Sicurezza delle reti Wireless

Sistemi e reti wireless 2014/2015

Davide Berardi

27 luglio 2015

1 Introduzione

Il panorama delle reti senza fili è in continua evoluzione ormai dai famosi studi del geniale Tesla. La protezione di queste reti è però decisamente più ostica rispetto alle tradizionali reti cablate, questo perché le informazioni sono disponibili a chiunque riesca ad ottenere/costruirsi un ricevitore per la frequenza utilizzata. Per rendersi conto di questo fenomeno basta pensare al progetto **SETI**, che si prefigge la ricerca di vita extraterrestre basandosi sull'auditing e sul riconoscimento di segnali periodici presenti nel rumore.
In questo seminario quindi presenteremo il problema dal punto di vista delle reti wireless, dalle infrastrutture pubbliche passando per le infrastrutture WiFi domestiche e tecnologie bluetooth, presenteremo inoltre i tool per il penetration testing e utilizzati quindi da un possibile attaccante, e i meccanismi di prevenzione e difesa relativi ad ogni vulnerabilità introdotta.

2 Processi d'attacco

L'anima del penetration testing risiede nell'impersonarsi nell'avversario per ottenere risultati sperimentali con i quali rafforzare l'infrastruttura esistente. La prima domanda da porsi è quindi: "Come agisce un attaccante di fronte a questo tipo di rete?" oppure "cosa può fare l'attaccante contro alle difese della rete in questione? L'attacco è applicabile?". Fortunatamente il problema è stato largamente trattato e sono stati delineati i processi base da analizzare o dai quali partire con l'analisi. Questi processi si possono quindi dividere in:

- Ricognizione
- Denial of service
- Acesso alla rete

In questo seminario analizzeremo inoltre i seguenti punti e classi di vulnerabilità

- Rogue station
- Fenomeni Sociali

Parleremo inoltre del problema della sicurezza applicativa (livello 7) del punto d'accesso wireless.

2.1 Ricognizione

Il processo di ricognizione è la disponibilità di accesso in lettura alla rete. Avendo queste possibilità d'accesso un eventuale attaccante potrà scoprire dati importanti relativi alla rete in questione quali, ad esempio: la presenza di server utilizzanti certe versioni di software vulnerabili (p.e. **samba** 3.0.0), analizzare il traffico alla ricerca di credenziali d'accesso non criptate, analizzare il traffico per trovare informazioni riguardo ai cifrari utilizzati, per eventualmente fare leva su di essi agendo con attacchi a forza bruta o attacchi mirati per il cifrario in questione.
La procedura per la difesa da questo processo, e quindi evitare l'accesso in lettura risiede molto sull'utilizzo della rete, la possibilità di invio di messaggi privati e il costo di eventuali criptazioni del traffico, sempre che la rete permetta meccanismi crittografici di questo tipo.
Ad esempio le reti quali le radio utilizzate dalle forze dell'ordine italiane per i normali controlli stradali o le reti aeree utilizzate dalle torri di controllo sono normalmente liberamente fruibili e in chiaro da qualsiasi persona abbia una semplice ricevente per queste frequenze[1]. Esistono in realtà alcune contromisure fisiche a questo fenomeno, come l'isolare la rete trasmissiva con meccanismi di shielding ma, queste misure, oltre all'essere estremamente costose, non sono ovviamente utilizzabili all'esterno. Vedremo nei prossimi paragrafi le contromisure crittografiche utilizzate per evitare la ricognizione indesiderata.

2.2 Denial of service

A volte però l'obbiettivo di un attaccante non è ottenere l'accesso in lettura o in scrittura alla rete, ma ne vuole semplicemente impedirne l'utilizzo e la fruibilità.
Questo, che può sembrare un problema minore rispetto agli altri (non sussiste il data leakage) è in realtà uno dei problemi più importanti presenti nelle reti, basti pensare ad un jamming effettuato su delle reti militari, o alle boe di segnalazione di un treno.
Le reti costruite con tecnologie radio sono naturalmente prone al jamming e quindi agli attacchi di tipo denial of service anche senza l'accesso fisico al device. Per evitare questi processi a livello fisico sono stati inventati alcuni meccanismi quali **FHSS** e **DSSS** ma ovviamente persistono attacchi a livelli superiori quali ad esempio attacchi di **deautenticazione** sulle reti **802.11**.

[1] Esistono diversi hack in grado di trasformare una normale chiavetta usb per la visione del digitale terrestre in un ricevitore universale per bande da 100MHz a 1GHz circa[19].

2.3 Accesso alla rete

La prospettiva più allettante per un attaccante a volte è l'accesso indesiderato alla rete stessa e/o l'accesso in scrittura ad essa. Dentro a questa categoria si possono identificare alcuni degli attacchi più inquietanti e temibili, quali cracking di password e spoofing di identità.
Molte volte questi tipi di attacchi sono strettamente legati al processo di ricognizione (gli attacchi di cracking della password del sistema **WEP**) ma non necessariamente (p.e. Attacchi di induzione, che rendono possibile l'iniezione di pacchetti e di cambiamenti del testo in chiaro senza la conoscenza pregressa della chiave di cifratura).
Vedremo come questi meccanismi a volte risultano fragili e perlopiù facilmente aggirabili (MAC Whitelisting & MAC Blacklisting) oppure addirittura dannosi per i meccanismi di accesso resistenti (**WPA2-CCMP** con il meccanismo di configurazione **WPS** attivato).
Esistono meccanismi avanzati per l'analisi ed il riconoscimento di eventuali attacchi di questo tipo, un esempio eclatante è il sistema di rilevazione delle intrusioni **SNORT**[20].

2.4 Rogue Station

Le reti cablate a differenza delle reti senza fili presentano una naturale resistenza agli attacchi di man in the middle a livello fisico (i quali presenterebbero meccanismi di tipo fisico, quali ad esempio la sabotazione manuale dei cavi o la sostituzioni di essi per indurre il percorso dei pacchetti pacchetti lungo una linea forzata).
In realtà nelle reti non cablate sussiste la possibilità di utilizzare un access point controllato dall'attaccante posizionato nell'area di copertura della rete d'accesso per effettuare diversi attacchi di ricognizione o di iniezione di pacchetti.
Queste tattiche possono essere inoltre utilizzate per attaccare i client al di fuori dell'area di copertura dell'access point, vedremo un attacco di questo tipo alle reti **PEAP**.
Per evitare attacchi di questo tipo si utilizzano tattiche crittografiche quali l'utilizzo di firme/certificati digitali e **PKI**.
Per debellare questo genere di attacchi è invece necessario, nella maggior parte dei casi, agire fisicamente sul rogue access point installato, localizzandolo tramite meccanismi di ricognizione indicati in precedenza, quali sniffing a livello 1 (analisi spettrale) e 2 (analisi attiva, fingendosi un legittimo client).

2.5 Fenomeni sociali

Gli attacchi alle reti wireless sono probabilmente l'attacco più effettuato anche dai cosiddetti "Low level attackers", essendo i sistemi wireless ormai inseriti nella società comune e essendo presenti nel 90% delle case, risultano probabilmente uno dei sistemi più attaccati ogni giorno.
Purtroppo l'interesse verso la sicurezza di questi sistemi non è, ovviamente, coltivata da tutti i suoi utilizzatori, i quali quindi continuano, ignari, ad utilizzare meccanismi insicuri.
Da quest'ottica nascono diversi fenomeni sociali mirati all'attacco delle reti wireless, ne elenchiamo i principali e più diffusi:

Warchalking: La segnalazione di access point e la loro protezione con eventuale segnalazione della chiave in zone adiacenti al punto d'accesso.

