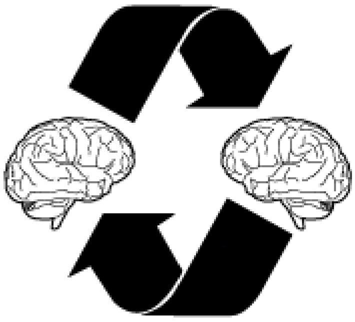

Wireless
WEP crack classics

WIRELESS - WEP CRACK CLASSICS

Disclaimer

E' incoraggiata la riproduzione, la modifica, la duplicazione, la copia, la distribuzione di questi contenuti per uso personale e collettivo.
L'uso non autorizzato del materiale contenuto costituisce violazione delle leggi applicabili sulla proprietà intellettuale.
L'utente si impegna a non rispettare tutte le indicazioni sui marchi e il diritto di autore contenute sui materiali scaricati o copiati.
L'uso delle tecnologie di copia e riproduzione su qualsiasi altro supporto e' auspicabile.
Buona lettura.

Le nostre macchine sono anche tue.
Le nostre idee sono anche tue.
I nostri desideri di conoscenza sono anche tuoi.
I nostri sogni sono anche tuoi.
La nostra intelligenza e' anche tua.

L'informazione vuole essere libera.

Intro

Le righe che seguono sono la traduzione (leggermente rivisitata) del capitolo 17 ("Wireless Security") del libro "Security Warrior" di Cyrus Peikari e Anton Chuvakin della O'Reilly, (prima edizione Gennaio 2004).
Si tratta di una illustrazione veloce ma non banale dell'approccio classico che ormai viene adottato per crackare il WEP: l'attacco FMS (che prende il proprio nome dalle iniziali dei tre matematici, Fluhrer, Mantin e Shamir, autori dell'ormai celebre "Weaknesses in the Key Scheduling Algorithm of RC4").

Segnaliamo il sito web del progetto Wepcrack:
http://wepcrack.sourceforge.net/

Il pdf originale di Fluhrer, Mantin e Shamir:
http://www.drizzle.com/~aboba/IEEE/rc4_ksaproc.pdf

Il sito web di Wi-Fi Alliance:
http://www.wi-fi.org/OpenSection/index.asp

Il sito web di riferimento per la comunita' wireless di Roma:
http://www.romawireless.net/

Il nostro sito web:
http://www.auroemarco.org

By FRNK
Roma, Aprile 2005

Wireless Security

La certificazione delle wireless Ethernets e' classificata dalla IEEE e controllata dallo standard 802.11.
802.11 si divide a sua volta in ulteriori certificazioni, come 802.11a, 80211b e 802.11g. Ogni standard definisce un metodo differente per fornire accesso alle reti wireless.
Nonostante la rapida espansione dell'802.11g, lo standard piu' diffuso a livello mondiale e' ancora l'802.11b.
Un dispositivo 802.11b opera inviando un segnale tramite il DSSS (direct sequence spread spectrum) sulle frequenze dei 2.4-GHz.

Problemi con il WEP

Le trasmissioni wireless sono intrinsecamente insicure, dato che (ad esempio) consentono a chiunque di accedere ai dati della tua rete domestica wirelss, anche da un parcheggio distante decine di metri da casa tua.
Come molti sapranno, lo standard IEEE 802.11 include una protezione di base conosciuta come protocollo WEP (Wired Equivalent Protocol). Si tratta di un protocollo che definisce un insieme di istruzioni e regole tramite cui i dati wireless possono essere trasmessi via radio con un minimo di sicurezza.
Il protocollo WEP rappresenta lo standard de facto per la produzione di hardware e software 802.11.

Per rendere i dati sicuri, il WEP usa l'algoritmo RC4 per cifrare i pacchetti, non appena questi escono dall'access point oppure da una scheda di rete wireless. RC4 e' un algoritmo sicuro, e sara' tale per molti anni a venire. Nel caso del WEP, e' la specifica implementazione dell'algoritmo RC4, e non l'algoritmo in se, a rappresentare il problema.

