

M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Híradástechnikai Tanszék

Mérési útmutató a „WIFI 1: Helyi hitelesítő eljárások elleni támadások” című méréshez



Hírközlő rendszerek biztonsága szakirány
Hírközlő rendszerek biztonsága laboratórium I.
(BMEVIHIM220)

A mérést kidolgozták:

BUTTYÁN LEVENTE, DÓRA LÁSZLÓ, LACZKÓ PÉTER
meres@crysys.hit.bme.hu



CrySyS – Adatbiztonság Laboratórium

2009. október 26.

Tartalomjegyzék

1. Mérés célja	1
2. Elméleti összefoglaló	2
2.1. Alapfoglamak	2
2.1.1. Vezeték nélküli interfész	2
2.1.2. Beacon üzenet	2
2.2. Rejtett SSID	2
2.3. MAC szűrés	3
2.4. WEP	3
2.4.1. A WEP működése	3
2.4.2. A WEP hibái	6
2.4.3. Egy komplett támadás WEP ellen	7
2.5. WPA	10
2.5.1. Ismertető	10
2.5.2. Helyi hozzáférés-védelem és kulcsmenedzsmnt	11
2.5.3. WPA gyengeségei	12
3. Mérési környezet	13
4. Feladatok	14
4.1. Csatlakozás rejtett SSID-jű AP-hoz	14
4.2. WEP elleni támadás	15
4.3. WPA gyengeségei	15
5. További információk	15
6. Minta beugró kérdések	17

1. Mérés célja

Wi-fi, WLAN, vezeték nélküli hálózat, IEEE 802.11: Ha nem is mind pontosan ugyanazt fedik le, mégis mindenki számára többnyire ugyanazt jelenti. Adott egy fix eszköz, úgynevezett access point (AP) vagy magyarul hozzáférési pont, amelyik többnyire vezeték nélküli kapcsolaton keresztül biztosítja az Internet hozzáférést az éppen hozzá kapcsolódó mobil állomások (STA) számára. Az egyik legelterjedtebb megoldást az IEEE 1999-ben kezdte szabványosítani 802.11-es kódszám alatt [1]. Ennek a megoldásnak a biztonsági vonatkozásait kívánjuk megismertetni a hallgatóval.

Hogy miért pont ezt? Egyrészt, mert ahogy már mondtuk, az egyik legelterjedtebb megoldás, tehát jó eséllyel minden informatikus találkozik vele munkája során. Másrészt pedig, elég régóta elkezdődött a szabványosítás, hogy találjanak benne hibát, amiből tanulni lehet. Azt reméljük, hogy az itt bemutatott támadásokat a hallgató általánosabban értelmezve megérti a hálózati biztonság komplexitását, és intő példaként szolgál a MAC szűrés kijátszhatósága, valamint a WEP tervezése során elkövetett hibák.

A mérés két részre oszlik, az elsőben helyi hitelesítésen alapuló megoldásokat vizsgálunk, illetve az adatforgalom rejtjelezéssel és integritás-védelemmel kapcsolatos vizsgálatokat végzünk el, míg a másik mérésben a központi hitelesítést járjuk körül. Sikeres mérés esetén a hallgató az első mérés során 1) láthatja, hogy a rejtett SSID és a MAC szűrés valójában mennyi biztonságot jelent; 2) visszafejt egy WEP jelszót; 3) megvizsgálja a WPA biztonságosságát, amin keresztül betekintést nyújtunk a brute force támadások hatékonyságába is.

2. Elméleti összefoglaló

Ebben a fejezetben áttekintjük a ma használatos WLAN biztonsági megoldásokat, valamint azok gyengeségeit. Mindezt egy rövid bevezető után, ahol néhány fogalmat és működési elvet tisztázunk.

2.1. Alapfoglamak

2.1.1. Vezeték nélküli interfész

Az IEEE 802.11-ben meghatározott teljes frekvenciasávot csatornákra osztották. A jelenlegi hardver többnyire csak egy csatornán való kommunikálást engedélyez. Ezért minden esetben ki kell választani, hogy melyik csatornán akarunk küldeni vagy hallgatni.

Normális esetben ez automatikusan és észrevétlenül történik. Normális esetnek tekinthető az, amikor egy STA egy adott AP-hoz kapcsolódik. Ilyenkor az STA beállítja a megfelelő csatornát és attól kezdve csak egy adott frekvenciatartományban értelmezi a jeleket. Ráadásul, ha nem promiscuous módban van a kártya, akkor azokat a csomagokat, amelyeknek nem mi vagyunk a címzettjei, eldobásra kerülnek. Ezzel szemben promiscuous módban minden olyan csomag megjelenik az interfészen, amelyik attól az AP-től érkezik, amelyikhez kapcsolódott a STA, függetlenül attól, hogy ki a címzett.

Még nagyobb szabadságot biztosít a monitor mód, amelyik nem igényel AP-hoz való kapcsolódást, és ennek megfelelően minden olyan csomagot megjelenít az interfészen, amelyek 802.11 fejléccel rendelkeznek. Itt már CRC vizsgálat sem történik, tehát megjelenhetnek sérült csomagok is.

Mint látható monitor módban a célcsoport nem egyértelmű, tehát azt a felhasználónak kell beállítani.

2.1.2. Beacon üzenet

Az AP periodikusan küld egy beacon üzenetet minden egyes csatornán, hogy a hatósugarában lévő STA-k számára tudtukra adja jelenlétét. Ezekben az üzenetekben az AP különböző információkat oszt meg úgy, mint a kommunikációra használt csatorna, támogatott átviteli sebesség, biztonsági mechanizmus vagy az SSID.

Az SSID (Service Set ID) egy azonosító, ami egy hálózati eszközcsoportot határoz meg. Az STA-k SSID alapján tudják megkülönböztetni a vezeték nélküli hálózatokat. Amikor több AP ugyanahhoz a hálózathoz tartozik, ugyanazt az SSID-t kapják. Ilyenkor a BSSID-vel (Basic SSID) lehet megkülönböztetni őket, ami egy egyedi azonosító, és többnyire a MAC címmel egyezik meg.

Amikor egy STA csatlakozik az AP-hoz, az asszociációs csomagba be kell illeszteni az AP SSID-jét is. Ekkor az AP ellenőrzi, hogy a kapott SSID megegyezik-e a saját SSID-jével. Amennyiben nem, az asszociációs kérést visszautasítja.

2.2. Rejtett SSID

A SSID-t ugyan lehet, de nem kötelező a beacon üzenetbe beilleszteni. Amennyiben ez nem történik meg, akkor beszélünk rejtett SSID-jű AP-ról. Mivel ilyenkor csak az az eszköz tud csatlakozni, amelyik ismeri a SSID, az SSID tekinthető megosztott közös titoknak is. Ezt az eljárást szokás biztonsági megoldásként használni.

Csakhogy csatlakozáskor az SSID-t nyíltan küldi át az azt ismerő STA, ezért egy támadó egy asszociációs üzenet lehallgatásával az SSID birtokába jut. Ezzel már a támadó is képes csatlakozni az AP-hoz. Ráadásul egy támadó képes küldeni egy deauthentication üzenetet egy legális STA nevében. Így a legális STA leválasztásra kerül, amelyik nagy valószínűséggel

automatikusan visszacsatlakozik. Ezzel nyíltan átküldi az SSID-t, amit a támadó azonnal megszerez.

2.3. MAC szűrés

A MAC szűrés tulajdonképpen már a vezetékes világból is ismert eljárás, és független az IEEE 802.11 szabványtól. MAC szűrés esetén az AP fenntart egy listát, hogy mely MAC című STA-k jogosultak a hálózat használatára. Amennyiben egy olyan STA küld csomagot, amely nincs a listán, az AP egyszerűen figyelmen kívül hagyja a csomagot.

A MAC cím a gyártó által a hálózati eszközbe beégetett egyedi kód. Azonban ezt a kódot szoftveresen át lehet állítani tetszőleges értékre, így akár egy olyan eszközére is, amelyik jogosult az AP-hoz való csatlakozásra. Ha egy támadó sikeresen szerez egy ilyen MAC címet, képes lesz csatlakozni az AP-hoz. Azonban, mivel minden elküldött csomagba bekerül forrásként vagy célként a csatlakozott STA MAC címe, már egy csomag elfogásával megszerezhető a cím.