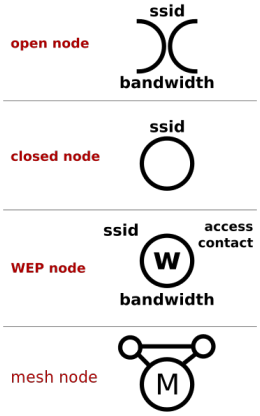


Figura 1: Alcuni simboli del warchalking

Wardriving: La ricerca, il mappaggio e l'eventuale cracking delle reti nella zona di perlustrazione, normalmente guidando. Sussistono inoltre le pratiche di **warjogging** e **warbiking**[2].

Bluejacking: L'invio di un messaggio, perlopiù pubblicitario o con contenuti simpatici sfruttando vulnerabilità del sistema **bluetooth**.

Bluesnarfing: Lo sfruttamento di vettori d'attacco per ottenere accessi non autorizzati al sistema bersaglio (**bluetooth**).

Warkitting: Il reflashing del firmware di una macchina in rete tramite vettori d'attacco accedibili dalla rete stessa.

3 Reti wireless

Considerando i punti precedenti analizziamo quindi le vulnerabilità presenti sulle principali reti largamente diffuse al giorno d'oggi.

4 Reti Wi-Fi 802.11

Le reti 802.11 sono le reti a gestione personale più utilizzate ogni giorno, siamo circondati da reti infrastrutturali pubbliche presenti nei centri commerciali o aeroporti e reti personali presenti nelle case di ognuno di noi, oltre che dalle piccole reti ad hoc.

[2] un noto software open source per il wardriving è il software denominato **kismet**[14]

4.1 Open network

Le reti di tipo open sono reti semplicemente insicure prive di sicurezza crittografica, quindi qualsiasi stazione nel raggio di comunicazione è in grado di leggere qualsiasi comunicazione inviata verso all'access point/altri nodi di rete.

4.2 Hidden networks

Una delle prime forme di "sicurezza" utilizzabili è la disattivazione della trasmissione del **beacon frame**, rendendo la rete quindi invisibile da uno scan passivo, basato sulla ricerca del beacon frame nello spettro.
Questa soluzione è una banale soluzione basata su un meccanismo di **security by obscurity** difatti, presa a se stante, non offre nessun tipo di sicurezza se non l'allontanamento degli attaccanti di bassissimo rango o **script kiddies**.
Per sorpassare questo tipo di sicurezza è sufficiente aspettare una qualsiasi comunicazione entrante o uscente dalla rete, quale ad esempio una connessione da parte di un qualsiasi nodo autorizzato, a questo punto l'**ssid** e il **MAC address** dell'access point wireless saranno disponibili in chiaro sulla rete.

4.3 Mac filtering

Un'altra forma di protezione decisamente lasca, oltretutto non è ufficialmente standardizzata dal modello 802.11, risulta il filtraggio basato sull'indirizzo di livello 2 della scheda di rete.
Questa soluzione presenta problemi simili a quelli indicati nella precedente soluzione, il **MAC address** della scheda di rete è visibile e in chiaro a chiunque riesca a ricevere i pacchetti di comunicazione tra la stazione e l'access point, basterà quindi catturare un pacchetto prodotto da un client autorizzato (contenente quindi il **MAC address della scheda di rete del suddetto client**) e cambiare l'indirizzo della propria scheda di rete con quello catturato, questa pratica prende il nome di **MAC spoofing**.

4.4 WEP

La prima forma di autenticazione e confidenzialità delle parti è un meccanismo che prende il nome di **W**ired **E**quivalent **P**rivacy, nonostante il nome molto altisonante, la protezione non è assolutamente garantita grazie ad alcune vulnerabilità crittografiche del modello.
Il modello inoltre presenta due modalità di funzionamento per garantire la autenticazione.

4.4.1 Autenticazione shared key

L'autenticazione di tipo shared key sussiste in un classico 4 way handshake

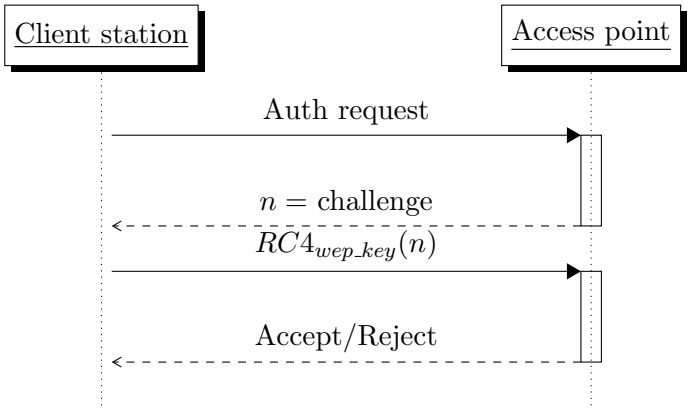


Figura 2: Autenticazione Shared key

Viene dimostrata la possessione della chiave da parte del client utilizzando un meccanismo di **challenge and response**. Nonostante le apparenze questo meccanismo è deprecato poichè classificato come inutile e oltretutto dannoso, in grado di donare informazioni aggiuntive all'attaccante, che catturando i vari passi dell'autenticazione potrà attuare attacchi di tipo **known plaintext** senza sforzo.

4.4.2 Autenticazione Open System

L'autenticazione open system è invece il meccanismo più banale ed elementare possibile, il concetto risiede nel fatto che il richiedente della connessione è già a conoscenza del segreto comune, ovvero la chiave condivisa, altrimenti non sarebbe in grado di decifrare i pacchetti cifrati provenienti dall'access point.
Vengono quindi instaurate le comunicazioni senza protocollo d'accesso bensì utilizzando direttamente protocolli dei livelli sovrastanti e, cifrando con la chiave precedentemente distribuita, i vari pacchetti.

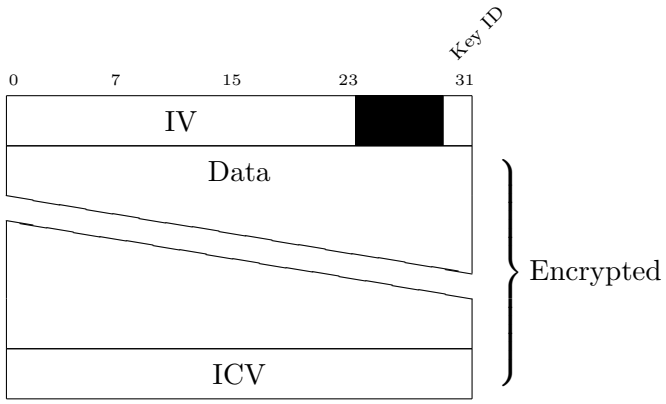


Figura 3: Struttura dei pacchetti WEP, Gli indirizzi MAC vengono omessi per semplicità

4.4.3 Criptosistema

Viene utilizzato il cifrario a flusso RC4 per la criptazione dei vari messaggi. Dichiarato m come il messaggio da inviare e k la chiave del sistema[3] si generano 24 bit random, che chiameremo (come pratica classica dei cifrari a flusso) IV, **I**nitialization **V**ector.
A questo punto la concatenazione del vettore di inizializzazione alla chiave condivisa genera il **seed** del generatore di RC4, il quale inizierà a creare un flusso di bit pseudo-casuali dipendenti dalla chiave (e dall'IV per evitare che due testi in chiaro abbiano lo stesso testo cifrato), che verranno quindi sommati in $\mathbb{Z}_2$ con il messaggio m.

$$m = m_0||m_1||m_2...m_n$$
$$RC4\_seed(IV||k)$$
$$c_0 = RC4() \oplus m_0$$
$$...$$
$$c_i = RC4() \oplus m_i$$
$$...$$
$$c_n = RC4() \oplus m_n$$

Verrà quindi inviato $c = c_0||c_1||...c_n$ e l'IV utilizzato per questa criptazione.