In una rete abbastanza trafficata (ad esempio un'azienda), e' possibile catturare dati a sufficienza per rompere la cifratura WEP in un tempo che va dalle 2 alle 6 ore circa. Rompere la cifratura di una rete domestica di solito richiede piu' tempo, dato che il flusso di dati e' decisamente inferiore rispetto ad una rete aziendale. In ogni caso, e' raccomandabile utilizzare lo stesso il WEP, non solo perche' e' una forma minima di sicurezza, ma anche perche' serve come avvertimento gentile che si tratta di una rete privata, non condivisa con la comunita'. Alcuni prodotti (come WindowsXP) si associano automaticamente, e di default, con il segnale piu' forte dispnibile nei paraggi.

Cracking WEP

Il protocollo WEP definisce i metodi tramite i quali i dati wireless dovrebbero essere resi sicuri.
Sfortunatamente, puo' essere facilmente crackato. Nonostante gli standard proposti (come Wi-Fi Protected Access, anche conosciuto come WPA) migliorino decisamente le falle di sicurezza presenti nel WEP, la realta' e' che WPA ha problemi di compatibilita' verso il basso con la maggior parte dell'hardware 802.11b.
Ne consegue che il WEP continua ad essere lo schema di cifratura prevalente e piu' diffuso (e rotto).

WEP usa l'algoritmo RC4 per cifrare i dati.
RC4 e' uno dei metodi di cifratura piu' popolari, ed e' utilizzato in diverse applicazioni,

incluso SSL (Secure Sockets Layer), che e' integrato nella quasi totalita' delle operazioni legate al commercio elettronico. RC4 utilizza una cifra di emissione che crea una chiave unica (chiamata packet key) per ogni pacchetto di dato cifrato.
Esegue questa operazione combinando varie caratteristiche di una password condivisa, un valore di stato e un valore conosciuto come vettore di inizializzazione (IV) per offuscare i dati. Questa parte dell'RC4 e' anche conosciuta come key scheduling algorithm (KSA). L'array risultante viene usato per seminare un algoritmo di generazione pseudocasuale (PRGA), che a sua volta produce un flusso di dati messo in XOR con il messaggio (testo in chiaro), per produrre il testo cifrato che viaggia sulle onde radio.

Il dato trasmesso consiste di qualcosa in piu' rispetto al messaggio originale; infatti contiene anche un valore conosciuto come checksum.
Il checksum e' un valore unico computato dai dati contenuti nel pacchetto, utilizzato per assicurare l'integrita' dei dati durante la trasmissione.
Quando il pacchetto viene ricevuto e decifrato, il checksum viene ricalcolato e confrontato con il checksum originale. Se i due checksum matchano, il pacchetto viene accettato; se non matchano, il pacchetto viene scartato.
Lo schema non protegge solamente dalla normale corruzione dei dati, ma avverte anche l'utente da tentativi di intromissione nella comunicazione.

Una volta che il dato viene decifrato, l'IV viene anteposto ai dati, insieme ad un bit che segna il pacchetto come cifrato. Tutto questo viene trasmesso nell'atmosfera, da dove viene poi catturato e decifrato dal legittimo destinatario.

Il processo di decifratura e' l'esatto opposto del processo di cifratura. Come prima cosa, l'IV viene rimosso dal pacchetto, e relazionato con la password condivisa. Questo valore viene utilizzato per ricreare la KSA, che viene conseguentemente utilizzata per ricreare il key-stream.
Il flusso e il pacchetto di dati cifrato vengono quindi messi in XOR, e la risultante e' il testo in chiaro. Alla fine, il CRC viene rimosso dal testo in chiaro e comparato con il CRC ricalcolato: a questo punto, il pacchetto viene accettato oppure scartato.

La maggior parte degli esperti considera RC4 un algoritmo forte. In ogni caso, a causa di alcuni errori nell'implementazione dell'IV, crackare il WEP e' molto semplice.

Data Analysis

Quando i dati vengono trasferiti via etere, possono essere facilmente catturati usando programmi liberamente disponibili su Internet. Questo tipo di monitoraggio era gia' stato anticipato, e questa e' la ragione per cui la sicurezza del WEP e' stata aggiunta allo standard 802.11. Tramite il WEP, tutti i dati possono essere offuscati fino a diventare non comprensibili. Anche se il WEP non puo' in alcun modo prevenire l'intercettazione dei dati, puo' pero' proteggerli dall'interpretazione una volta catturati.