Bizonyos szempontból ez a megoldás annyi biztonságot sem nyújt, mint a rejtett SSID, mivel ott legalább csak csatlakozáskor küldték át nyílt formában a közös titkot. Itt azonban minden csomagban megjelenik a cím.

Fontos megjegyezni viszont, hogy egyszerre két eszköz azonos címmel nem csatlakozhat egy AP-hoz címütközés miatt. Éppen ezért a sikeres csatlakozáshoz meg kell várni, hogy a jogosult eszköz lekapcsolódjon az AP-ról, vagy hamis deauthentication csomaggal lehet erőszakkal leválasztani. Azonban ilyenkor megeshet, hogy az STA megpróbál automatikusan visszacsatlakozni.

2.4. WEP

Az IEEE 802.11 vezeték nélküli LAN szabvány tervezői kezdetől fogva fontosnak tartották a biztonságot. Ezért már a 802.11 korai verziója [1] is tartalmazott biztonsági mechanizmusokat, melyek összességét WEP-nek (Wired Equivalent Privacy) nevezték el. Ahogy arra a név is utal, a WEP célja az, hogy a vezeték nélküli hálózatot legalább olyan biztonságossá tegye, mint egy - különösebb biztonsági kiegészítésekkel nem rendelkező - vezetékes hálózat. Ha például egy támadó egy vezetékes Ethernet hálózathoz szeretne csatlakozni, akkor hozzá kell férnie az Ethernet hub-hoz. Mivel azonban a hálózati eszközök általában fizikailag védve, zárt szobában találhatóak, ezért a támadó nehézségekbe ütközik. Ezzel szemben egy védelmi mechanizmusokat nélkülöző vezeték nélküli LAN-hoz való hozzáférés - a rádiós csatorna nyitottsága miatt - triviális feladat a támadó számára. A WEP ezt a triviális feladatot hivatott megnehezíteni. Fontos azonban megjegyezni, hogy a WEP tervezői nem törekedtek "tökéletes" biztonságra, mint ahogy a zárt szoba sem jelent tökéletes védelmet egy Ethernet hub számára.

A tervezők tehát nem tették túl magasra a léceket, ám a WEP még ezt a korlátozott célt sem érte el. Pár évvel a megjelenése után, a kriptográfusok és az IT biztonsági szakemberek súlyos biztonsági hibákat találtak a WEP-ben [2, 3, 4], és nyilvánvalóvá vált, hogy a WEP nem nyújt megfelelő védelmet. A felfedezést tett követte, és hamarosan megjelentek az Interneten a WEP feltörését automatizáló programok. Válaszul, az IEEE új biztonsági architektúrát dolgozott ki, melyet a 802.11 szabvány *i* jelzésű kiegészítése tartalmaz [5]. A 802.11i elosztott hitelesítéssel kapcsolatos részét a következő mérésen, a helyi hitelesítéssel kapcsolatos területeit a következő fejezetben tárgyaljuk. Ebben a fejezetben a WEP működését és hibáit tekintjük át.

2.4.1. A WEP működése

Vezeték nélküli hálózatok esetében két alapvető biztonsági probléma merül fel. Egyrészt a rádiós csatorna jellege miatt a kommunikáció könnyen lehallgatható. Másrészt — s ez

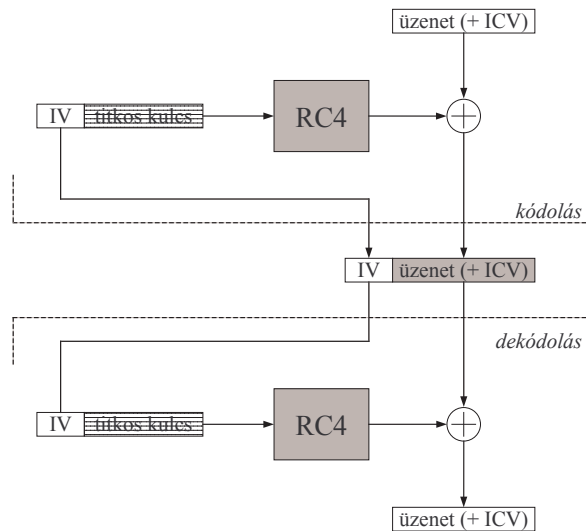
talán fontosabb — a hálózathoz való csatlakozás nem igényel fizikai hozzáférést a hálózati csatlakozóponthoz (Access Point, vagy röviden AP), ezért bárki megpróbálhatja a hálózat szolgáltatásait illegálisan igénybe venni. A WEP az első problémát az üzenetek rejtjelezéssel igyekszik megoldani, a második probléma megoldása érdekében pedig megköveteli a csatlakozni kívánó mobil eszköz (Station, vagy röviden STA) hitelesítését az AP felé.

A hitelesítést egy egyszerű kihívás-válasz alapú protokoll végzi, mely négy üzenet cseréjéből áll. Elsőként a STA jelzi, hogy szeretné hitelesíteni magát (authenticate request). Válaszul az AP generál egy véletlen számot, s azt kihívásként elküldi a STA-nak (authenticate challenge). A STA rejtjelezi a kihívást, s az eredményt visszaküldi az AP-nak (authenticate response). A STA a rejtjelezést egy olyan titkos kulccsal végzi, melyet csak a STA és az AP ismer. Ezért ha az AP sikeresen dekódolja a választ (azaz a dekódolás eredményeként visszakapja saját kihívását), akkor elhiszi, hogy a választ az adott STA állította elő, hiszen csak az ismeri a helyes válasz generálásához szükséges titkos kulcsot. Más szavakkal, a válasz sikeres dekódolása esetén az AP hitelesítette a STA-t, és ennek megfelelően dönthet arról, hogy a csatlakozást engedélyezi vagy sem. A döntésről az AP a protokoll negyedik üzenetében tájékoztatja a STA-t (authenticate success vagy failure).

Miután a hitelesítés megtörtént, a STA és az AP üzeneteiket rejtjelezve kommunikálnak. A rejtjelezéshez ugyanazt a titkos kulcsot használják, mint a hitelesítéshez. A WEP rejtjelező algoritmus az RC4 kulcsfolyam kódoló. A kulcsfolyam kódolók úgy működnek, hogy egy kis méretű, néhány bájtos titkos kulcsból egy hosszú álvéletlen bájtsorozatot állítanak elő, és ezen sorozat bájtjait XOR-olják az üzenet bájtjaihoz. Ez történik a WEP esetében is. Az M üzenet küldője (a STA vagy az AP) a titkos kulccsal inicializálja az RC4 kódolót, majd az RC4 által előállított K álvéletlen bájtsorozatot XOR-olja az üzenethez. Az $M \otimes K$ rejtjelezett üzenet vevője ugyanazt teszi mint a küldő: a titkos kulccsal inicializálja az RC4 algoritmust, amely így ugyanazt a K álvéletlen bájtsorozatot állítja elő, amit a rejtjelezéshez használt a küldő. Ezt a rejtjelezett üzenethez XOR-olva — az XOR művelet tulajdonságai miatt - a vevő az eredeti üzenetet kapja vissza: $(M \otimes K) \otimes K = M$.

A fent leírtak majdnem megfelelnek a valóságnak, van azonban még valami amit a WEP rejtjelezés kapcsán meg kell említeni. Könnyen látható, hogy ha a rejtjelezés a fent leírtak szerint működne, akkor minden üzenethez ugyanazt a K álvéletlen bájtsorozatot XOR-olnánk, hiszen a kódolót minden üzenet elküldése előtt ugyanazzal a titkos kulccsal inicializáljuk. Ez több szempontból is hiba lenne. Tegyük fel például, hogy egy támadó lehallgat két rejtjelezett üzenetet, $M_1 \otimes K$ -t és $M_2 \otimes K$ -t. A két rejtjelezett üzenetet XOR-olva, a támadó a két nyílt üzenet XOR összegét kapja: $(M_1 \otimes K) \otimes (M_2 \otimes K) = M_1 \otimes M_2$. Ez olyan, mintha az egyik üzenetet a másik üzenettel, mint kulcsfolyammal rejtjeleztük volna. Ám ebben az esetben M_1 és M_2 nem álvéletlen bájtsorozatok. Valójában tehát $M_1 \otimes M_2$ egy nagyon gyenge rejtjelezés, és a támadó az üzenetek statisztikai tulajdonságait felhasználva könnyen meg tudja fejteni mindkét üzenetet.