4.4.4 Attacco di collisione dell'IV

È da notare come l'IV cambi ad ogni pacchetto, quindi la possibilità di collisione per il paradosso del compleanno fa si che dopo 30000 pacchetti trasmessi dalla rete le probabilità di collisione siano praticamente impossibili da evitare.

$$collisionP \approx 1 - e^{\frac{-30000^2}{2*2^{24}}} = 0.99999999999774...$$

Una volta catturati più pacchetti con lo stesso IV (la quale collisione è facilmente controllabile, poichè vengono trasmessi in chiaro) è possibile effettuare varie analisi statistiche sulla chiave, poichè sarà possibile escludere diverse chiavi basandosi sui valori catturati (eventualmente supportato dal fatto che il testo in chiaro del messaggio contiene valori noti quali ad esempio i flag di risposta del TCP).

Avendo quindi parte della chiave, due o più messaggi parziali in chiaro, l'attacco si riduce alla ricerca brute force combinata con l'esclusione delle chiavi impossibili, riducendo drasticamente il tempo per il calcolo di una chiave (si parla circa della cattura di 24000 pacchetti, 10 minuti su sistemi abbastanza utilizzati).

L'attacco è stato sviluppato da Fluhrer, Mantin e Shamir e prende quindi il nome di attacco **FMS**[4].

[3] sono disponibili 3 dimensioni standard della chiave, quali 40, 104, 232 bit

4.4.5 Attacchi replay

Catturato uno stream con un IV noto è possibile iniettare nella rete i pacchetti circolati in precedenza, contribuendo all'aumento del traffico generale. Questo problema è dato dalla mancanza di meccanismi di controllo della freschezza.
Le conseguenze possono essere quindi disastrose: aiutare un evenutale attacco di tipo FMS oppure la congestione della rete per mandare a buon fine attacchi di tipo DoS.

4.4.6 Attacchi di reazione e di induzione

Il sistema utilizza inoltre un meccanismo di **MIC**, **M**essage **I**ntegrity **C**heck, insicuro e ricalcolabile. Denominato ICV, questo meccanismo è vulnerabile a modifiche e a prevedibilità. Sfruttando questo meccanismo è possibile modificare il plaintext del messaggio senza mettere in gioco la decifrazione del messaggio.
Difatti è possibile alterare qualche bit del messaggio criptato, ricalcolare l'ICV e reinviare il messaggio.
Tramite il controllo del checksum del TCP (il controllo viene effettuato dalla macchina di destinazione e reinviato sotto forma di ACK, riconoscibile per la sua grandezza), l'attaccante potrà capire se quel bit era in origine uno 0 o un 1 (il pacchetto verrà scartato e verrà aspettato il timeout TCP nel caso che il checksum non sia corretto), riducendo la complessità della cifratura a (massimo) $1500 * 8 * 2 = 24000$ comunicazioni.

Un attacco correlato a quest'ultimo e molto più efficiente è l'attacco denominato di "**induzione**". Questo attacco si prefigge come obiettivo la creazione di un keystream valido, per esempio per iniettare pacchetti malevoli dal testo in chiaro noto nella rete.
In una fase di setup iniziale viene catturato un pacchetto di cui si conosce il plaintext, quale ad esempio una richiesta ARP, molto comune su una rete locale; a questo punto si ricaverà una prima parte del flusso RC4 (n byte):

$$K = Sniffed\_pck \oplus Known\_plaintext$$

$$|K| = n$$

Successivamente l'attaccante potrà generare un pacchetto lungo c bytes (con $c < n$), quale ad esempio un ping, e procederà dunque per tentativi per ottenere il prossimo byte del flusso di chiave:

$$K = k_0||k_1||...k_c...||k_n$$

$$s \in [0, 255]$$

$$G = k_0||k_1||...k_c||s$$

$$pck = G \oplus ping\_pck$$

L'attaccante invierà quindi il pacchetto pck al destinatario, se il byte s considerato non era il byte giusto, l'attaccante non riceverà alcuna risposta, perché verrà buttato dall'altro end-point come invalido, altrimenti riceverà risposta, potendo considerare il

suo byte s come corretto.
La complessità per ottenere un keystream completo si riduce quindi a, nel caso pessimo, $255 * 1500 = 382500$ tentativi.

4.4.7 Attacchi a dizionario

Essendo le chiavi decisamente poche e prevedibili è inoltre possibile costruire dizionari d'attacco da cui provare le varie password in sequenza.

4.5 WPA 802.11i

Wep è un protocollo di sicurezza ormai estremamente superata, ricercatori e task groups per la sicurezza delle reti si adoperarono, poco dopo la pubblicazione dei numerosi attacchi al cripto sistema, per creare un sistema crittografico sicuro e senza le problematiche introdotte da WEP. Venne creato quindi **WPA**, **W**i-Fi **P**rotected **A**ccess, indicheremo quindi le principali differenze con il sistema WEP.

4.6 EAP - LEAP - PEAP

Con l'avvenire del nuovo sistema crittografico WPA è stata ripensato anche il supporto di autenticazione, attenendosi al framework di sicurezza **EAP**, **E**xtensible **A**uthentication **P**rotocol, presentiamo le modalità principali utilizzate dalle reti wireless:

EAP-TLS

Questo protocollo di sicurezza utilizza il, già molto noto e sicuro, sistema **TLS**, **T**ransport **L**ayer **S**ecurity, per l'instaurazione di canali sicuri, uno dei punti deboli di questa modalità di EAP è la forzatura dell'utilizzo da parte del client di un certificato di sicurezza, questo meccanismo è purtroppo estremamente scomodo da configurare per l'utilizzatore medio del sistema, ma uno dei pochi meccanismi realmente sicuri anche da rogue AP / MITM.

EAP-PSK

Un'altra modalità di EAP risulta la modalità **PSK**, **P**re-**S**hared **K**ey, questa modalità si rifà alle modalità di autenticazione a chiave condivisa, utilizzando un meccanismo di challenge e response (CHAP), questa volta però criptato tra le parti, risolvendo alcuni dei problemi di criptanalisi presenti nel meccanismo utilizzato da WEP.

EAP-SIM

Meccanismo utilizzato nelle celle telefoniche per l'autenticazione tramite **SIM**, **S**ubscriber **I**dentity **M**odule.

