Fakulta Informatiky, Masarykova Univerzita v Brně

Referát do predmetu: M0170 – Kryptografie

Technológia WEP a jej nedostatky

10. mája 2012

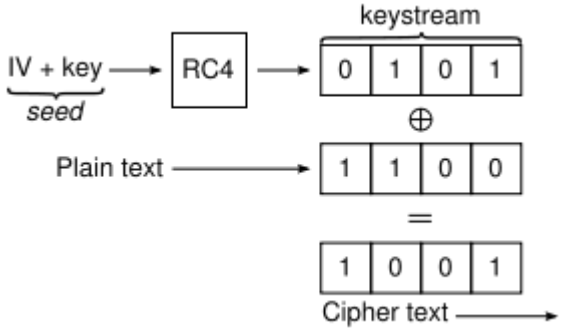
Autor: Tomáš Benčok, 395607@mail.muni.cz

1 Úvod

Technológia WEP (Wired Equivalent Privacy) vznikla ako súčasť štandartu IEEE 802.11 v roku 1999. Jedná sa o bezpečnostný protokol, ktorého úlohou je poskytnúť užívateľom bezdrôtových sietí takú úroveň bezpečnosti, ktorá by bola porovnateľná s úrovňou bezpečnosti klasických sietí typu LAN. To sa mu však nepodarilo - po zverejnení algoritmu a mechanizmov, ktoré používa na autentifikáciu užívateľov bolo odhalených jeho niekoľko nedostatkov. Tieto nedostatky sú závažné až natoľko, že útočníkovi dovoľujú prelomiť jeho ochranu už v rámci niekoľkých minút. No napriek všetkému, WEP sa stále používa.

2 Charakteristika

Existuje niekoľko variant bezpečnosti WEP-u, 64-bitová, 128-bitová a 256-bitová (najnovšia z nich, ale nepoužíva sa až tak často). Každý packet je zašifrovaný samostatne pomocou **RC4** algoritmu a skladá sa z **iniciačného vektoru** a **kľúča**. Dáta, ktoré sa posielajú budú prevedené na bity a šifrovanie sa spraví pomocou kľúča a operácie XOR, tak ako je to ukázané na obrázku.



Obr. 1: Šifrovanie pomocou operácie XOR

Iniciačný vektor (IV) sa vždy skladá z 24 bitov a to bez ohľadu na to, koľko bitová úroveň bezpečnosti sa použije. Generuje ho vždy vysielacia strana a jedná sa o dynamicku časť kľúča – v priebehu komunikácie sa môže periodicky meniť. Každý packet by teoreticky mohol byť zašifrovaný inak, ale v praxi sa používajú dlhšie intervaly.

Kľúč je naproti IV stály a nemení sa. V prípade, že sa použije 64-bitová varianta WEP-u, na kľúč samotný pripadne 40 bitov, ktoré zadá uživateľ ako 10 hexadecimálnych znakov (na každý znak (0-9, A-F) pripadnú 4 bity). Kľúč je tá tajná časť, ktorú potrebujeme poznať, alebo získať ak sa chceme do danej siete pripojiť. Zo začiatku mohla exitavať iba 64-bitová varianta kvôli zákonom platiacich v USA, no keď boli tieto nezmyselné obmedzenia zrušené, bola doimplementovaná aj 128 a 256-bitová varianta (na prelomenie potrebuje útočník odchytiť viac packetov).

K zašifrovanému textu je ešte konkatenáciou pripojený zašifrovaný 4 bitový kontrolný súčet na test integrity dát (ICV – **Integrity Check Value,** metódou CRC-32) – na zašifrovaný packet to samozrejme nemá žiadny efekt.

3 RC4

Algoritmus RC4 (Rives Cipher 4) bol navrhnutý Ronom Rivestom pracujúcim vo firme RSA Security už v roku 1987. Jedná sa o symetrický algoritmus, ktorý využíva prúdové šifrovanie – tento spôsob šifrovania je veľmi rýchly a efektívny pokiaľ spracovávame veľké množstvo dát v reálnom čase. Firma RSA Security oficiálne nikdy nezverejnila spôsob jeho implementácie a jej existencia dodnes ostáva obchodným tajomstvom. No napriek tomu je známa – v roku 1994 ju niekto odcudzil a anonymne zverejnil na Interente. Takto sa rozšírila a jednoduchý spôsob jej implementácie (či už v SW, alebo v HW) bol postupne predstavený viacerým ľuďom. Okrem zabezpečenia WEP-u bol algoritmus použitý aj v ranných technológiách SSL.

