Felipe Almeida Lessa

O protocolo WEP: Sigilo contra Acidentes

16 de dezembro de 2009

Felipe Almeida Lessa

O protocolo WEP: Sigilo contra Acidentes

Professor:
Pedro A. D. Rezende

Universidade de Brasília

16 de dezembro de 2009

Sumário

1 Introdução

A redes de computadores sem fio hoje são ubíquas, estando presentes em celulares, *notebooks* e vários outros dispositivos. O padrão mais utilizado, popularmente referido simplesmente por Wi-Fi, é o IEEE 802.11, com *drafts* ainda sendo escritos e evoluindo com o tempo. Como o meio de transmissão são as ondas de rádio, as redes sem fio são mais sucetíveis a ataques de escuta e portanto violação de sigilo. De acordo com uma pesquisa feita em 2003, é prático escutar redes sem fio protegidas com WEP a uma distância de uma milha (aprox. 1,6 km) ou mais [1]. Também são vulneráveis a ataques de *man-in-the-middle* porque o atacante não precisa estar no meio, pode estar em qualquer ponto ao alcance do *access point*.

Este documento está organizado da seguinta maneira: no Capítulo 2 o protocolo WEP e sua cifra são apresentados com detalhes. No Capítulo 3 são listadas as vulnerabilidades descobertas ao longo do tempo. No Capítulo 4 as principais ferramentas utilizadas hoje para atacar redes WEP são descritas. No Capítulo 5 são apresentadas as soluções ao problema de sigilo (e integridade) em redes sem fio.

2 O protocolo WEP

Introduzido em 1997 [2], desde sua primeira versão o padrão já especificava o protocolo *Wired Equivalent Privacy* (lit. privacidade equivalente a cabos), ou simplesmente WEP. De acordo com o padrão IEEE 802.11-1997,

> *IEEE 802.11 specifies a wired LAN equivalent data confidentiality algorithm. Wired equivalent privacy is defined as protecting authorized users of a wireless LAN from casual eavesdropping. This service is intended to provide functionality for the wireless LAN equivalent to that provided by the physical security attributes inherent to a wired medium.* [2]

Em particular, o protocolo WEP não tinha como objetivo impedir que usuários legítimos da rede (possuidores da chave) violassem o sigilo das comunicações dos outros usuários. Em uma rede cabeada uma interface de rede no modo promíscuo pode realizar a mesma ação. Por outro lado, sem acesso físico ao cabo um atacante não pode escutar o tráfego de dados. O protocolo WEP tem como objetivo que um atacante possa escutar o tráfego de dados na rede sem fio. Contudo hoje o padrão WEP é considerado inseguro e provê segurança apenas contra conexões acidentais; qualquer atacante determinado utilizando *hardware* de prateleira (e.g. qualquer *notebook* produzido nos últimos cinco anos) pode obter acesso total à rede.

Ainda de acordo com o padrão de 1997, o protocolo WEP possui as seguintes características:

É razoavelmente forte A segurança do protocolo está baseada na dificuldade em adivinhar a chave.

É auto-sincronizante A cada mensagem o protocolo se auto-sincroniza, visto que o nível de perda de pacotes pode ser n alto.

É eficiente, exportável e opcional Pode ser implementado em *hardware* ou *software*.

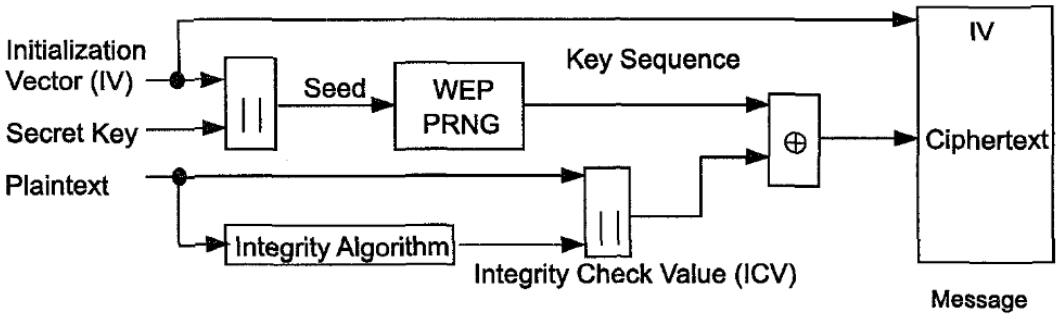

Figura 2.1: Diagrama ilustrando o funcionamento do protocolo WEP e o uso de sua cifra. Imagem obtida de [2].