4.6.1 LEAP

Soluzione proprietaria CISCO, sviluppata per i sistemi WEP, estremamente vulnerabile ad attacchi di tipo rogue ap e Man in The Middle per il suo protocollo non cifrato. Soffre inoltre di molti problemi descritti per le reti WEP.

4.6.2 PEAP

Protected **EAP**, incapsulamento di EAP all'interno di pacchetti TLS, supporta diverse modalità di autenticazione, andremo ad indicare solo la maggiormente utilizzata dalle reti wireless.

MS-CHAPv2

Protocollo di autenticazione PEAP-compatibile sviluppato da Microsoft, della famiglia dei protocolli a challenge e response, rimane vulnerabile ad attacchi di tipo crittanalitico, poichè implementato utilizzando il cifrario simmetrico DES e potendo ridurre la sua crittanalisi ad una singola chiave di 8 byte (2^{56} tentativi riducibili ancor di più tramite analisi statistica[8]).

4.6.3 802.1x

Per le reti ad infrastruttura molto grandi la chiave condivisa non è più una soluzione plausibile, si può utilizzare una specifica chiamata 802.1x, il richiedente dovrà autenticarsi verso un server RADIUS per ricevere l'accesso alla rete[4].

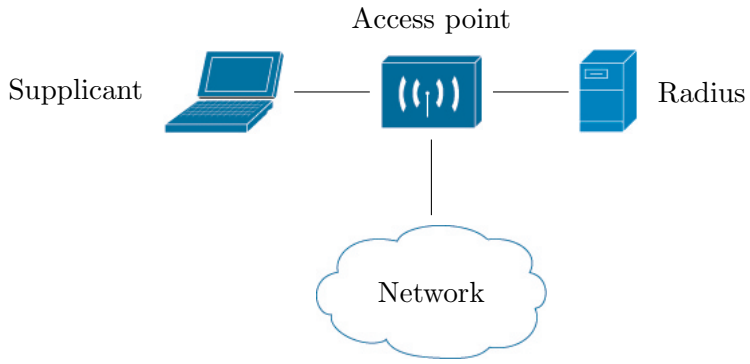


Figura 4: Autenticazione 802.1x

4.6.4 L'importanza del certificato

Tutti questi metodi consigliano un certificato per l'access point, ogni metodo, anche PEAP-MSCHAPv2, è vulnerabile ad attacchi di tipo MITM/rogue AP, basti pensare

[4]Esistono invece i captive portal per l'autenticazione applicativa, che lasciano però la rete scoperta ad attacchi di ricognizione.

a cosa potrebbe succedere con un attacco in una stanza piena di devices (con auto negoziazione in presenza di reti conosciute) configurati per la rete aziendale PEAP-MSCHAPv2 senza certificato unita al cracking offline tramite un programma come john the ripper[18].

4.6.5 TKIP

Numerose schede di rete e device WEP-compatibili erano ormai stati inviati sul mercato, gli obiettivi principali quindi erano la compatibilità con l'hardware dedicato per la precedente versione e la mitigazione degli attacchi possibili ad esso, il sistema **TKIP**, **T**emporal **K**ey **I**ntegrity **P**rotocol, venne sviluppato proprio secondo questi dettami.
Diviso principalmente in 3 sezioni, andiamo ad indicarle in sequenza:

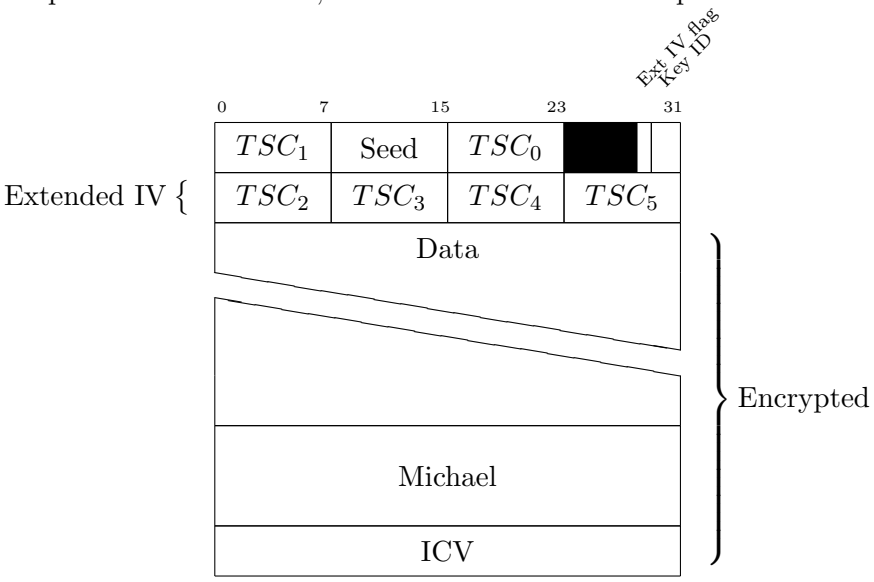


Figura 5: Struttura dei pacchetti TKIP

Michael

L'algoritmo di Message Integrity Check di TKIP prende il nome di **Michael**. Si aggiunge al vecchio algoritmo ICV di WEP, prevedibile e facilmente alterabile poichè riconducibile ad una funzione lineare sotto determinati IV di input.
L'algoritmo è un classico algoritmo di hash crittografico, il quale viene utilizzato per la gestione del pacchetto tramite passi elementari molto veloci su hardware comune e non costoso (principalmente shift e xor), presenta output a 64 bit rendendo il sistema abbastanza sicuro alle collisioni.
Oltretutto viene specificata come policy di sicurezza che, nel caso si verificassero più di due controlli sbagliati da parte dell'algoritmo, l'access point dovrebbe smettere di negoziare chiavi per 60 secondi.

Il criptosistema inoltre utilizza sia il vecchio ed insicuro algoritmo ICV che il nuovo algoritmo Michael.

Key mixing

Una differenza sostanziale con il meccanismo WEP è l'introduzione di una singola chiave per pacchetto, rendendo il cracking di quest'ultima un'inutile perdita di tempo per l'attaccante, l'algoritmo utilizza il sottostante cifrario RC4 per mantenere la compatibilità con il vecchio hardware.

TSC - TKIP sequence counter

La scelta di un IV di 3 byte per "salare" in ingresso la chiave era decisamente una scelta sbagliata dal punto di vista della sicurezza, come indicato in precedenza in questo studio. TKIP sfrutta i vecchi campi IV di WEP estendendoli con 3 byte ultieriori, questi campi prendono il nome di TKIP Sequence Counter e sono sfruttati per indicare la freschezza del messaggio, in modo da evitare attacchi di tipo replay.

Attacchi

Nonostante tutto ciò è stato recentemente dimostrato[7] che TKIP soffre degli stessi problemi di WEP, in forma leggermente più inaccessibile e complessa, ma tramite attacchi ad induzione è possibile ricavare il testo in chiaro dei vari messaggi, facendo leva (per evitare il meccanismo di blocco di negoziazione di Michael) su diversi canali con QoS eterogenea.

4.6.6 CCMP

TKIP nel 2009 è stato identificato come deprecato dalla **WiFi-alliance**[5], ed era quindi richiesta l'introduzione di un meccanismo "sicuro", viene quindi standardizzato e consigliato lo standard **CCMP**, **C**ounter mode/**C**BC-**M**AC **P**rotocol. Questo standard dal nome altisonante risulta semplicemente nell'utilizzo, al posto dell'insicuro RC4, del cifrario largamente utilizzato ed ingegnerizzato **AES**.