További példaként tegyük fel, hogy a támadó lehallgat egy rejtjelezett üzenetet, $M \otimes K$ -t, ahol M -et teljes egészében, vagy részben ismeri. Ez nem irreális feltételezés, hiszen az M üzenet tartalmazza a felsőbb szintű protokollok fejléceit, melyeknek mezői többnyire ismertek lehetnek a támadó számára. Az egyszerűség kedvéért most tegyük fel, hogy a támadó a teljes M üzenetet ismeri. Ekkor a megfigyelt $M \otimes K$ és az ismert M üzenet XOR összege pont K -t adja: $(M \otimes K) \otimes M = K$. Ennek ismeretében azonban a támadó a további rejtjelezett üzeneteket egyszerűen dekódolni tudja. Ha nem a teljes M üzenetet ismeri a támadó, akkor csak K egy részéhez jut hozzá, ám a további rejtjelezett üzeneteket ekkor is részben dekódolni tudja. A fenti problémák elkerülése érdekében, a WEP nem egyszerűen a titkos kulcsot használja a rejtjelezéshez, hanem azt kiegészíti egy IV-nek (Initialization Vector) nevezett értékkel, mely üzenetenként változik. A rejtjelezés folyamata tehát a következő: az IV-t és a titkos kulcsot összefűzzük, a kapott értékkel inicializáljuk az RC4 kódolót, mely előállítja az álvéletlen bájtsorozatot, amit az üzenethez XOR-olunk. A dekódolás folyamata ezzel analóg.



1. ábra. A WEP rejtjelezés folyamata

Ebből következik, hogy a vevőnek szüksége van a kódolásnál használt IV-re. Ez a rejtjelezett üzenethez fűzve, nyíltan kerül átvitelre. Ez elvileg nem jelent problémát, mert az üzenet dekódolásához csupán az IV ismerete nem elegendő, ahhoz a titkos kulcsot is ismerni kell. A méreteket illetően megemlítjük - s ennek később még lesz jelentősége - hogy az IV 24 bites, a titkos kulcs pedig (általában) 104 bites¹. A WEP rejtjelezés teljes folyamatát az 1. ábra szemlélteti.

Az 1. ábra azt is mutatja, hogy a rejtjelezés előtt, a küldő egy integritás-védő ellenőrző összeggel (Integrity Check Value, vagy röviden ICV) egészíti ki a nyílt üzenetet, melynek célja a szándékos módosítások detektálásának lehetővé tétele a vevő számára. A WEP esetében az ICV nem más mint a nyílt üzenetre számolt CRC érték. Mivel azonban a CRC önmagában nem véd a szándékos módosítások ellen (hiszen egy támadó a módosított üzenethez új CRC értéket tud számolni), ezért a WEP a CRC értéket is rejtjelezi. A mögöttes gondolat az, hogy így a támadó nem tudja manipulálni az üzeneteket, hiszen a titkos kulcs hiányában nem tudja a módosított üzenethez tartozó rejtjelezett CRC értéket előállítani. Mint azt alább látni fogjuk, ez a gondolatmenet nem teljesen hibamentes.

Végezetül a WEP kulcsokról szólunk röviden. A szabvány lehetővé teszi, hogy minden STA-nak saját titkos kulcsa legyen, amit csak az AP-vel oszt meg. Ez azonban megnehezíti a kulcsmenedzsmentet az AP oldalán, mivel ekkor az AP-nek minden STA kulcsát ismernie és gondoznia kell. Ezért a legtöbb implementáció nem támogatja ezt a lehetőséget. A szabvány előír egy ún. default kulcsot is, amit az AP és a hálózathoz tartozó *minden* STA ismer. Eredetileg ezt a kulcsot azon üzenetek védelmére szánták, melyeket az AP többszórással (broadcast) minden STA-nak el szeretne küldeni. A legtöbb WEP implementáció azonban csak ezt a megoldást támogatja. Így a gyakorlatban, egy adott hálózathoz tartozó eszközök egyetlen közös kulcsot használnak titkos kulcsként. Ezt a kulcsot manuálisan kell telepíteni a mobil eszközökben és az AP-ben. Nyilvánvaló, hogy ez a megoldás csak egy külső támadó ellen biztosítja a kommunikáció biztonságát; az eszközök dekódolni tudják egymás üzeneteit.

¹Különböző marketing anyagokban ezt gyakran úgy interpretálják, hogy a WEP „128 bites biztonságot” nyújt. Ez természetesen félrevezető (mint a marketing anyagok általában), hiszen a 128 bitből 24 nyíltan kerül átvitelre

2.4.2. A WEP hibái

A WEP tulajdonképpen a rossz protokolltervezés mintapéldája. Az alábbi tömör összefoglalóból látható, hogy lényegében egyetlen kitűzött biztonsági célt sem valósít meg tökéletesen:

Hitelesítés. A WEP hitelesítési eljárásának több problémája is van. Elsőként mindjárt az, hogy a hitelesítés egyirányú, azaz a STA hitelesíti magát az AP felé, ám az AP nem hitelesíti magát a STA felé. Másodsor, a hitelesítés és a rejtjelezés ugyanazzal a titkos kulccsal történik. Ez azért nem kívánatos, mert így a támadó mind a hitelesítési, mind pedig a rejtjelezési eljárás potenciális gyengeségeit kihasználhatja egy, a titkos kulcs megfejtésére irányuló támadásban. Biztonságosabb lenne, ha minden funkcióhoz külön kulcs tartozna.

A harmadik probléma az, hogy a protokoll csak a hálózathoz történő csatlakozás pillanatában hitelesíti a STA-t. Miután a hitelesítés megtörtént és a STA csatlakozott a hálózathoz, bárki küldhet a STA nevében üzeneteket annak MAC címét használva. Úgy tűnhet, hogy ez annyira nem nagy gond, hiszen a támadó, a titkos kulcs ismeretének hiányában, úgysem tud helyes rejtjelezett üzenetet fabrikálni, amit az AP elfogad. Ám ahogy azt korábban említettük, a gyakorlatban az összes STA egy közös titkos kulcsot használ, s így a támadó megteheti azt, hogy egy STA₁ által küldött - és a támadó által lehallgatott - rejtjelezett üzenetet STA₂ nevében megismétel az AP felé; ezt az AP el fogja fogadni.

A negyedik probléma egy gyöngyszem a protokolltervezési hibák között. Emlékeztetünk arra, hogy a WEP rejtjelezési algoritmus az RC4 folyamkódoló. Nemcsak az üzeneteket kódolják az RC4 segítségével, hanem a STA ezt használja a hitelesítés során is az AP által küldött kihívás rejtjelezésére. Így a támadó a hitelesítés során küldött üzenetek lehallgatásával könnyen hozzájut a C kihíváshoz és az arra adott $R = C \otimes K$ válaszhoz, melyből $C \otimes R = K$ alapján azonnal megkapja az RC4 algoritmus által generált K álvéletlen bájt sorozatot. A játéknak ezzel vége, hiszen K segítségével a támadó bármikor, bármilyen kihívásra helyes választ tud generálni a STA nevében (s ezen az IV használata sem segít, mert az IV-t a rejtjelezett üzenet küldője, jelen esetben a támadó választja). Sőt, mivel a gyakorlatban minden, az adott hálózathoz tartozó eszköz ugyanazt a titkos kulcsot használja, a támadó ezek után bármelyik eszköz nevében csatlakozni tud a hálózathoz. Persze a csatlakozás önmagában még nem elegendő, a támadó használni is szeretné a hálózatot. Ehhez olyan üzeneteket kell fabrikálnia, amit az AP elfogad. A rejtjelezés követelménye miatt ez nem triviális feladat (hiszen magához a titkos kulcshoz még nem jutott hozzá a támadó), de a WEP hibáinak tárháza bőven tartogat még lehetőségeket.

