- WPA pour Wi-Fi Protected Access est un protocole créé en 2003 pour remplacer le WEP.
- Le WPA utilise le même algorithme de chiffrement que le WEP (RC4)
- Pour combler les failles du WEP, WPA implémente le protocole TKIP pour Temporal Key Integrity Protocol
- WPA chiffre les paquets avec une clé de base qui est périodiquement modifié
- Au lieu que l'IV soit diffusé en clair sur le réseau, celle-ci est haché
- Le CRC32 est remplacé par le MIC pour Message Integrity Code (Michael)
- Documentation : Lien

TKIP n'est pas sûr

- Depuis 2008, Erik Tews et Martin Beck ont découvert une faille dans le chiffrement TKIP WPA.
- La faille permet de sniffer et d'injecter des paquets à destination d'un client et peut permettre un man in the middle.
- aircrack-ng possède un outil nommé Tkiptun-ng mais celui-ci ne semble pas complètement fonctionnel, nous ne verrons pas cette attaque ici.
- Documentation : Lien 1 Lien 2 Lien 3

Attaque par dictionnaire

- Si le mot de passe du point d'accès est faible, il est possible de l'obtenir
- Pour que l'attaque puisse réussir, il faut qu'un client soit connecté au réseau
- Tout d'abord, nous allons écouter le réseau

```
# airodump-ng wlan0mon -bssid C8:BE:19:71:C1:10 -c 1
```

Attaque par dictionnaire

- Le but étant de capturer le handshake pour que l'on puisse le bruteforcer par la suite
- Le handshake permet de prouver que l'AP et le client possèdent la même clé
- La clé n'est pas envoyée dans le handshake mais elle est vérifiée en calculant le MIC
- Avec le handshake capturé, on peut lancer une attaque par dictionnaire dessus jusqu'à ce que l'on obtienne le même MIC
- Il est donc obligatoire qu'un client soit connecté à l'AP
- Documentation : Lien

Attaque par dictionnaire

- On va forer le client à se déconnecter pour qu'il se reconnecte et ainsi capturer le handshake

  ```
  # aireplay-ng -0 10 -a C8:BE:19:71:C1:10 -c 00:C0:CA:82:C6:85 wlan0mon
  ```

- On peut vérifier dans notre fenêtre airodump que le handshake est bien capturé
- On peut fermer airodump et lancer l'attaque par dictionnaire
- Pour l'exemple, on utilisera un "dictionnaire" maison

  ```
  # aircrack-ng -w mini_dico.txt out-01.cap
  ```

- Documentation : Lien

Faiblesse du WPS

- WPS pour Wi-Fi Protected Setup permet de simplifier la connexion des clients au réseau WiFi
- La méthode PIN (Personal Identification Number) permet d'obtenir la clé WPA, elle consiste en un nombre à 8 chiffres
- Statistiquement, il y a donc peu de combinaison différente possible pour le PIN
- On peut donc le bruteforcer, cela peut prendre plusieurs heures/jours

- reaver est un des outils permettant de bruteforcer un pin

```
# reaver -i wlan0mon -b C8:BE:19:71:C1:10
```

- Certains routeurs peuvent détecter l'attaque et désactivent l'accès par WPS
- Il est possible dans les options de reaver de réduire le délai de l'attaque, mais cela prendra des années pour obtenir le PIN !
- Documentation : Lien

Le WPA2

- Le WPA2 est la version la plus récente du WPA
- Le CCMP pour Counter-Mode/CBC-Mac protocol remplace le TKIP
- Il n'existe pas à ce jour de grosse faiblesse contre le WPA2
- Néanmoins, une attaque par bruteforce comme vu précédemment est toujours possible
- Il faut donc utiliser un mot de passe fort

Northsec

- Compétition de sécurité informatique annuel à montréal fin mai
- C'est le moment de s'inscrire