Šifra RC4 generuje pseudonáhodný prúd bajtov, ktoré sú potom spojené s textom, ktorý chceme zašifrovať. Ako už bolo napísané vyššie, toto spojenie sa robí pomocou operácie XOR - tá je symetrická, takže rovnaký (ale inverzným) spôsob je možné aj dešifrovanie.

XOR	Input 1	
	0	1
Input 2 0	0	1
1	1	0

Tab. 1: Pravdivostné hodnoty pre operáciu XOR (exkluzívne sčítanie)

Na vygenerovanie kľúčového prúdu bajtov používa algoritmus tajný vnútorný stav, ktorý sa skladá z dvoch častí:

1. Permutácia všetkých 256 možných bajtov (S)
2. Dva 8-bitové indexové ukazatele (i, j)

Permutácia je inicializovaná kľúčmi rôznej dĺžky, typicky medzi 40 a 256 bitmi pomocou algoritmu KSA. Potom sa použie pseudonáhodný algoritmus (PRGA) na vygenerovanie prúdu bitov.

Algotimus **KSA**:

```
for i from 0 to 255
      S[i] := i
endfor

j := 0
for i from 0 to 255
      j := (j + S[i] + key[i mod keylength]) mod 256
      swap values of S[i] and S[j]
endfor
```

Algoritmus **PRGA**:

```
i := 0
j := 0
while GeneratingOutput:
    i := (i + 1) mod 256
    j := (j + S[i]) mod 256
    swap values of S[i] and S[j]

    K := S[(S[i] + S[j]) mod 256]
    output K
endwhile
```

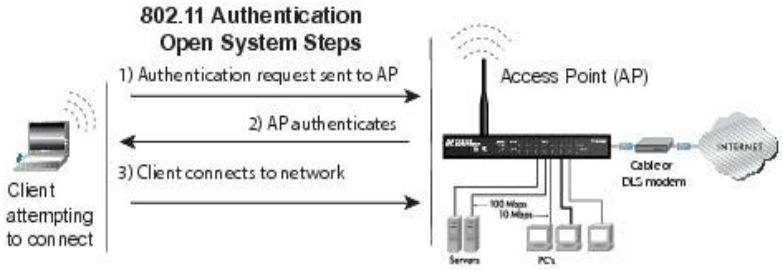
4 Autentifikácia

Existujú dva možné spôsoby akými sa možno autentifikovať do siete:

A - otovrená autentifikácia (Open System Authentification)
B - autentifikácia zdieľaným kľúčom (Shared Key Authentification)

V prvom prípade sa v podstate o žiadnu autentifikáciu nejedná – ak sieť používa takýto spôsob, môže sa do nej pripojiť ktokoľvek a bude mu poskytnuté šifrovanie. Postupnosť krokov vyzerá nasledovne:

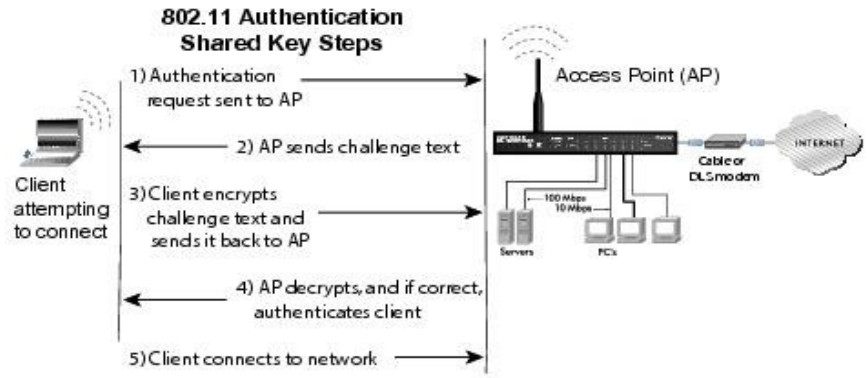
1. Stanica pošle žiadosť o pripojenie do siete prístupovému bodu (AP)
2. Prístupový bod ju authentifikuje
3. Stanica je asociovaná s access pointom a može používať sieť



Obr. 2: Open system authentification

Ako už názov druhého spôsobu napovedá, na propojenie do siete bude pracovná stanica potrebovať najprv heslo a ak bude toto správne, až potom jej bude umožnené pripojiť sa do siete. Scenár vyzerá takto:

1. Stanica pošle žiadosť o autetifikáciu prístupovému bodu (AP)
2. AP jej odpovie textovou výzvou – požiada o zašifrovanie nejakého textu
3. Stanica použije svoj 64/128 bitový kľúč na zašifrovanie textu a pošle ho AP-tu
4. AP ho dešifruje pomocou svojho kľúča a v prípade, že stanica zašifrovala text správne, AP ju autentifikuje a dovolí jej používať sieť. (Ak bol text zašifrovaný nesprávne, AP odmietne stanici prístup do siete a celý proces sa musí opakovať)
5. Stanica je odteraz asociovaná s access pointom a môže používať sieť



Obr. 3: Shared key authentification

5 Nevýhody

Mohlo by sa zdať, že 2. spôsob authentifikácie je lepší, ale v praxi je veľmi jednoduché odchytit a nazbierať dostatočné množstvo packetov (20.000-85.000), ktoré vysiela AP a ich analýzou zistiťzdieľaný kľúč. Je potrebné nazbierať hlavne rôzne iniciačné vektory. Jeden IV môže byť použitý niekoľko krát, preto bude množstvo všetkých odchytených packetov väčšie, než množstvo tých užitočných.

Pretože RC4 je prúdová šifra, žiadny kľúč by sa z dôvodu bezpečnosti nemal použiť viac než jeden krát. Ale iniciačný vektor má iba 24 bitov (16.7 miliónov možností) – a to spôsobuje problémy, pretože ak je v sieti veľa počítačov a veľká premávka, nutne dôjde k ich vyčerpaniu. Pri tejto veľkosti dokonca existuje až 50% pravdepodobnosť, že sa nejaký iniciačný vektor znovu zopakuje po každých 5000 vygenerovaniach nového vektora.

Okrem toho, niektoré z vygenerovaniach IV sú slabé a neprodukujú dostatočne náhodné dáta, z ktorých je potom možné jednoduchšie „uhádnuť“ zdieľaný kľúč. Niektorí výrobcovia z dôvodu bezpečnosti tieto hodnoty implicitne zakazujú používať, čo však na druhú stranu redukuje množinu možných vygenerovaných IV.

Z kryptografického hľadiska vôbec nie je doporučené používať hlavné heslo priamo počas komunikácie, lebo môže byť odchytené, resp. prelomené. Hlavné heslo by malo byť použité len na generovanie dočasných hesiel a tie by mali byť použité v komunikácii.

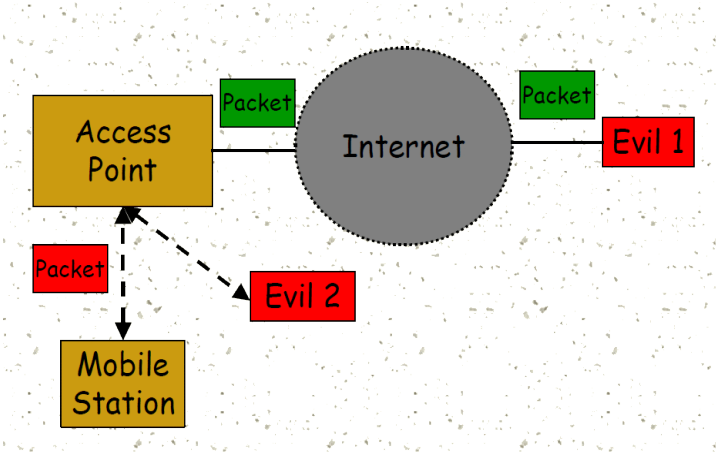
Zhrnutie nevýhod:

- dĺžka iniciačných vektorov nie je dostatočná
- iniciačné vektory sa po čase môžu opakovať
- existujú iniciačné vektory, ktoré nie sú dostatočne náhodné
- hlavné heslo je použité priamo v komunikácii
- test na integritu packetov nie je efektívny (je možné prepočítať CRC hodnotu a podvrhnúť packet znovu)

6 Útoky

Existujú rôze spôsoby akými sa dá na WEP ochranu zaútočiť a prelomiť ju. V dnešnej dobe je taktiež dostupných mnoho open source programov (väčšinou pod linuxom) a tutoriálov na webe, ktoré vám dopodrobna vysvetlia a ukážu, ako je možné nejaký útok urobiť.