Integritás-védelem. A WEP-ben az üzenetek integritásának védelmét az üzenetekhez csatolt ellenőrző összeg (ICV) hivatott biztosítani. Az ICV nem más, mint az üzenetre számolt CRC érték, mely az üzenettel együtt rejtjelezésre kerül. Formális jelöléseket használva, a rejtjelezett üzenet a következő módon írható fel: $(M||CRC(M)) \otimes K$, ahol M a nyílt üzenet, K az RC4 által az IV-ből és a titkos kulcsból előállított álvéletlen bájt sorozat, $CRC(.)$ jelöli a CRC függvényt, és $||$ jelöli az összefűzés műveletét. Ismeretes, hogy a CRC lineáris művelet az XOR-ra nézve, azaz $CRC(X \otimes Y) = CRC(X) \otimes CRC(Y)$. Ezt kihasználva, a támadó a rejtjelezett WEP üzenetek bármely bitjét módosítani tudja (át tudja billenteni), bár magát az üzenetet nem látja. Jelöljük a támadó szándékolt módosításait ΔM -mel. Ekkor a támadó az $((M \otimes \Delta M)||CRC(M \otimes \Delta M)) \otimes K$ rejtjelezett üzenetet szeretné előállítani az eredetileg megfigyelt $(M||CRC(M)) \otimes K$ rejtjelezett üzenetből. Ehhez egyszerűen $CRC(\Delta M)$ -et kell kiszámolnia, majd a $M||CRC(\Delta M)$ értéket kell az eredeti rejtjelezett üzenethez XOR-olnia. A következő egyszerű levezetés mutatja, hogy ez miért vezet célra:

$$\begin{aligned}
& ((M||CRC(M)) \otimes K) \otimes (\Delta M||CRC(\Delta M)) = \\
& ((M \otimes \Delta M)|| (CRC(M) \otimes CRC(\Delta M))) \otimes K = \\
& ((M \otimes \Delta M)||CRC(M \otimes \Delta M)) \otimes K
\end{aligned}$$

ahol az utolsó lépésben kihasználtuk a CRC linearitását. Mivel $CRC(\Delta M)$ kiszámolásához nincs szükség a titkos kulcsra, ezért láthatóan a támadó könnyen tudja manipulálni a WEP üzeneteket, az integritás-védelem és a rejtjelezés ellenére.

Az üzenetfolyam integritásának védelme kapcsán szokás említeni az üzenetvisszajátzás detektálását, mint biztonsági követelményt. A WEP esetében ennek vizsgálatával egyszerű dolgunk van, mert a WEP-ben egyáltalán nincs semmilyen mechanizmus mely az üzenetek visszajátzásának detektálását lehetővé tenné. A tervezők nemes egyszerűséggel erről a biztonsági követelményről megfeledkeztek. A támadó tehát bármely eszköz korábban rögzített üzenetét vissza tudja játszani egy későbbi időpontban, s ezt a WEP nem detektálja. Nyilvánvaló, hogy ez miért gond, ha arra gondolunk, hogy a rögzített üzenet akár egy bejelentkezési folyamatból is származhat, s például egy felhasználói név/jelszó párt tartalmazhat.

Titkosítás. Mint azt korábban említettük, folyamkódoló használata esetén fontos, hogy minden üzenet más kulccsal legyen rejtjelezve. Ezt a WEP-ben az IV használata biztosítja; sajnos nem teljesen megfelelő módon. A probléma abból adódik, hogy az IV csak 24 bites, ami azt jelenti, hogy kb. 17 millió lehetséges IV van. Egy WiFi eszköz kb. 500 teljes hosszúságú keretet tud forgalmazni egy másodperc alatt, így a teljes IV teret kb. 7 óra leforgása alatt kimeríti. Azaz 7 óránként ismétlődnek az IV értékek, s ezzel az RC4 által előállított álvéletlen bájtsorozatok is. A problémát súlyosbítja, hogy a gyakorlatban minden eszköz ugyanazt a titkos kulcsot használja, potenciálisan különböző IV értékekkel, így ha egyszerre n eszköz használja a hálózatot, akkor az IV ütközés várható ideje a 7 óra n -ed részére csökken. Egy másik súlyosbító tényező, hogy sok WEP implementáció az IV-t nem véletlen értékről indítja, hanem nulláról. Ezért beindítás után a különböző eszközök ugyanazt a nullától induló és egyesével növekvő IV sorozatot használják, legtöbbször ugyanazzal a közös titkos kulccsal. Azaz, a támadónak várakoznia sem kell, azonnal IV ütközésekhez jut.

A WEP teljes összeomlását az RC4 kódoló nem megfelelő használata okozza. Ismeretes, hogy léteznek ún. gyenge RC4 kulcsok, melyekre az a jellemző, hogy belőlük az RC4 algoritmus nem teljesen véletlen bájtsorozatot állít elő [6]. Ha valaki meg tudja figyelni egy gyenge kulcsból előállított bájtsorozat első néhány bájtyát, akkor abból következtetni tud a kulcsra. Ezért a szakemberek azt javasolják, hogy az RC4 által előállított bájtsorozat első 256 bájtyát mindig dobjuk el, s csak az utána generált bájtokat használjuk a rejtjelezéshez. Ezzel a gyenge kulcsok problémáját meg lehetne oldani. Sajnos a WEP nem így működik. Ráadásul a változó IV érték miatt előbb-utóbb biztosan gyenge kulcsot kap a kódoló, s az IV nyílt átvitele miatt, erről a támadó is tudomást szerezhet. Ezt kihasználva, néhány kriptográfus olyan támadó algoritmust konstruált a WEP ellen, melynek segítségével a teljes 104 bites titkos kulcs néhány millió üzenet lehallgatása után nagy valószínűséggel megfejthető. A WEP minden korábban leírt hibája eltölpül ezen eredmény mellett, ugyanis ezzel a támadással magához a titkos kulcsra jut hozzá a támadó. Ráadásul a támadás könnyen automatizálható, és néhány „segítőkéz” embernek köszönhetően, az Internetről letöltött támadó programok használatával amatőrök által is rutinszerűen végrehajtható.

2.4.3. Egy komplett támadás WEP ellen

Miután a hallgató megismerkedett a WEP működésével, és ismeri a főbb hibáit, itt az ideje egy komplex, gyors és sikeres támadás bemutatásának. Ahogy látható, az eddig bemutatott gyengeségek amellet, hogy kiaknázásával csak részsikereket érünk el, sokszor olyan

feltételezésen alapulnak (pl. ismert nyílt szöveg részlet, néhány millió elfogott csomag), amelyek jelenlétének felismerése nem könnyű feladat, és a WEP-től teljesen független. A hamarosan bemutatásra kerülő támadás bemutatása során egy nem mellékes cél, hogy a hallgatóval érzékeltesük, hogy a gyengeségek, melyekre önmagukban vizsgálva talán sokan könnyen legyintenek is, miként állnak össze egy sikeres és nem mellesleg gyors támadássá.

A támadás kivitelezéséhez, melyben nem kisebb célt kívánunk elérni, mint a WEP-es jelszó visszafejtése, a szerzők vezetékneveinek kezdőbetűiről elnevezett FMS módszert [6] használjuk. Az FMS módszer az RC4 használatából adódó gyengeségek kihasználása mellett statisztikai eszközöket is bevet. A statisztikai módszereket csak nagy számú mintán lehet alkalmazni, éppen ezért sok különböző IV-hez tartozó rejtjelezett üzenetet kell elfogni. Mindez nagyon hosszadalmas lehet, ha nincs nagy forgalom a hálózaton. „Szerencsére” a WEP többi gyengesége lehetőséget ad a mesterséges forgalomgenerálásra. Az egyik út a sikeres támadáshoz a következő lépéseken át vezet. 1) Először is egy bármilyen IV-hez tartozó 1500 bájtos kulcsfolyamot kell szerezni, hogy általunk generált üzenetet tudjunk visszainjektálni a hálózatba. 1500 bájttal azért elegendő, mert az Interneten a IP csomagokat általában 1500 bájtos darabokban szállítanak tovább (MTU), és ettől nemigen szoktak eltérni kisebb hálózatok esetén sem. 2) Ezek után egy olyan üzenetet kell generálni, amire tudjuk, hogy mit válaszol az AP, és mivel azt egy másik IV-vel küldi, ezért megismerünk egy másik IV-hez tartozó kulcsfolyamot. Ennek köszönhetően újabb üzeneteket tudunk generálni, miközben újabb IV-khez tartozó üzeneteket kapunk.

A következőkben részletesebben áttekintjük, hogy miként kivitelezhetőek ezek a résztámadások. A megértést segíti a 2. ábra. További részletekhez a *The Final Nail in WEP's Coffin* című cikket [7] ajánlom.