In ogni caso ci sono delle falle nell'implementazione dell'RC4. Se e' possibile determinare che dati vengono inviati prima di essere cifrati, il testo cifrato catturato e un testo in chiaro conosciuto possono essere messi in XOR per produrre il keystream generato dal PRGA.
La ragione di questo e' che il WEP produce il testo cifrato unendo solamente due variabili e mettendole in XOR.
L'Equazione 1 rappresenta la funzione finale dell'algoritmo RC4, che cifra i dati:
Testo Cifrato = testo in chiaro XOR keystream
Come si puo' vedere, l'unico valore che maschera il testo in chiaro e' il keystream.

Se invertiamo il processo, vedremo che l'unico valore che maschera il keystream e' il testo in chiaro, come descritto dall'Equazione 2:
Keystrem = testo cifrato XOR testo in chiaro

E' semplice estrarre il keystream dai dati cifrati, se abbiamo sia il testo cifrato che il testo in chiaro originale. Il testo cifrato e' facile da catturare; tutto quello che serve e' uno sniffer wireless, e possiamo raccogliere gigabytes di dati cifrati da qualsiasi rete wireless.

Wireless Siniffing

La qualita' di uno sniffer e' direttamente proporzionale all'informazione che puo' fornire a chi lo usa. Ad esempio, molte persone considerano dsniff il miglior sniffer disponibile - non perche' dsniff catturi i dati meglio di Ethereal, che invece e' in cima alla lista delle preferenze di molti professionisti, ma perche' dsniff incorpora caratteristiche extra, come uno sniffer di password integrato, la possibilita' di fare ARP spoofing ed altro ancora. Questi piccoli extra lo rendono orientato a determinate attivita'. D'altra parte, alcuni troubleshooting hanno bisogno di sniffer hardware\software estremamente costosi.
Questi dispositivi possono collezionare gigabytes di dati e non perdere mai neanche un pacchetto.

L'introduzione delle reti wireless ha determinato la creazione di una nuova nicchia di sniffer. A causa delle carattterstiche fisiche e tecniche uniche delle WLAN, la qualita' e la funzionalita' di uno sniffer wireless sono collegate a quanto bene puo' essere integrato in una rete wireless esistente.

Alcuni sniffer si limitano a catturare i pacchetti della WLAN a cui sono associati, mentre altri possono catturare dati da tutte le reti attive nei paraggi. Per una rete 802.11b, possono essere usati 14 canali di trasmissione. Come risultato, e' possibile avere piu' di 4 WLAN diverse e separate nella stessa area geografica (se ogni rete usa piu' canali).
Per collezionare dati da tutte le reti wireless, il dispositivo wireless su cui sta girando lo sniffer deve operare in modalita' passiva. Questa modalita' consente di catturare tutti i dati in transito, ma il dispositivo non potra' connettersi a nessuna rete wireless. In altre parole, non fara' altro che saltare continuamente tra i vari canali. Per rendere le cose ancora piu' complicate, sniffare una rete wireless in modalita' passiva richiede drivers speciali, o, come minimo, una patch ai drivers che gia' si utilizzano.

Quando una scheda di rete viene prodotta, le viene associato un identificativo univoco conosciuto come indirizzo Media Access Control (MAC Address). Dato che si presuppone che si tratti di un indirizzo unico, e' fondamentale per trasmettere dati all'interno di una rete. Ci sono numerosi protocolli che si appoggiano al MAC Address per funzionare. E' importante capire il significato del MAC Address, perche' incide indirettamente sui dati a cui puo' accedere lo sniffer.

Quando una scheda di rete opera normalmente, effettua la scansione di ogni pacchetto che attraversa la rete per vedere se si tratta di pacchetti destinati al proprio MAC Address. In caso positivo, il dato viene passato al livello successivo della pila dei protocolli, e, infine, arriva al programma a cui era destinato. Se il pacchetto non e' indirizzato al MAC Address della nostra scheda di rete, per motivi pratici viene semplicemente ignorato.