A cifra usada pelo protocolo WEP é uma cifra encadeada simétrica. A chave é concatenada a um vetor de inicialização (IV) e usada em um gerador de números pseudo-aleatórios (PRNG). É feito um XOR *bit* a *bit* da sequência de saída do PRNG com o texto pleno concatenado com um código de integridade (implementado como um CRC). O texto cifrado resultante é concatenado com o IV mantido às claras (pois ele é necessário para a decriptação) e o resultado é enviado pelo meio inseguro da rede sem fio. A Figura 2.1 ilustra o processo.

O padrão então adota como cifra (i.e. como PRNG) o RC4, desenvolvido originalmente por Ron Rivest da RSA Security em 1987 [3]. Ele foi escolhido porque é extremamente simples e possui implementações eficientes em *hardware* e *software*. Seu algoritmo opera *byte* a *byte* e se baseia em uma permutação dos 256 possíveis *bytes* e dois índices dessa permutação. A permutação inicial é escolhida através de um algoritmo de geração de chave (*key-scheduling algorithm*, KSA). A Wikipedia apresenta uma implementação funcional e simples do RC4 [3]:

```
typedef unsigned char byte;
byte S[256];
unsigned int i, j;
void swap(byte *s, unsigned int i, unsigned int j) {
    byte temp = s[i]; s[i] = s[j]; s[j] = temp;
}
void rc4_init(byte *key, unsigned int key_length) {
    for (i = 0; i < 256; i++) /* Permutação identidade. */
        S[i] = i;
    for (i = j = 0; i < 256; i++) {
        j = (j + key[i % key_length] + S[i]) & 255;
        swap(S, i, j);
```

```
    }
    i = j = 0;
}
byte rc4_output() {
    i = (i + 1) & 255;
    j = (j + S[i]) & 255;
    swap(S, i, j);
    return S[(S[i] + S[j]) & 255];
}
```

Da maneira como foi implementado acima, basta invocar `rc4_init` uma vez e então cada invocação de `rc4_output` retorna um *byte* do *stream* usado para fazer XOR com o texto pleno.

Como a saída do PRNG é usada com XOR, uma mesma saída não pode ser usada mais de uma vez. Caso contrário, suponha que um atacante obtém acesso a $c_1 = w \oplus m_1$ e a $c_2 = w \oplus m_2$, onde m_i são as mensagens em texto pleno, $\oplus$ é a operação XOR e w é a saída do RC4. Basta o atacante fazer

$$\begin{aligned} c_1 \oplus c_2 &= (w \oplus m_1) \oplus (w \oplus m_2) && \text{(associatividade)} \\ &= w \oplus m_1 \oplus w \oplus m_2 && \text{(comutatividade)} \\ &= \cancel{w \oplus w} \oplus m_1 \oplus m_2 && (a \oplus a = 0) \\ &= m_1 \oplus m_2 \end{aligned}$$

para obter acesso ao XOR dos textos plenos. Uma vez feito isso há uma variedade de técnicas de criptoanálise que podem ser aplicadas. Este problema não é restrito ao RC4 e é consequência do desenho das cifras encadeadas. Mas ao contrário de cifras encadeadas mais recentes, o RC4 não possui entrada específica para um IV e portanto o cliente do algoritmo deve gerenciar as diferentes chaves. Neste caso o protocolo WEP concatena a chave compartilhada entre os usuários legítimos com o IV do pacote.

O protocolo WEP pode ser utilizado com chaves de 40 *bits* e IVs de 24 *bits* ou com chaves de 104 *bits* e IVs de mesmo tamanho. Com 24 bits existem apenas 16,7 milhões de IVs possíveis. Depois de apenas 5.000 pacotes há uma probabilidade de 50% de que o mesmo IV seja repetido.

As vulnerabilidades do protocolo WEP são provenientes de uma combinação de dois fatores: avanços em criptoanálise do RC4 e mal-uso do RC4 pelo protocolo WEP.

3 *Vulnerabilidades*

Atualmente é possível extrair a chave (independentemente de qual ela seja) de uma rede sem fio que utiliza o protocolo WEP *em menos de cinco minutos*. Além disso ela pode ser extraída tanto a partir do *access point* quanto a partir dos seus clientes. Neste capítulo nós veremos como o WEP foi se tornando tão inseguro ao longo do tempo.