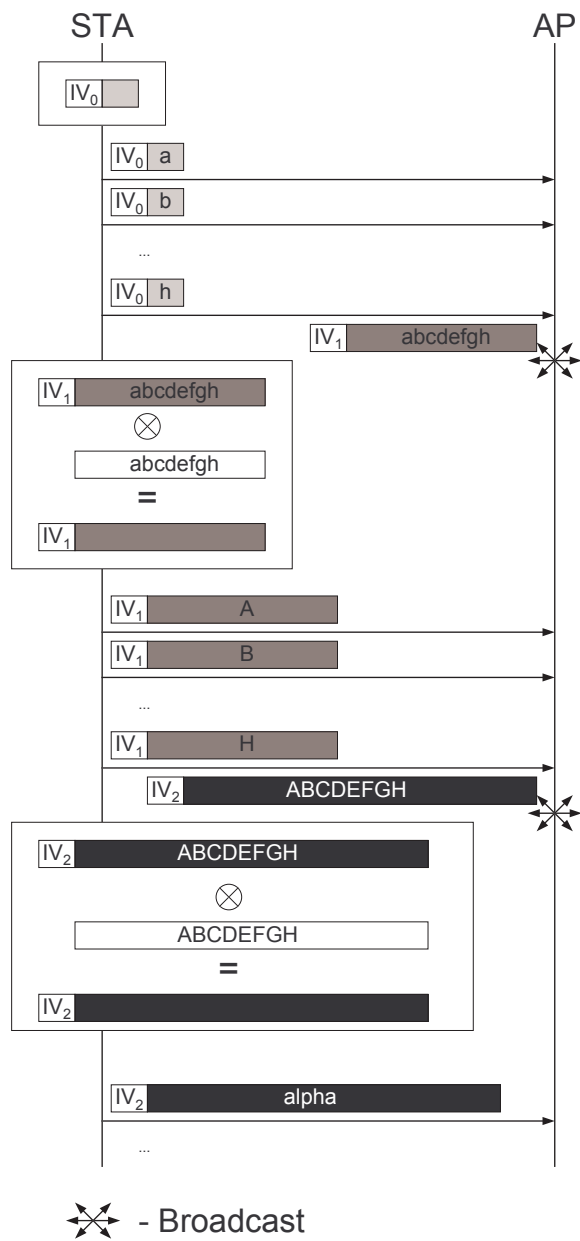
Első lépésként egy adott IV-hez tartozó kulcsfolyam első 8 bájttját kell megszerezni. Ez azért könnyű feladat, mert szinte minden 802.11 adatcsomag rejtjelezett részének első 8 bájttja egy úgynevezett LLC/SNAP fejléc. Ennek a fejlécnek köszönhetően lehet ARP és IP csomagokat küldeni a WLAN hálózaton keresztül. Az első 7 bájtt adott, a 8. pedig kétféle értéket vehet fel: ARP esetén 0x06, IP esetén 0x00. A valóságban a csomag méretéből következtetni lehet, hogy ARP vagy IP csomaggal van dolgunk, mert az ARP üzenetek mindig fixen 36 bájtt hosszúak, az IP csomagok pedig ettől valószínűleg eltérőek.

Tehát az első 8 bájttot rejtjelezett üzenetből az ismertnek tekinthető LLC/SNAP fejléccel egyszerűen XOR-olva megkapjuk a kulcsfolyamkódoló első 8 bájttját. Ez önmagában még nyilvánvalóan nem elegendő üzenet küldésre. Csakhogy a WEP megengedi, hogy adatkapcsolat rétegbeli üzeneteket daraboljunk, úgy hogy közben ugyanazt az IV-t használjuk. Erre azonban van egy korlát, legfeljebb 16 azonos IV-jű üzenetet fogad el az AP. Így összesen $4 \times 16 = 64$ bájttos üzenetet tudunk küldeni, mivel minden egyes üzenetdarab 8 bájttjából az utolsó 4 bájttot elfoglalja a CRC kód.

A következő lépéshez ismerni kell a broadcast üzenetek küldési mechanizmusát. Amennyiben egy STA_0 akar üzenetet küldeni az AP-hoz kapcsolódó minden másik STA_i -nak, azt csak az AP-n keresztül tudja megtenni. Ennek oka, hogy létezik egy másik olyan STA_i , aki STA_0 hatósugarán kívül tartózkodik, de az AP-t mindenképpen eléri. Tehát az STA_0 -tól érkező broadcast üzenetet csak az AP dolgozza föl, aki viszont újra küldi azt. Az AP az üzenet küldésekor az STA_0 -t jelöli meg forrásként. Ezt azért fontos kiemelni, mert lehallgatáskor nem lehet megkülönböztetni az STA_0 és az AP által küldött csomagokat.

A támadás a következő módon folytatható. A támadó a darabokban elküldött 64 bájttban egy broadcast üzenetet küld (abcdefgh a 2. ábrán), amit az AP először összerak, majd új IV-vel ellátva újraküldi. A támadó pontosan tudja, hogy mi a küldött üzenet teljes tartalma, hiszen ő állította össze, viszont az AP nem darabolva küldi, hanem egyben, amihez a 4 bájttos CRC-t hozzáteszi, amit viszont most már a támadó is ki tud számolni. Tehát a támadó így már 68 bájttnyi egy másik IV-hez tartozó kulcsfolyamot ismer.

A darabolást kihasználó támadás immár nagyobb üzenetdarabokkal megismételhető (pl.



2. ábra. Egy IV és a hozzá tartozó kulcs megszerzésének lépései

ABCDEFGH a 2. ábrán). 16 darab 64 bájtos (CRC elfoglal minden darabban 4 bájtot) üzenet elküldése után az AP egy 1028 bájtos ($64 \times 16 + 4$) csomagot broadcastol. További két üzenet küldésével egy komplett 1500 bájtos adott IV-hez tartozó kulcsfolyam áll rendelkezésre.

Ezek után könnyű további IV-khez tartozó kulcsfolyamokat szerezni, mivel már tudunk komplett broadcast csomagokat rejtjelezetten küldeni az AP számára. Fontos megjegyezni, hogy a támadás akkor lehet sikeres, ha az AP új IV-t használ, amikor újraküldi a csomagokat. Ez az esetek igen jelentős többségében azonban fennáll.

Miután számos csomagot elfogtunk különböző IV-kkel, az FMS módszerrel a kulcs visszafejthető. Ahogy korábban már említettük, az FMS módszer nem determinisztikus módon fejt vissza a kulcsot, mivel sok esetben egy lefutás során téved, és hibás eredményt ad. A támadás megalkotói ezt úgy kezelték, hogy sok mintára futtatják le a visszafejtő algoritmust, és a szerzők bebizonyították, hogy a jó kulcsot nagyobb valószínűséggel adja vissza az általuk javasolt algoritmus, mint bármely más kulcsot.

Az FMS támadásban a kulcsot bájtonként vizsgálják, és minden bájtra több részkulcsjelöltet tartanak fenn, amikhez szavazatokat rendeltek. Egy kulcsjelölt minden egyes bájtra akkor kap egy szavazatot helyiértéknek megfelelően, ha a visszafejtő algoritmus egy rejtjelezett szövegre azt hozza ki, hogy az a kulcsjelölt lehetett a kulcs. Ezek után az FMS ellenőrzi a kulcsjelölteket, azaz végignézi, hogy egy adott kulcsjelölttel valóban dekódolni lehet-e az üzeneteket úgy, hogy a CRC kód megfelelő legyen. Az FMS azokat a kulcsjelölteket vizsgálja, amelyek egy bizonyos hányadát érték el a legtöbbit elérő kulcsjelölt szavazatainak. Ezért minél pontosabb az elsődleges kulcsjelölt annál kevesebb kulcsjelöltet kell megvizsgálni, és annál gyorsabb az algoritmus. Tehát minél több IV-jű csomagot sikerül elfogni, annál gyorsabb a feltörő algoritmus lefutása.

Ahhoz, hogy az FMS néhány percen belül visszafejtsen egy 40 bites WEP kulcsot, 500.000-nél több különböző IV-jű csomagot kell összegyűjteni. Azonban az FMS többszöri optimalizálásval, jelenleg 20.000 csomag is elegendőnek bizonyul.