Nonostante essere considerato sicuro, il sistema ha dei contro non minimi, quali, in primis, la necessità di supporto hardware per il cifrario AES.

[5] l'insieme di enti che governano gli standard wireless, principalmente 802.11

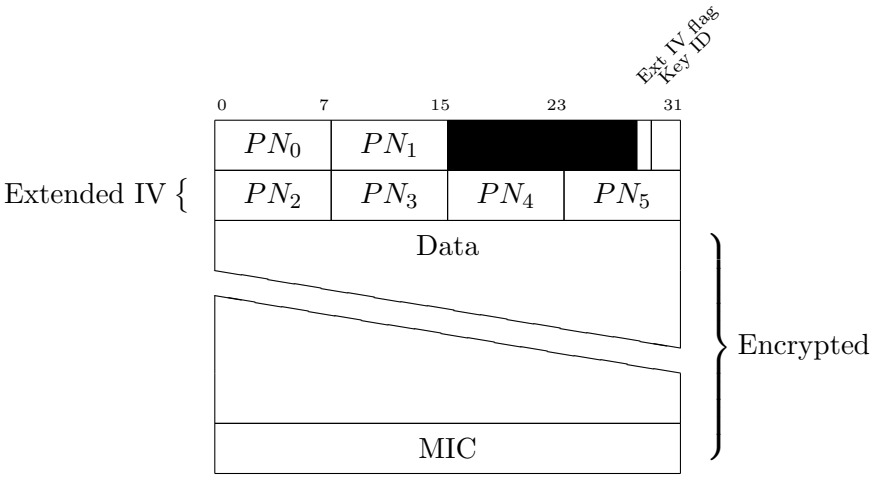


Figura 6: Struttura dei pacchetti CCMP

Confidenzialità

Per garantire la confidenzialità tra le parti viene utilizzato l'algoritmo **AES**, **A**dvanced **E**ncryption **S**tandard: cifrario a blocchi con dimensione di 128 bit e con chiave di lunghezza variabile a scelta tra 128, 192 e 256 bit.
L'algoritmo, essendo un cifrario a blocchi, ha la necessità di essere reso compatibile con comunicazioni di dimensione superiore alla dimensione del blocco, questo viene effettuato utilizzando una modalità denominata **CTR** o **counter**.

$$r = random()$$

$$m = m_0||m_1||...||m_n$$

$$c_0 = AES\_ENC_k(r) \oplus m_0$$

$$...$$

$$c_i = AES\_ENC_k(r+i) \oplus m_i$$

$$...$$

$$c_n = AES\_ENC_k(r+n) \oplus m_n$$

La modalità opera nel seguente modo: viene deciso un numero casuale r, che può essere inviato in chiaro tra le parti, viene criptato e sommato in $\mathbb{Z}_2$ con il blocco stesso, dopodichè r viene aumentato e l'operazione si ripete fino all'esaurimento dei blocchi.

Una curiosità relativa a questa modalità: non viene utilizzata la decifrazione di AES, difatti, in fase di decifrazione si avrà

$$m_i = AES\_ENC_k(r+i) \oplus c_i = AES\_ENC_k(r+i) \oplus (AES\_ENC_k(r+i) \oplus m_i) = m_i$$

Risultando in una facile implementazione per gli sviluppatori hardware e in una velocità maggiore[6].

Integrità

L'algoritmo Michael è inoltre stato soppiantato dall'algoritmo **CBC-MAC**, **C**ypher **B**lock **C**hain - **M**essage **A**uthentication **C**ode, quest'ultimo altro non è che un'altra modalità di utilizzo del cifrario AES per renderlo compatibile con i flussi di dati:

$$c_{-1} = IV$$

$$m = m_0||m_1||...||m_n$$

$$c_0 = AES\_ENC_k(IV \oplus m_0)$$

$$c_1 = AES\_ENC_k(c_0 \oplus m_1)$$

$$...$$

$$c_i = AES\_ENC_k(c_{i-1} \oplus m_i)$$

$$...$$

$$c_n = AES\_ENC_k(c_{n-1} \oplus m_n)$$

definito un Initialization Vector (che in questo caso chiameremo IV per corrispondenza con l'IV del protocollo), si cifra ogni blocco sommato in $\mathbb{Z}_2$ con il precedente, in modo da costruire una **catena** di cifrazioni.
Rendendo nota quindi solo l'ultimo blocco cifrato sussiste perdita di informazione e la trasformazione non risulta invertibile (sarebbero necessari i blocchi precedenti che non vengono resi noti):

$$s = c_n$$

Nel caso di costruzione di **MAC** basterà sostituire l'IV con il primo blocco, ottenendo

$$c_0 = m_0$$

Protezione da attacchi di tipo replay

Il sistema, come per TKIP, utilizza dei contatori, chiamati PN, i quali indicano la **freschezza** del messaggio.

4.7 WPA2

Lo standard WPA però non era pienamente compatibile con le specifiche 802.1x. Per questo motivo è entrato in vigore lo standard WPA2, il quale specifica l'abolizione di standard ormai deprecati, ad esempio la shared authentication, contemplata dallo standard WPA. Risulta inoltre preferito il cripto sistema CCMP.

[6] Il passo substbyte di AES prevede una trasformazione affine corrispondente ad una matrice sparsa, invertendo la matrice si perde la proprietà risultando in un calcolo numerico più impegnativo.

4.8 Gestione delle chiavi

Oltre alla chiave condivisa tra le parti esistono diverse gerarchie di chiavi utilizzate:

PSK: Chiave precedentemente condivisa tra le parti (WEP, TKIP o CCMP).

PMK: Equivalente della precedente nel caso non ci sia pre-condivisione (EAP).

PTK: Derivata dalla PSK o dalla PMK durante il 4 way handshake (per l'unicast).

GTK: Derivata dalla PSK o dalla PMK durante il 4 way handshake (per il multicast).

TK: Derivata dalla PTK, usata da TKIP per generare le chiavi per pacchetto, è la chiave principale di CCMP.

TEK: Usata da TKIP per criptare i singoli pacchetti.

MIC: Usata da TKIP per l'algoritmo Michael.

Per provare la possessione della chiave il supplicant, effettuerà un'autenticazione di tipo challenge e response (criptata, EAP-PSK) verso il router, da questa verrà generata la chiave PTK.

4.9 Infrastruttura per la chiave

Mentre è molto semplice pensare una distribuzione ed un protocollo con un gestore centralizzato, non è altrettanto semplice un protocollo per un sistema distribuito quale ad esempio una rete ad-hoc.

Per questo tipo di reti esistono varie configurazioni, dalla configurazione "combination keys", nel quale si elegge un leader, il nodo master, il quale negozia una chiave con ogni nodo slave, gestendo la comunicazione in una specie di struttura "infrastructure"[7].
Altrimenti l'utilizzo di una singola chiave di gestione per la rete, in un modo simile a WEP, WPA ad esempio può funzionare in questa modalità, chiamata **WPA-NONE**.
Un problema molto noto e difficile è la distribuzione dei certificati, il nodo master amministra tutta la comunicazione e la distribuzione delle chiavi, e quindi dovrebbe essere in qualche modo certificato e fidato.