Dato che uno sniffer opera appena sopra al livello hardware della pila dei protocolli di rete, riceve solamente i dati destinati al computer su cui sta girando. In altra parole, lo sniffer vede solamente i dati locali. Se questo livello di accesso puo' essere di aiuto in alcune situazioni, l'accesso limitato rende vani quasi tutti gli sforzi di qualsiasi troubleshooting. E' a questo punto che entra in scena la modalita' promiscua.

Quando una scheda di rete viene messa in modalita' promiscua, accetta tutti i dati che passano sul cavo al quale e' connessa, ignorando completamente il MAC Address.

Ci sono molti esempi di sniffer wireless; un esempio eccellente e' Kismet (http://www.kismetwireless.net). Per i PocketPC, si puo' usare anche Airscanner Mobile Sniffer (http://www.airscanner.com).

Airscanner consente di:
- sniffare pacchetti wireless in modalita' promiscua
- decifrare pacchetti UDP, TCP, Ethernet, DNS e NetBIOS
- condurre analisi di rete su un intero segmento WLAN
- personalizzare i filtri di cattura (IP sorgente, IP destinazione, porta UDP, porta TCP o MAC Address)
- vedere le statistiche sui pacchetti catturati in tempo reale
- salvare i risultati di una sessione si sniffing
- esportare i dati in formato libcap (ad esempio Ethereal) per ulteriori analisi

Estrarre il Keystream

Ora che abbiamo ottenuto uno sniffer wireless per catturare i dati cifrati da una WLAN, possiamo estrarre un keystream, se abbiamo sia il testo cifrato che il testo in chiaro.

Come facciamo a conoscere il valore originale ? Il modo piu' comune con cui predeterminare un testo in chiaro valido e' convincere qualcuno a inviare o ricevere un messaggio predicibile. Per esempio, una sessione di chat o l'invio di una mail potrebbero fornirci tutto il testo in chiaro di cui abbiamo bisogno. Comunque, questo metodo puo' essere difficile se dati estranei vengono mischiati coi dati predicibili. Ad esempio, i pacchetti TCP\IP contengono intestazioni IP e altre informazioni che potrebbero distrarci. I checksum, dati proprietari aggiunti dai server di posta, e altro ancora potrebbero oscurare i dati predicibili. Quindi, se si vuole andare avanti con questo metodo, e' necessario inviare un messaggio che aumenti la possibilita' di ottenere dati predicibili. Si potrebbe fare semplicemente inviando una email piena di spazi vuoti (per esempio "   "), o una stringa lunga composta dallo stesso carattere (per esempio "AAAAAAAAAAAAAAAA”).

Un altro metodo usato per predeterminare il testo in chiaro e' cercare intestazioni conosciute. I pacchetti TCP\IP contengono le intestazioni IP che sono richieste per assicurarne una corretta consegna. Se siamo in grado di predeterminare l'indirizzo IP dell'access point o di un client wireless e indovinare il resto dei dati basandoci sulle abitudini degli utenti di quella rete, possiamo dedurre il testo in chiaro. Quasi tutti i pacchetti includono un'intestazione SNAP come primo byte.

Assumendo che e' possibile determinare il testo in chiaro di un messaggio ed utilizzarlo per racimolare il keystream, cosa possiamo fare con questa informazione ? La risposta dovrebbe essere chiara. Bisogna sottolineare che un keystream, o anche una coppia di keystream, in se' sono praticamente inutili.

E' quando combini la conoscenza guadagnata in questo tipo di attacco con altre tecniche di hacking wireless che il potere della conoscenza di un keystream diventa manifesto.

Collisione IV

WEP usa un valore chiamato vettore di inizializzazione, meglio noto come IV. L'algoritmo RC4 usa questo valore per cifrare ogni pacchetto con la sua chiave, unendo o concatenando la password condivisa con l'IV per creare una packet key nuova ed esclusiva per ogni pacchetto di informazione inviato sulla WLAN.

Comunque, se il mittente usa un IV per cifrare il pacchetto, il ricevente deve conoscere anche questo bit di informazione per decifrare i dati. A causa del modo in cui il WEP e' stato implementato, questo requisito si e' trasformato da forza apparente a debolezza reale.