2.5. WPA

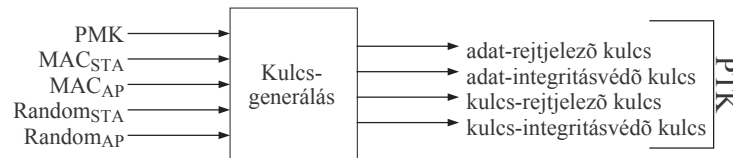
2.5.1. Ismertető

A WEP hibáit felismerve, az IEEE új biztonsági megoldást dolgozott ki, melyet a 802.11i specifikáció tartalmaz [5]. A WEP-től való megkülönböztetés érdekében, az új koncepciót RSN-nek (Robust Security Network) nevezték el. Az RSN-t körültekintőbben tervezték meg, mint a WEP-et. Új módszer került bevezetésre a hitelesítés és a hozzáférés-védelem biztosítására, mely a 802.1X szabvány által definiált modellre épül, az integritás-védelmet és a titkosítást pedig az AES (Advanced Encryption Standard) algoritmusra támaszkodva oldották meg.

Sajnos azonban az új RSN koncepcióra nem lehet egyik napról a másikra áttérni. Ennek az az oka, hogy a használatban levő WiFi eszközök az RC4 algoritmust támogató hardver elemekkel vannak felszerelve, és nem támogatják az RSN által előírt AES algoritmust. Ezen pusztán szoftver upgrade-del nem lehet segíteni, új hardverre van szükség, s ez kezdetben lassította az RSN elterjedésének folyamatát.

Ezt a problémát az IEEE is felismerte, és egy olyan opcionális protokollt is hozzáadott a 802.11i specifikációhoz, mely továbbra is az RC4 algoritmust használja, és így — szoftver upgrade után — futtatható a régi hardveren, de erősebb mint a WEP. Ezt a protokollt TKIP-nek (Temporal Key Integrity Protocol) nevezik.

A WiFi eszközöket gyártó cégek azonnal adaptálták a TKIP protokollt, hiszen annak segítségével a régi eszközökből álló WEP-es hálózatokat egy csapásra biztonságossá lehetett varázsolni. Meg sem várták amíg a 802.11i specifikáció a lassú szabványosítási folyamat során végleges állapotba kerül, azonnal kiadták a WPA (WiFi Protected Access) specifikációt [8], ami a TKIP-re épül. A WPA tehát egy gyártók által támogatott specifikáció, mely az



3. ábra. PTK generálása a PMK-ból, a felek MAC címéből, és a véletlenszámokból

RSN egy azonnal használható részhalmazát tartalmazza. Mára már elterjedtté vált a WPA2 kifejezés is, mely hivatalosan a komplett IEEE 802.11i-t magában foglalja, de hétköznapi használatban utalhat csupán a CCMP-re is, TKIP nélkül.

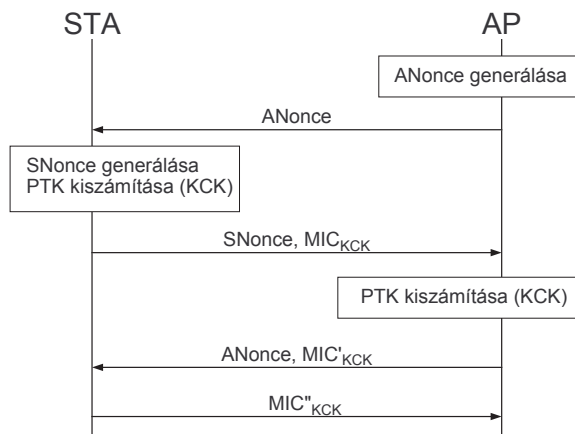
A továbbiakban áttekintjük a 802.11i-ben definiált helyi hozzáférés-védelmi, és kulcsmenedzsment módszereket, melyek tehát megegyeznek az RSN-ben és a WPA-ban. A TKIP és AES-CCMP protokollok működésének áttekintésétől most el tekintünk, mivel a mérés során nem játszik fontos szerepet. Az érdeklődő hallgatóknak ajánljuk ennek a leírásnak egy bővített kiadását [9].

2.5.2. Helyi hozzáférés-védelem és kulcsmenedzsment

A helyi hozzáférés-védelemet egy úgynevezett páronkénti mesterkulcs (pairwise master key, vagy röviden PMK) biztosítja. Az elnevezés jelen esetben némiképp félrevezető, mivel a „páronkénti” kifejezés arra utalna, hogy csak az adott STA és az AP ismeri. Ez csak abban az esetben igaz, amikor a hitelesítés egy központi szerveren keresztül történik, és a PMK valóban egyedi. Azonban, mikor a WEP-es jelszóhoz hasonlóan egyetlen jelszóval biztosítják a hozzáférés védelmét, a PMK egyszerűen a megosztott jelszó lesz. Viszont minden esetben igaz, hogy a PMK „mester” kulcs, mert ezt a kulcsot nem használják közvetlenül rejtjelezésre, hanem további kulcsokat generálnak belőle. Egészen pontosan a PMK-ból mind a mobil eszköz, mind pedig az AP négy további kulcsot generál: egy adat-rejtjelező kulcsot, egy adat-integritás-védő kulcsot, egy kulcs-rejtjelező kulcsot, és egy kulcs-integritás-védő kulcsot. Ezeket együttesen páronkénti ideiglenes kulcsnak (Pairwise Transient Key, vagy röviden PTK) nevezik. Megjegyezzük, hogy az AES-CCMP protokoll az adatok rejtjelezéséhez és az adatok integritás-védelméhez ugyanazt a kulcsot használja, ezért AES-CCMP használata esetén csak három kulcs generálódik a PMK-ból. A PTK előállításához a PMK-n kívül felhasználják még a két fél (mobil eszköz és AP) MAC címét, és két véletlenszámot, melyet a felek generálnak. Ezt a 3. ábra szemlélteti.

A véletlenszámokat az ún. *négyutas kézfogás* (four way handshake) protokollt használva juttatják el egymáshoz a felek. Ennek a protokollnak további fontos feladata az, hogy segítségével a felek közvetlenül meggyőződjenek arról, hogy a másik fél ismeri a PMK-t. A négyutas kézfogás protokoll üzeneteit az EAPOL protokoll Key típusú üzeneteiben juttatják el egymáshoz a felek. Az üzenetek tartalma és a protokoll működése vázlatosan a 4. ábra alapján a következő:

- Első lépésként az AP elküldi az általa generált véletlenszámot (*ANonce*) a mobil eszköznek. Mikor a mobil eszköz ezt megkapja, rendelkezésére áll minden információ a PTK előállításához. A mobil eszköz tehát kiszámolja az ideiglenes kulcsokat.
- A mobil eszköz is elküldi az általa generált véletlenszámot (*SNonce*) az AP-nek. Ez az üzenet kriptográfiai integritás-ellenőrző összeggel (Message Integrity Code, vagy röviden MIC) van ellátva, amit a mobil eszköz a frissen kiszámolt kulcs-integritás-védő kulcs (*KCK*) segítségével állít elő, ami a PTK része. Az üzenet vétele után az AP-nek is rendelkezésére áll minden információ a PTK kiszámításához. Kiszámolja az ideiglenes kulcsokat, majd a *KCK* segítségével ellenőrzi a MIC-et. Ha az ellenőrzés sikeres, akkor elhiszi, hogy a mobil eszköz ismeri a PMK-t.



4. ábra. Négy-utas kézfogás lépései

- Az AP is küld egy MIC-et (MIC') tartalmazó üzenetet a mobil eszköznek, melyben tájékoztatja a mobil eszközt arról, hogy a kulcsokat sikeresen telepítette, és készen áll a további adatforgalom rejtjelezésre. Az üzenet vétele után a mobil eszköz a KCK kulccsal ellenőrzi a MIC' -et, és ha az ellenőrzés sikeres, akkor elhiszi, hogy az AP ismeri a PMK-t.
- Vegül a mobil eszköz nyugtázza az AP előző üzenetét, mely egyben azt is jelenti, hogy a mobil eszköz is készen áll a további adatforgalom rejtjelezésére.