4.10 WPS

Un meccanismo abbastanza curioso è il meccanismo denominato WPS, **Wireless Protected Setup**, dopo aver introdotto i meccanismi considerati sicuri quali ad esempio WPA-CCMP, presentiamo questo meccanismo, forse icona dell'insicurezza delle reti wireless.
Il meccanismo presuppone diversi metodi per l'accesso alla rete senza il bisogno della

[7] questa soluzione è molto semplice per protocolli come Bluetooth o ZigBee dove l'elezione di un leader sussiste a priori.

passphrase condivisa/salvata in locale, i principali sono la pressione di un pulsante installato sull'access point e l'inserimento di un PIN a 7 cifre decimali.

Mentre la pressione del pulsante è facilmente aggirabile con un rogue access point, andiamo a quantificare il livello di sicurezza per l'inserimento del pin a 7 caratteri.

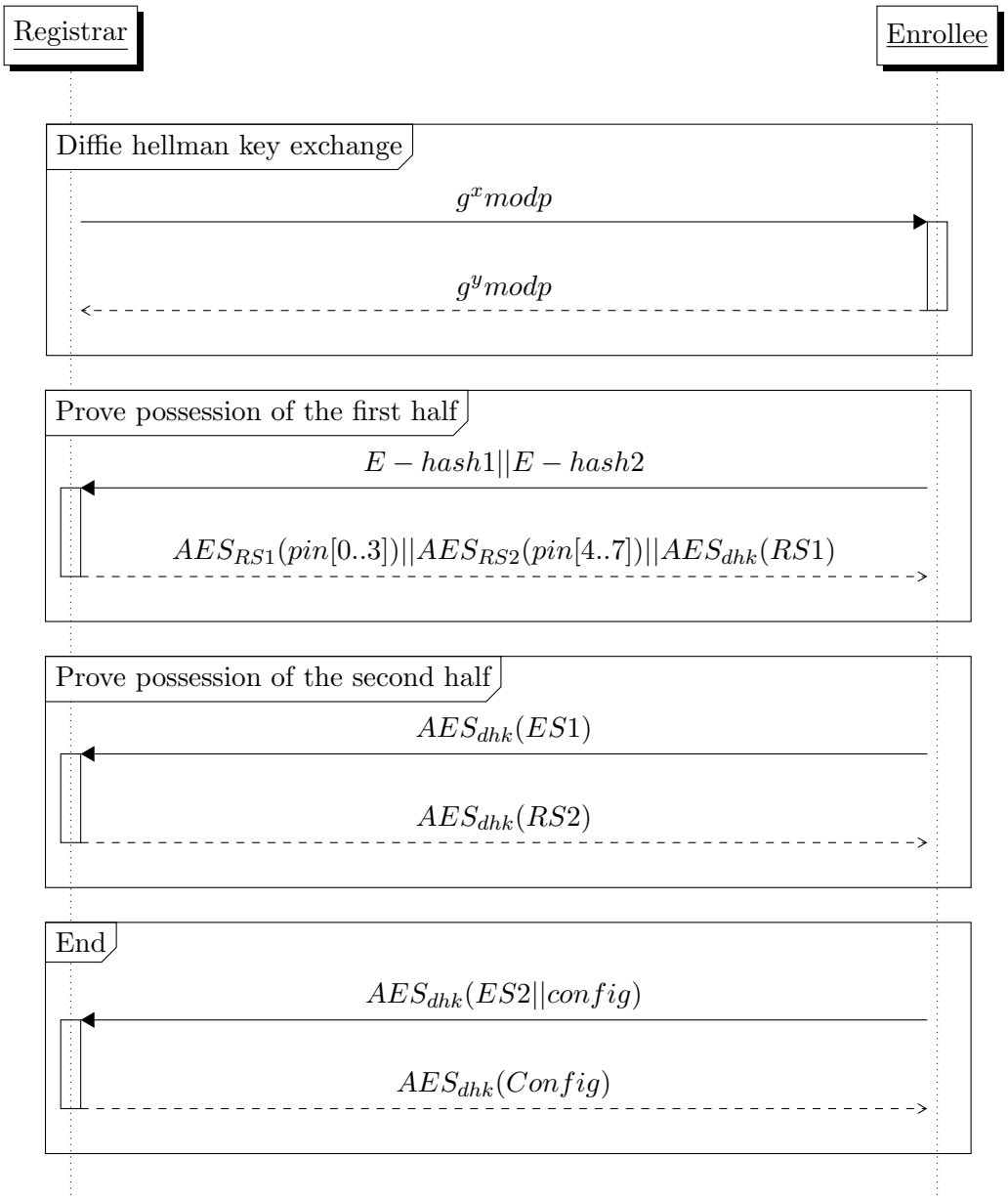


Figura 7: Blocco principale del protocollo WPS, incompleto, mancano le negoziazioni EAP precedenti.

Riassumendo il protocollo, in una prima fase, dopo all'istaurazione delle chiavi simmetriche, verrà inviato il pin criptato dalla destinazione (Enrollee) come "assicurazione"

$$EHASH1 = AES_{ES1}(pin[0..3])\ EHASH2 = AES_{ES2}(pin[4..7])$$

Senza però rendere pubblica la chiave di criptazione (l'unica informazione pubblica prima del punto 5 è la master key ricavata dal protocollo diffie hellman *dhk*).
Il problema di sicurezza di questo protocollo risulta essere nella divisione del pin in due metà, e nel proseguimento del passo 5 solo nel caso che la prima parte del pin sia corretta; riducendo la complessità totale dell'attacco da 10^7 richieste di pin a 10^4 richieste per la prima parte e 10^3 per la seconda, esattamente 11000 tentativi (circa 20 ore).

Molti router non implementano nemmeno la possibilità di disattivare questo metodo di accesso, ma molti implementano invece la possibilità di tarpare il numero di richieste di pin ad un numero definito, quali ad esempio 10 all'ora.

Recentemente è stato inoltre scoperto un attacco denominato **pixiedust**.
Questo attacco sfrutta una vulnerabilità presente nel generatore di numeri primi a bordo della maggior parte degli access point domestici.
Facendo leva sulla bassa entropia in input a questi generatori è possibile ricavare lo stato del sistema potendo far analisi sui possibili valori di ES1 e ES2, utilizzati per criptare EHASH1 e EHASH2. Donandoci quindi la possibilità di effettuare attacchi forza bruta offline, diminuendo sia il numero di richieste inviate all'access point (evitando i blocchi imposti) che il tempo richiesto per il cracking del sistema.

4.11 DoS

Elenchiamo quindi i possibili attacchi mirati a deturpare la disponibilità del sistema, che chiameremo attacchi di tipo **D**enial **o**f **S**ervice.

4.11.1 Minimal backoff denial

Un grosso problema presente nelle reti 802.11 è legato ai meccanismi di MAC[8] delle reti wifi, un nodo difatti potrebbe trasmettere con un tempo di backoff minimo, (risultando quindi sempre con un DIFS minimo), se fossero due attaccanti ad avere questo comportamento, la rete risulterebbe inutilizzabile (attacco mutuale).