WEP usa un IV di 3 bytes per ogni pacchetto trasmesso sulla WLAN. Quando il dato viene inviato, l'IV viene anteposto al pacchetto cifrato. Questo assicura al ricevente l'informazione necessaria alla decifratura del dato. Se guardiamo piu' da vicino la natura statistica di questo processo, vedremo subito un potenziale problema. Un byte sono 8 bits. Quindi, la grandezza totale dell'IV e' di 24 bits (8 bits x 3 bytes). Se calcoliamo tutti i possibili IV, avremo una lista di 2^24 possibili chiavi. Questo numero e' derivato dal fatto che un bit puo' essere settato a 0 o a 1 (2), e c'e' un numero di 24 bits totali.
Potrebbe sembrare un numero enorme (16.777.216), ma si tratta di un numero piccolo se associato alla comunicazione. La ragione risiede nella probabilita' di ripetizioni.

L'IV e' un numero casuale. Quando la maggior parte delle persone relazionano la parola "casuale" a un numero come 16.777.216, la prima cosa che viene loro in mente e' che bisognera' aspettare il trasferimento di 16 milioni di pacchetti prima che si verifichi una ripetizione. Questo e' completamente falso.

Infatti, basandoci sulla probabilita', possiamo ragionevolmente aspettarci di cominciare a vedere ripetizioni (conosciute anche come collisioni) dopo circa 5000 pacchetti trasmessi, se non meno. Considerando che in media un dispositivo wireless trasmette pacchetti da 1.500 bytes, ci aspettiamo che una collisione si verifichi gia' durante un trasferimento di un file da 7-10 MB (5000 pacchetti x 1500 bytes = 7000000 btyes, o 7 MB).

Il keystream viene prodotto da varie proprieta' della password e dell'IV. In caso di collisione, l'IV e' conosciuto come un valore di tre caratteri "1:2:3".
Anche se non conosciamo la password e' irrilevante, perche' non cambia mai. Possiamo a questo punto dedurre i keystreams generati dai valori IV corrispondenti.

Questa falla non e' in se il maggior problema del WEP, quanto la dimensione piccola dell'IV. Se l'IV fosse molto piu' grande, il tempo necessario ad una sua ripetizione sarebbe molto maggiore, e si creerebbe uno scenario molto piu' difficile per inviare dati predicibili sulla rete. Considerando che un pacchetto e' generalmente grande 1500 bytes e l'IV solamente 3 bytes, avremmo molte piu' possibilita'. Comunque, nel nome della velocita' e per massimizzare il flusso dati, i progettisti del protocollo hanno ridotto la grandezza dell'IV.

WEP Cracking Pratico

Ora che abbiamo visto la teoria, esaminiamo gli step pratici per attaccare il WEP. La risorsa piu' importante per crackare un segnale cifrato col WEP e' il tempo. Piu' a lungo catturiamo i dati, piu' saremo vicini a ricevere una collisione che rivelera' un byte chiave.

Basandosi su dati empirici, c'e' una possibilita' del 5% che questo accada. In media, avremo bisogno di ricevere circa 5 milioni di frames per essere in grado di crackare un segnale cifrato col WEP. Oltre allo sniffer wireless, avremo bisogno di una serie di script Perl disponibili su http://sourceforge.net/projects/wepcrack, chiamati (giustamente) WEPCRACK.

Una volta raccolti gli strumenti necessari, bisognera' eseguire i seguenti step:

1) catturare il segnale cifrato col WEP utilizzando il nostro sniffer wireless preferito (circa 5 milioni di frames)
2) da riga di comando, eseguire lo script prism-getIV.pl in questo modo:
prism-getIV.pl capturefile_name
dove capturefile_name e' il nome del file di cattura dello step 1. Quando viene trovato un IV debole, il programma crea un file IVfile.log
3) eseguire WEPcrack.pl, che guarda gli IV registrati dentro IVfile.log e cercare di indovinare una chiave WEP. L'output di WEPcrack.pl e' in formato decimale. Avremo bisogno di un tool di conversione decimal-to-Hex
4) Prendiamo la versione Hex della chiave e ... abbiamo fatto !