A továbbiakban a mobil eszköz és az AP az adat-integritás-védő és az adat-rejtjelező kulccsal védik egymásnak küldött üzeneteiket. Szükség van azonban még olyan kulcsokra is, melyek segítségével az AP többszórással küldhet üzeneteket biztonságosan minden mobil eszköz számára. Értelmszerűen, ezeket a kulcsokat az összes mobil eszköznek és az AP-nek is ismernie kell, ezért ezeket együttesen ideiglenes csoportkulcsnak (Group Transient Key, vagy röviden GTK) nevezik. A GTK egy rejtjelező és egy integritás-védő kulcsot tartalmaz. AES-CCMP esetén a két kulcs ugyanaz, ezért csak egy kulcsból áll a GTK. A GTK-t az AP generálja, és a négyutas kézfogás során létrehozott kulcs-rejtjelező kulcsok segítségével titkosítva juttatja el minden mobil eszközhöz külön-külön.

2.5.3. WPA gyengeségei

Ahogy már korábban említettük a WPA-t már jóval körültekintőbben tervezték meg, mint a WEP-et. Ennek megfelelően a bűnlajstroma jóval rövidebb. És azok közül is csak a gyakorlati szempontból legfontosabbakat emeljük ki. Ugyan formálisan már bebizonyították, hogy a négyutas kézfogás teljesíti a tőle elvártakat bizonyos feltételek figyelembevételével. Mégis itt található néhány apróság.

Először is nem volt és nem is lehetett elvárás helyi hozzáférés-védelem mellett valódi hitelesítés nélkül, hogy ismert jelszó (PMK) mellett ne lehessen dekódolni mások üzeneteit. Ahogy korábban már leírtuk, a PTK-t számoló függvény öt bemenetéből (emlékeztetőül PMK, AP és STA MAC címe, valamint AP és STA által generált véletlen számok) mindegyik ismert a négyutas kézfogás lehallgatása után, melyben a véletlen számok kerülnek kicserélésre. A PTK ismeretében az üzenetek dekódolhatóak.

Ráadásul a négyutas kézfogás lehallgatásával egy kulcs jelölt off-line módon ellenőrizhetővé válik, hiszen a kulcs-integritásvédő kulcsot felhasználják a MIC-ek kiszámításánál. Amennyiben egy kulcsjelölt azonos MIC-et eredményez, mint a lehallgatott üzenetben, a kulcsjelölt nagy valószínűséggel a PMK. Azonban így is csak a szótáras támadás a leghatékonyabb PMK visszafejtő eljárás jelenleg.

Hivatkozások

- [1] IEEE Std 802.11TM. part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications, 1999.
- [2] Jesse R. Walker, Submission Page Jesse Walker, and Intel Corporation. Unsafe at any key size; an analysis of the wep encapsulation. IEEE 802.11-00/362, 2000.
- [3] Nikita Borisov, Ian Goldberg, and David Wagner. Intercepting mobile communications: the insecurity of 802.11. In *MobiCom '01: Proceedings of the 7th annual international conference on Mobile computing and networking*, pages 180–189, New York, NY, USA, 2001. ACM.
- [4] W. Arbaugh, N. Shankar, J. Wan, and K. Zhang. Your 802.11 network has no clothes. *IEEE Wireless Communications Magazine*, 9(6):44–51, 2002.
- [5] IEEE Std 802.11iTM. Medium Access Control (MAC) security enhancements, amendment 6 to IEEE Standard for local and metropolitan area networks part 11: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications, July 2004.
- [6] Scott R. Fluhrer, Itsik Mantin, and Adi Shamir. Weaknesses in the key scheduling algorithm of rc4. In *SAC '01: Revised Papers from the 8th Annual International Workshop on Selected Areas in Cryptography*, pages 1–24, London, UK, 2001. Springer-Verlag.
- [7] Andrea Bittau, Mark Handley, and Joshua Lackey. The final nail in wep's coffin. *Security and Privacy, IEEE Symposium on*, 0:386–400, 2006.
- [8] Wi-Fi Alliance. Wi-Fi Protected Access.
- [9] Levente Buttyán and László Dóra. WiFi biztonság - A jó, a rossz, és a csúf. *Híradástechnika*, May 2006.

3. Mérés környezet

A méréshez minden mérőcsoport számára egy Linksys WRT54GL típusú AP-t helyeztünk üzembe. Ezen AP-k képesek az OpenWRT beágyazott Linux futtatására, mely lehetővé teszi az AP-k dinamikus konfigurálását annak megfelelően, hogy az egyes mérőcsoportok mely feladatokat végeztek el, és melyek vannak hátra. A különböző AP-k különböző csatornán működnek úgy, hogy a lehető legkevesebb zavarják egymás kommunikációját.

A feladatok elvégzéséhez szükséges forgalmat egy automatikusan működő kliens generálja. Ez szintén egy Linksys WRT54GL típusú eszköz OpenWRT operációs rendszerrel. Az automatikus kliens sorban minden mérőcsoport AP-jához csatlakozik, és lekér egy weblapot, melyet az AP helyben kiszolgál.

A mérőcsoportok a feladatokat egy Atheros AR2413 (802.11bg) típusú vezeték nélküli hálózati kártyával ellátott PC-n hajtják végre. A kártya olyan szempontból különleges, hogy szemben az átlagos kártyákkal, támogatja az egyedileg összeállított csomagok visszainjektálását. Erre egyébként az összes Madwifi driverrel vezérelhető vezeték nélküli kártya képes.

A mérőgépeken a mérés alatt Live CD fut (BackTrack), melyen megtalálhatóak az méréshez szükséges alkalmazások. Mivel kikapcsoláskor minden adat és módosítás elveszik, amit a CD-n található rendszeren hajtottunk végre, kerüljük a felesleges újraindításokat, illetve ügyelni kell arra, hogy minden szükséges adatot, amire szükség lehet a mérés során olyan tárolón helyezzünk el, ahol kikapcsolás után is megmarad.

A mérési jegyzőkönyvet a mérés során folyamatosan kell kitölteni, és mérés végén a mérésvezető számára véglegesen állapotában elérhetővé kell tenni. A kitöltést a Google Docs segítségével lehet elvégezni, ami egy online dokumentumtároló és szövegszerkesztő szolgáltatás. Ez több okból is szükséges és előnyös is. Egyrészt a BackTrack nem rendelkezik előre telepített szövegszerkesztővel. Másrészt a Live CD újraindítása esetén is (amit okozhat egy esetleges áramszünet) biztosan megmarad a kitöltött jegyzőkönyv. Harmadrészt pedig a mérésvezető számára is rögtön elérhetővé válik a elkészült mérési jegyzőkönyv.

A Google Docs bejelentkezést igényel, ahol mérés előtt már elérhető lesz a jegyzőkönyv egy kitöltetlen példánya. A <http://docs.google.com>-on a `wifimeres<mérőhely>@crysystech.hu` felhasználónévvel lehet belépni. A jelszavat a mérőcsoportok a mérés elején kapják meg, ami a mérés végeztével megváltozik.

AP beállítás. Mindegyik AP-n található egy úgynevezett SES (Secure Easy Setup) gomb. A SES gombbal állíthatjuk, hogy melyik feladathoz tartozó biztonsági beállítás legyen érvényes. Az aktuális állapotról a SES színe és egy WLAN feliratú LED ad visszajelzést. Bekapcsoláskor, alapértelmezésként a WLAN nincs bekapcsolva, és ilyenkor a WLAN felirat alatti LED nem világít, minden más esetben igen. A SES gomb visszajelzését az 1. táblázat írja le.

1. táblázat. SES gomb színének jelentése

SES	DMZ	Biztonsági beállítás
Nem világít	Nem világít	Wi-fi kikapcsolva
Fehér	Nem világít	MAC szűrés és rejtett SSID
Narancssárga	Nem világít	40 bites WEP
Fehér	Világít	WPA (TKIP vagy CCMP)
Narancssárga	Világít	RADIUS (Wifi 2. mérés)

A csatornák frekvencia-átlapolódása miatt akkor is elveszhetnek csomagok, ha az AP-k különböző csatornán működnek. Ez különösen a WEP-es rész mérésénél jelenthet problémát, mivel az ott elvégzett támadások hatalmas forgalommal járnak. Ezért a feladatokat nem csak „térben” (frekvencia), hanem időben is szét kell osztani. A mérésvezető a mérés elején kiosztja, hogy melyik csoport mikor dolgozhat a WEP-es mérésen.