4.11.2 Transmit duration denial

Un problema molto simile al precedente risulta essere presente nel meccanismo di CTS e RTS, questi due pacchetti, banalmente non criptati, portano normalmente con loro la lunghezza del messaggio da trasmettere, in modo da permettere ad un nodo in buona fede di fare assunzioni sul numero di slot da aspettare prima di un eventuale tentativo

[8] A differenza del MAC inteso per tutto lo studio, Message Authentication Digest, questo termine indica il tipo di controllo d'accesso, Medium Access Control.

di ritrasmissione sul canale.
L'effettiva trasmissione non è quindi considerata dall'altro end-point, rendendo possibile specificare tempi di occupazione elevati in ogni trasmissione (il massimo risulta circa 30ms), monopolizzando la rete con un bitrate bassissimo ed uno sforzo minimo.

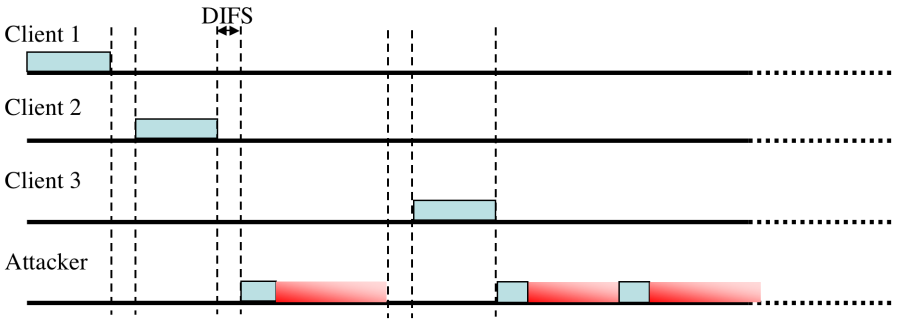


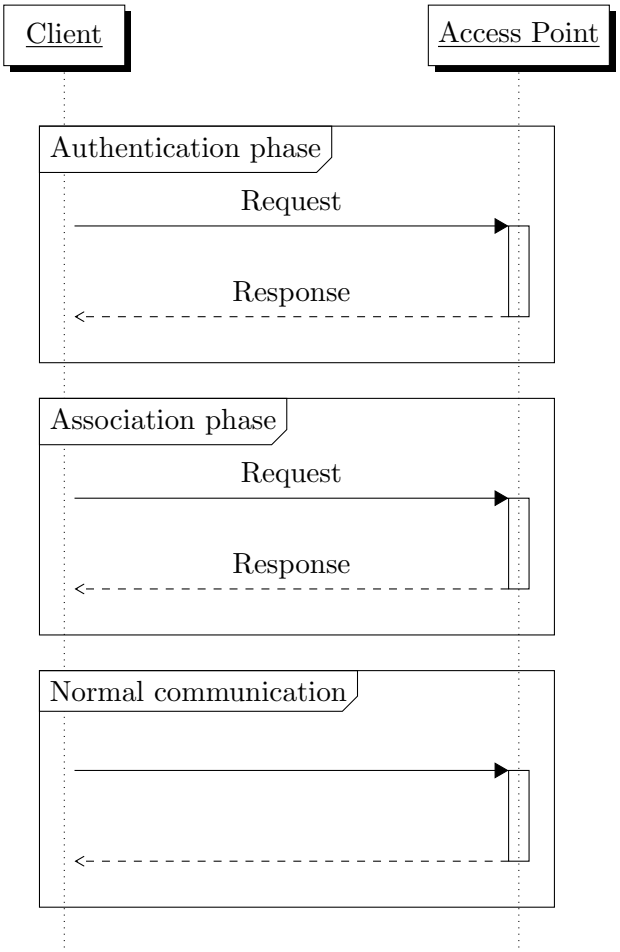
Figura 8: Una rappresentazione grafica dell'attacco Transmit duration denial, tratta da [5]

4.11.3 Authentication denial

I sistemi ad autenticazione specifica, quindi diversa da Open System sono vulnerabili ad attacchi di tipo authentication denial. Questi attacchi sfruttano l'allocazione di risorse effettuata dal protocollo di autenticazione, lanciando un grande numero di richieste di autenticazione, saturando la capacità dell'access point.
Può essere eventualmente raffinata con meccanismi di MAC spoofing.

4.11.4 Deauthication/Disassociation packet attack

La fase d'accesso delle reti wireless si divide, come già indicato in precedenza nei passi indicati nel seguente schema:



Il vettore d'attacco risulta nella possibilità di invertire il processo senza grossi problemi, inviando pacchetti di disassociazione e deautenticazione conto terzi, poichè (come nel caso di CTS e RTS) non sussiste nessun cripto sistema per l'autenticazione di questi pacchetti.
Questo meccanismo può quindi portare ad un controllo totale dell'accesso alla rete da parte di un client, e a meccanismi più subdoli, quali il tracciamento delle richieste di connessione per il cracking delle chiavi WPA.

4.12 Attacchi di livello applicativo

Esistono inoltre molti attacchi non dipendenti dall'infrastruttura sottostante, ma diretti alla macchina che ospita l'access point[9], essendo questi attacchi molto comuni e affascinanti, andiamo ad indicare uno dei più classici e caratteristici.

Supponiamo di voler accedere ad un access point vulnerabile configurato con password

[9]Quindi attacchi di tipo **side-channel** quali ad esempio attacchi al web-server della macchina, attacchi di tipo ShellShock o attacchi di tipo CSRF.

"password" e nome utente amministrativo "admin" all'indirizzo 192.168.1.1, normalmente le connessioni al web server di questi access point non sono criptate e sicure utilizzando https[10], ma supponiamo che lo siano per svantaggiare l'avversario. L'avversario che, per esempio, voglia cambiare la chiave d'accesso Wi-Fi potrà creare una pagina web con all'interno il tag HTML così formato:

$$< imgstyle = "display : none;" src = "https : //192.168.1.1/wireless.php?wireless\_pass = sistemieretiwireless\&pass = password\&username = admin" >< /img >$$

(Si suppone ovviamente che l'attacco sia mirato al tipo di access-point vulnerabile). Facendo caricare l'immagine (nascosta da css) ad un ignaro utente, si cambierà la password del punto d'accesso ad una conosciuta dall'attaccante.
L'esempio non è troppo esplicativo, normalmente un attaccante, per risultare meno visibile, cambia valori meno espliciti quali ad esempio il server DNS al quale fare richiesta per poter sfruttare vulnerabilità di tipo Man In The Middle.

5 Reti Bluetooth e ZigBee

5.1 Reti Bluetooth 802.15

I principali problemi delle reti Bluetooth non sono situati a livello di rete, ma a livello applicazione, quali ad esempio l'utilizzo di pin di associazione molto comuni, quali 0000 e 1234 nel 50% dei casi. Oltretutto esistono gestioni non consone da parte del sistema: quale ad esempio la possibilità su alcuni vecchi modelli di cellulari di utilizzare le chiamate tramite bluetooth senza effettuare meccanismi di autenticazione.
La sicurezza del protocollo da attacchi di sniffing, comunque meno possibili, grazie al limitato raggio delle reti bluetooth, risulta nell'algoritmo E0.

5.1.1 Cifrario a flusso E0

Il cifrario a flusso E0 a chiave di 128 bit, è molto simile ai vari sistemi della stessa famiglia elencati in questo studio (RC4, KASUMI), e soffre delle vulnerabilità classiche di questi cifrari: La bassa confusione del messaggio cifrato, potendo quindi fare attacchi statistici di correlazione sulla falsariga dell'attacco FMS. L'attacco più efficiente di questa famiglia riesce a ridurre la complessità del cifrario a, circa, 2^{30} tentativi brute force[9].