- **Pasívny útok na dešifrovanie traffic-u** – útočník odchytáva packety a čaká, kým nenastane kolízia iniciačných vektorov. V takomto prípade, ak sa mu podarí dešifrovať text jednej zo správ, dokáže automaticky dešifrovať aj tú druhú.
- **Aktívny útok injektovaním packetov** – ak útočník pozná text aspoň jednej zašifrovanej správy, môže ho zmeniť, prepočítať kontrolný súčet (CRC-32) a preodiť niektoré bity. Ak takáto správa dorazí na AP, bude akceptovaná. Je to možné vďaka tomu, že RC4(X) *xor* X *xor* Y = RC4(Y).
- **Aktívny útok z oboch strán** – ak útočník odchytí packet a zistí, kde sa nachádza jeho hlavička (obsahujúca IP adresu destinácie), potom môže prehodiť relevantné bity tak, aby táto cieľová stanica posielala packety na jeho systém a on ich môže preposielať ďalej na mobilnú stanicu v sieti. Packet bude dešifrovaný na access pointe a bude ďalej preposlaný (pomocou nejakého router-u) nezašifrovane na útočníkov počítač.
- **Útok na základe tabuliek** – pretože inicializačné vektory majú len 24-bitov, útočník si môže zostaviť dešifrovaciu tabuľku. Keď zistí text niektorého z packetov, môže vypočítať RC4 kľúč pomocou inicializačného vektora a ten môže byť použitý na dešifrovanie všetkých ostatných packetov s rovnakým IV. Postupom času získa úplnú tabuľku (bude zaberať približne 15 GB) a s jej pomocou môže dešifrovať *každý* packet.



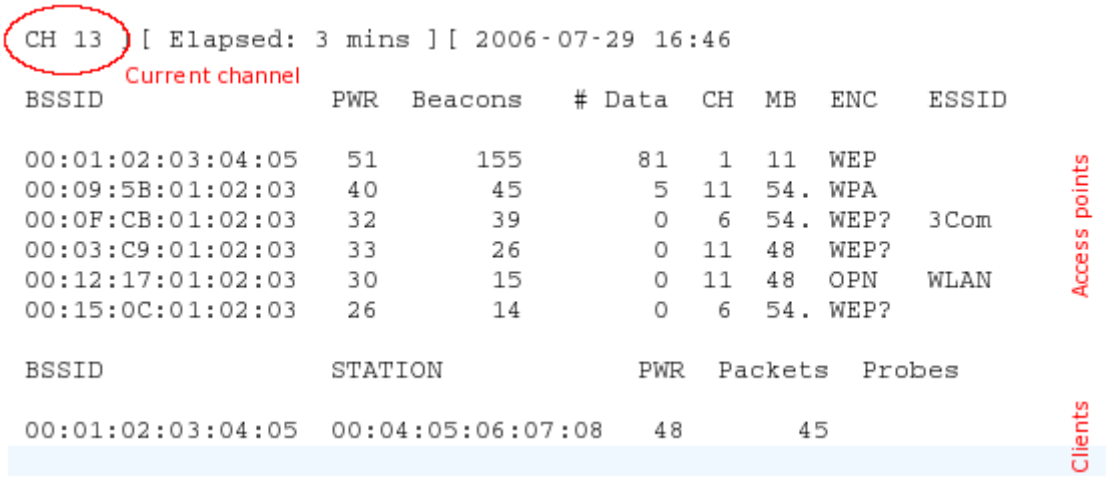
Obr. 4: Útok z oboch strán

6.1 Základné znalosti

Každý access point posiela približne 10 „beacon-ov“ za sekundu. Beacon je typ packetu, ktorý obsahuje nasledovné informácie:

- ESSID – názov siete,
- či je použité nejaké šifrovanie, prípadne aké,
- aké prenosové rýchlosti v Mbit sú podporované,
- na ako kanále pracuje sieť.

Tieto informácie je možné zistiť napr. príkazom ***airodump-ng*** *[device]*. (Kde *[device]* je názov sieťového rozhrania, napr. wlan0, prípadne mon0). Ešte pred tým je nutné dať sieťovú kartu do monitorovacieho módu – ***modprobe -r*** *[driver]* a následne ho zapnúť ***airmon-ng*** *start [device]*. Po zadaní príkazu airodump-ng dostanete takýto výstup:



Obr. 5: Výstup programu airodump-ng – siete, ktoré sú v dosahu

- BSSID – MAC adresa AP
- PWR – sila signálu, vzdialenosť od AP
- CH – kanál na ktorom beží AP
- ESSID – názov siete
- Data – počet prijatých dátových frameov

PWR a Data signalizujú, ako rýchlo je možné nazbierať potrebné packety.