Feladat Hitelesítő Kód. Minden egyes AP-n fut egy mini webkiszolgáló. Az automata kliens minden csatlakozás után innen kér le egy weblapot, amihez elküld egy felhasználónevet és egy jelszót. A válasz egy kód, ami az aktuális időtől, a felhasználó nevéből és az AP sorszámától függ, és SHA-1-et használó HMAC-ből számolja az AP. Ezt a kódot Feladat Hitelesítő Kódnak nevezzük.

A végső cél minden feladat esetén egy érvényes Feladat Hitelesítő Kód megszerzése. Ehhez a következő lépéseket biztosan el kell végezni: 1) meg kell szerezni a jelszót, amit az automata kliens küld minden kérés esetén (WEP és WPA esetén ez rejtjelezetten kerül átvitelre), 2) hozzá kell kapcsolódnia az AP-hoz (minden esetben valamilyen fokú védelmet kell kijátszani), 3) le kell kérni a Feladat Hitelesítő Kódot. Ez utóbbihoz más felhasználónevet kell használni, mint az automata kliensnek: ‘M’ és a mérőhely száma (pl. ‘M4’).

4. Feladatok

4.1. Csatlakozás rejtett SSID-jű AP-hoz

1. Fedje fel a rejtett SSID-t!
2. Csatlakozzon a rejtett SSID-jű AP-hoz! Ha nem nyílt az AP, a további lépésekhez kérjen jelszót a mérésvezetőtől.

3. Derítse ki, hogy ki csatlakozik a MAC szűréssel ellátott AP-hoz!
4. Csatlakozzon a rejtett SSID-jű AP-hoz a megfelelő MAC címet hamisítva!
5. Szerezze meg a mérés helyhez tartozó Feladat Hitelesítő Kódot!

4.2. WEP elleni támadás

1. Derítse ki a megtámadandó AP adatait! Előtte egyeztessen a mérésvezetővel, hogy rendelkezésre álljon szabad (mások által nem támadott), WEP-pel védett AP!
2. Indítson el egy lehallgatást kizárólag az adott AP-ra szűrve!
3. Végezzen el álhitelesítést az AP-nál!
4. Szerezzen meg egy IV-hez tartozó 1500 bájt hosszú kulcsfolyamot!
5. Készítsen egy visszainjektálható szabályos csomagot!
6. Kényszerítse az AP-t különböző IV csomagok küldésére!
7. A WEP kulcs kiszámítása
8. Csatlakozzon az WEP-pel védett AP-hez, és kérje le a Feladat Hitelesítő Kódot!

4.3. WPA gyengeségei

1. Fejtse vissza a WPA jelszót szótagos támadással! Ha az alapszótag használataival nem jár sikerrel, egészítse ki a szavakat egy számmal!
2. Fejtse vissza az idegen eszköz kommunikációját!
3. Csatlakozzon az WPA-val védett AP-hez, és kérje le a Feladat Hitelesítő Kódot!

5. További információk

Monitor mód. A mérés fontos részét képezi a lehallgatás. Ehhez sok esetben monitor módba kell állítani a kártyát. Azonban, amikor csatlakozni kell egy AP-hoz, akkor vissza kell állítani kliens vagy station módba. Ehhez a `wlanconfig` parancs ad segítséget madwifi driver esetén. Akár station, akár monitor módban használjuk a kártyán egy virtuális interfészt kell definiálni a következő paranccsal:

```
wlanconfig <virtuális interfész> create wlandev <valódi interfész>  
wlanmode sta|monitor
```

Virtuális interfészt a következő paranccsal lehet törölni:

```
wlanconfig <virtuális interfész> destroy
```

Néhány hibalehetőséget kizárhatunk azzal, ha a `air*-ng` parancsok futtatásához a monitor mód váltását rábizzuk egy beépített eljárásra az `airmon-ng start|stop <interfész>` parancs segítségével.

Lehallgatás. Lehallgatást számos program megvalósít, ezek közül a következő kettőt használatát javasoljuk: ‘airodump-ng’ és ‘Wireshark’. Mindkét alkalmazás kezelni tudja az elterjedt PCAP (packet capture) formátumot, mely lehetővé teszi az alkalmazások közötti átjárás. Az előbbi program különlegessége, hogy támogatja a rejtjelezett csomagok lementését is megfelelő paraméterezés mellett.

Az airodump parancs futtatása után megjelenít minden fontosabb információt az AP-król (SSID, BSSID, csatorna, biztonsági beállítások), valamint megjeleníti az AP-khoz csatlakozó STA-k MAC címét is. Az airodump-ng a következő minta szerint paraméterezhető (A mérést könnyítő kapcsolókat a 2. táblázatban soroljuk fel):

```
airodump-ng <kapcsolók listája> <interfész>
```

2. táblázat. airodump-ng kapcsolói

Kapcsoló	Leírás
--bssid <mac>	Kizárólag a <mac> MAC címhez kapcsolódó kommunikációt hallgatja le, egyébként minden elfogott csomagot kezel
--channel <ch>	Kizárólag a <ch> csatornán lévő kommunikációt hallgatja le, egyébként ciklikusan váltogat
-w <file>	<file> fájlba menti a lehallgatott kommunikációt

A Wireshark a grafikus interfésznek köszönhetően megkönnyíti a csomagok kezelését. A Wireshark képes dekódolni a rejtjelezett csomagokat, amennyiben megadjuk a hozzá tartozó jelszót. Ezt a ‘File → preferences → Protocols → IEEE 802.11’ menüben a ‘Enable decryption’ beikszelésével és az alatt található mező(k) kitöltésével tudjuk elérni.

Csatlakozás. Az access ponthoz való csatlakozást többféleképp is el lehet végezni: 1) user interfésszel rendelkezik a ‘Wireless assistant’, 2) kézzel kell konfigurálni a ‘wpa_supplicant’-ot.

MAC cím változtatás. MAC cím változtatására szükség van, amikor MAC címre is szűr az AP. Itt is több lehetőség közül választhatunk:

```
ifconfig <interfész> hw ether <új mac cím>
```

vagy

```
macchanger --mac=<új mac cím> <interfész>
```

Visszajátszás. A csomagok megfelelő visszajátszását az ‘aireplay-ng’ programcsomaggal lehet elvégezni. Különböző kapcsolókkal különböző támadásokat lehet végrehajtani. A 3. táblázatban összeszedtük a mérés során használatosakat. További kapcsolókhoz ajánljuk a manual-t.

Üzenet generálás. A ‘packetforge-ng’ egy a WEP törésére használható program, mely egy adott IV és kulcsfolyam ismeretében generál egy visszainjektálható broadcast üzenetet. A kimenet egy PCAP formátumú fájl, mely az azt kezelő alkalmazásokkal feldolgozható és felhasználható.

3. táblázat. Aircrack-ng főbb kapcsolói

Kapcsoló	Röv. kapcs.	Támadás
--deauth=<count>	-0 <count>	Deauthentikál egy megadott STA-t <count>-szor
--fake-auth=<delay>	-1 <delay>	Álhitelesítést végez el <delay> ms múlva a további kapcsolókkal megadott AP-hoz
--interactive	-2	Visszainjektálható broadcast üzenettel forgalmat generál a megadott AP-nál
--fragment	-5	Üzenet darabolási eljárással valamelyik IV-hez megszerez 1500 bájtnyi kulcsfolyamot
--test	-9	Leteszteli, hogy támogatja-e a kiválasztott kártya a csomagok visszainjektálását

WEP és WPA törés. A 'aircrack-ng' WEP és WPA jelszó visszafejtésre/feltörésre használható program. WEP esetén a korábban már említett FMS eljárást implementálja kiegészítve számos optimalizáló eljárással. WPA esetén egyszerűen szótáras támadást hajt végre elfogott négyutas kézfogás segítségével.

6. Minta beugró kérdések

- Hogyan fedne fel egy rejtett SSID-t?
- Mi az a beacon üzenet?
- Hogyan lehet csatlakozni egy AP-hoz, amelyekben csupán MAC szűréssel védekeznek?
- Két különböző WEP-pel védett csomagok esetén hogyan állapítható meg, hogy azonos kulcsfolyammal lett a két üzenet kódolva?
- Nevezzen meg három WEP gyengeséget!
- Írja le a WEP-es támadás főbb lépéseit!
- Mi szükséges más mobil állomás üzeneteinek visszafejtéséhez WPA esetén ismert jelszó mellett?
- Mi a négyutas kézfogás célja?
- Mi a leghatékonyabb ismert támadás WPA jelszó visszafejtésre?