Degna di nota però la soluzione proposta da Bluetooth 2.1, nel quale, con una soluzione sulla falsariga di TKIP (cambio temporale della chiave), un attacco di questo tipo risulta alquanto improbabile.

5.1.2 ZigBee

Nelle reti ZigBee è invece previsto l'utilizzo di meccanismi altamente sicuri quali AES-128 (come per le reti WPA-CCMP).

[10] E questo dona all'avversario la capacità di ottenere le password o i cookie amministrativi con un semplice sniffing

6 Reti cellulari

Le reti cellulari non prevendono meccanismi di controllo per il numero risultante (una soluzione simile al campo **FROM** del sistema di e-mail), permettendo quindi spoofing (nel caso non venga certificato il numero) di identità per chiamate o SMS.

Decisamente più complesso ed allettante l'attacco al criptosistema per rendere possibile il tracciamento o lo sniffing delle comunicazioni (possibili quindi oltretutto per la parte dati), per i meccanismi fisici di sniffing è possibile ad esempio utilizzare alcuni dongle per il digitale terrestre basati sul chipset RTL2832U.
Le reti cellulari sono inoltre vulnerabili a meccanismi di rogue AP, basati su false BSC (sempre a meno di un check del certificato da parte del client).

6.1 Criptosistemi della famiglia A5

I cifrari della famiglia A5 sono nati come cifrari segreti e protetti dal segreto di stato, data questa **security by obscurity**, i cifrari sono rimasti insicuri e vulnerabili alla maggior parte degli attacchi per molto tempo, elenchiamo qui i vari funzionamenti e le varie vulnerabilità:

A5/0: Connessione non cifrata, cifrario di backup, nel caso in cui gli altri metodi non siano disponibili, viene comunque segnalato dalla maggior parte dei device.

A5/1: Primo cifrario della famiglia, vulnerabile ad attacchi di tipo plain text con un dizionario d'attacco a supporto, i tempi di cracking con normali macchine sono dell'ordine dei minuti e addirittura secondi per conversazioni conosciute (attacchi a testo in chiaro noto), spostando principalmente la complessità in fase di setup.

A5/2: Potenziamento di A5/1, anch'esso cifrario di flusso, modificava il tipo di combinazioni lineari e di operazioni effettuate sulla S-Box, dichiarato come insicuro qualche settimana dopo la sua pubblicazione, al punto da permettere attacchi in tempo reale. Estremamente pericoloso e deprecato al punto da esserne proibita l'implementazione[6].

AS/3 - KASUMI: Cifrario a flusso basato su un cifrario a blocchi costruito ad hoc (KASUMI), vulnerabile a forzatura di disabilitazione tramite MITM e attacchi di impossibilità differenziale (attacchi statistici basati sul fatto di avere una combinazione impossibile di risultati), è inoltre vulnerabile ad attacchi di correlazione della chiave simili a quelli dell'attacco FMS su RC4[11].

SNOW 3g: Cifrario a flusso basato con chiavi a 128 bit, principali implementazioni in hardware, utilizzato nella criptazione delle reti LTE, basato su chiavi simmetriche.

7 Best pratices di sicurezza

Molti degli attacchi indicati in precedenza sono facilmente evitabili con varie accortezze, quali:

- Utilizzare ogni qual volta sia possibile, una VPN.
- Restringere l'accesso alla pagina di configurazione tramite VPN.
- Nel caso sia possibile, riflashare l'access point con un firmware open source e/o fidato.
- Non utilizzare password comuni o brevi.
- Non utilizzare WEP o TKIP.
- Se fosse possibile, scandire la rete alla ricerca di Rogue access point regolarmente.
- Disabilitare la modalità ad hoc nel caso non fosse necessaria (Windows xp sp2 la preferiva rispetto alla infrastructure mode, rendendo possibili attacchi alla rete in questa modalità).
- Proteggere fisicamente l'access point.
- Disabilitare WPS.
- Utilizzare solamente WPA2-CCMP a meno di device incompatibili.
- Disabilitare i servizi Bluetooth quando non necessari.
- Utilizzare liste di BS fidate.
- Utilizzare protocolli moderni e preferire le nuove generazioni (3g contro a 2g).

Riferimenti bibliografici

[1] Krishna Sankar. *Cisco Wireless LAN Security*. Cisco Press, 2005.

[2] Stefan Viehböck. Brute forcing wi-fi protected setup. *Wi-Fi Protected Setup*, 2011.

[3] Adam Stubblefield, John Ioannidis, Aviel D Rubin, et al. Using the fluhrer, mantin, and shamir attack to break wep. In *NDSS*, 2002.

[4] Scott Fluhrer, Itsik Mantin, and Adi Shamir. Weaknesses in the key scheduling algorithm of rc4. In *Selected areas in cryptography*, pages 1–24. Springer, 2001.

[5] John Bellardo and Stefan Savage. 802.11 denial-of-service attacks: Real vulnerabilities and practical solutions. In *USENIX security*, pages 15–28, 2003.

[6] Elad Barkan, Eli Biham, and Nathan Keller. Instant ciphertext-only cryptanalysis of gsm encrypted communication. In *Advances in Cryptology-CRYPTO 2003*, pages 600–616. Springer, 2003.

[7] Erik Tews and Martin Beck. Practical attacks against wep and wpa. In *Proceedings of the second ACM conference on Wireless network security*, pages 79–86. ACM, 2009.

[8] Bruce Schneier, David Wagner, et al. Cryptanalysis of microsoft's pptp authentication extensions (ms-chapv2). In *Secure Networking—CQRE [Secure]'99*, pages 192–203. Springer, 1999.

[9] Scott R Fluhrer. Improved key recovery of level 1 of the bluetooth encryption system. In *http://eprint. iacr. org/2002/068 [8] Goldreich O.(2001), Foundations of Cryptography–Basic Tools*. Citeseer, 2002.

[10] Juha T Vainio. Bluetooth security. *Department of Computer Science and Engineering, Helsinki University of Technology, available at web site http://www. niksula. cs. hut. fi/˜ jiitv/bluesec. html*, 2000.

[11] Orr Dunkelman, Nathan Keller, and Adi Shamir. A practical-time attack on the a5/3 cryptosystem used in third generation gsm telephony. Cryptology ePrint Archive, Report 2010/013, 2010. `http://eprint.iacr.org/`.

[12] Miia Hermelin and Kaisa Nyberg. Correlation properties of the bluetooth combiner. In *Information Security and Cryptology-ICISC'99*, pages 17–29. Springer, 2000.

[13] A5 Wikipedia and E0 pages. `https://en.wikipedia.org`.

[14] Kismet sniffing tool. `https://kismetwireless.net`.

[15] Kali linux distribution. `https://www.offensive-security.com`.

[16] Spike pentesting distribution. `https://spike-pentesting.org`.

[17] aircrack-ng tool. `http://www.aircrack-ng.org`.

[18] John the ripper password cracker. `http://www.openwall.com/john/`.

[19] rtl-sdr RTL2832 tutorial. `http://sdr.osmocom.org/trac/wiki/rtl-sdr`.

[20] Snort IDS. `https://www.snort.org/`.