6.2 Útok

Predpokladajme, že ste si už vybrali sieť, ktorá je pre vás zaujímavá. Jediné čo musíte je nazbierať dostatočné množstvo packetov a preskúmať ich. No ešte pred tým (najlepšie úplne na samom začiatku) by bolo dobré zmeniť si vašu MAC adresu na nejakú vymyslenú:

- ifconfig down [device] (wlan0, mon0)
- ***macchanger –mac*** [fake_mac] [device] (00:11:22:33:44:55)
- ***airmon-ng start***

Teraz už možte začať bezbečne zbierať packety a ukladať si ich niekam do súboru. Na úspešné cracknutie potrebujete získať 10-80.000 packetov, ktoré obsahujú rôzne inicializačné vektory. Koľko toho máte vám indikuje položka **#data.** Pokiaľ je v sieti malý traffic, môže byť celkom náročné nazbierať dostatočné množstvo dát – kľudne to môže trvať aj niekoľko hodín. Vtedy je dobré vynútiť si ich posielanie.

- ***airodump-ng -c*** *[chanel]* ***-w*** *[subor]* ***--bssid*** *[adr_ap]*

Po nazbieraní spustíte samotné crackovanie. V závislosti na použitom spôsobe to môže trvať od pár sekúnd až do niekoľkých hodín, no pokiaľ sa vám podarilo získať dostatočné množstvo IV, nebude to trvať viac než 2-3 minúty.

- ***aircrack-ng -a 1 -b*** *[bssid]* ***-l*** *[subor_s_datami]*

6.3 Poznámky

- ak sa vám podarí odchytiť prvé 4 authentifikačné packet, heslo môžete prelomiť veľmi rýchlo. Možný spôsob útoku – vynútiť si odpojenie stanice od AP (***mdk3***) a čakať na jej opätovnú authentifikáciu.
- Vynútenie trafficu:
  najprv treba zistit, či je injekcia packetov vôbec možná:
  aireplay-ng -1 0 -a *[bssid_ap]* ***-h*** *[moja_mac]* ***-e*** *[essid_siete] [device]*
  posielanie requestov:
  aireplay-ng -3 -b *[bssid_ap]* ***-h*** *[moja_mac] [device]*,
  a následné odchytenie odpovedí pomocou ***airodump-ng***

7 Záver

WEP síce nie je ideálnym bezpečnostným riešením, ale je lepšie použiť aspoň takúto ochranu, než nepoužiť žiadnu. Treba zistiť, či výrobca zamedzil použitiu slabých inicializačných vektorov, tiež treba radšej používať 128-bitové kľúče. Tiež je dobré nainštalovať si IDS (Intruder Detection System) na prípadné zistenie / monitorovanie útokov. Prípadne je možné prejsť na vyšší bezpečnostný štandart, akým je napr. WPA2-PSK.

Použitá literatúra

http://www.tech-faq.com/wep-wired-equivalent-privacy.html

http://www.openxtra.co.uk/articles/wep-weaknesses

http://www.wongchonkit.com/2011/09/wireless-hacking-wep.html

http://english.turkcebilgi.com/Wired+Equivalent+Privacy

http://www.isaac.cs.berkeley.edu/isaac/wep-faq.html

http://www.aircrack-ng.org/doku.php?id=newbie_guide

http://documentation.netgear.com/reference/fra/wireless/WirelessNetworkingBasics-3-09.html

http://documentation.netgear.com/reference/sve/wireless/WirelessNetworkingBasics-3-08.html

https://www.networkworld.com/details/715.html

https://en.wikipedia.org/wiki/Wired_Equivalent_Privacy

http://www.wireless-center.net/Wireless-Internet-Technologies-and-Applications/1938.html

http://www.isaac.cs.berkeley.edu/isaac/wep-slides.pdf

http://www.cs.sjsu.edu/~stamp/CS265/projects/Spr04/section1/presentations/Devireddy.ppt

http://netlab18.cis.nctu.edu.tw/html/wlan_course/powerpoint/RC4.pdf

http://www.mcs.sdsmt.edu/ecorwin/cryptography/2003/talks/RC4_and_WEP.ppt