3.1 Andres Roos – 1995

Dois anos antes da homologação da primeira versão do padrão IEEE 802.11, Andrew Roos experimentalmente observou que o primeiro byte de saída do RC4 é correlacionado aos três primeiros bytes da chave [4]. Na ocasião Roos já recomendou que os primeiros *bytes* fossem descartados:

> *After initializing the algorithm, generate and discard a number of bytes. Since the algorithm used to generate bytes also introduces additional non-linear dependencies into the state table, this would make analysis more difficult* [4].

Talvez porque Roos não tenha provado teoricamente a correlação, ou talvez porque não havia ainda um *exploit*, o protocolo WEP não descarta os primeiros *bytes* do RC4.

3.2 Fluhrer, Mantin e Shamir – 2001

Quatro anos após a publicação definitiva do protocolo WEP, Fluhrer, Mantin e Shamir publicaram o primeiro ataque prático às redes sem fio protegidas pelo WEP [5]. Observando que havia uma classe de IVs fracos na forma como o WEP os utiliza (concatenando com a chave), o ataque desvela os *bytes* utilizados na chave um a um utilizando um processo iterativo. Para cada *byte*, as mensagens captadas são analisadas e a partir dela extrai-se a distribuição de probabilidades dos valores do próximo *byte* da chave. O *byte* correto então pode ser descoberto pois ele é o que possui maior probabi-

lidade. Para facilitar ainda mais o ataque, o primeiro *byte* do texto pleno dos pacotes é um cabeçalho de valor `0xAA` [6].

Para o ataque ser bem sucedido são necessários entre 4.000.000 e 6.000.000 de pacotes [7]. Em uma rede sem fio movimentada essa quantidade pode ser obtida passivamente em menos de 15 minutos. Também há formas de estimular tráfego ativamente.

Uma implementação aberta e pronta para o uso do ataque de Fluhrer et al. foi construída, o aircrack. Com ele é possível utilizar *hardware* convencional para interceptar os pacotes colocando a interface da rede sem fio em modo promíscuo. Concomitantemente, ou em um momento posterior se desejado, os pacotes capturados são analisados e a chave é derivada a partir deles.

3.3 KoreK – 2004

Em junho de 2004 um *hacker* autodenominado KoreK publicou em um fórum uma implementação capaz de desvelar a chave utilizando menos tempo, precisando entre 500.000 e 2.000.000 de pacotes [8, 7]. Em meados de agosto do mesmo ano o aircrack implementou seu ataque [9].

KoreK também descreveu em linhas gerais um ataque diferente, conhecido atualmente como chopchop capaz de decodificar um pacote WEP com a ajuda do *access point*. Um a um os bytes são cortados do final do pacote (por isso *chopping attack*). Porém ao fazer isso o CRC é quebrado e o pacote, rejeitado. O chopchop então consiste em utilizar o XOR para modificar de uma forma previsível o CRC (um *bit-flipping attack*) e verificar qual alteração torna o pacote válido. Repetindo o ataque o pacote inteiro pode ser revelado sem descobrir a chave utilizada.

3.4 Pychkine, Tews e Weinmann – 2007

Em 2005, Adreas Klein publicou uma nova criptoanálise da cifra RC4 [10]. Ele melhorou o ataque de Fluhrer et al. para funcionar mesmo que os IVs fracos não fossem utilizados e mesmo que os primeiros 256 *bytes* do RC4 fossem descartados.

Baseando-se no trabalho de Klein, em abril de 2007 foi desenvolvido por Pychkine, Tews e Weinmann o aircrack-ptw [7]. Segundo os autores, com esta técnica é possível recuperar a chave WEP capturando apenas 40.000 pacotes com uma chance de 50%,

ou 85.000 pacotes com uma probabilidade de 95%. Esse número de pacotes pode ser obtido em menos de um minuto. A computação em si levava 3 segundos e utilizava 3 MiB de memória em um Pentium-M de 1,7 GHz utilizado nos testes dos autores [7].

3.5 Ramachandran e Ahmad – 2007

Em outubro do mesmo ano de 2007 a AirTight Networks anunciou um novo ataque inovador [11]. O ataque Caffe Latte é direcionado a clientes de redes sem fio, e não aos *access points*. Ele é assim denominado porque pode ser conduzido enquanto um cliente de uma rede protegida pelo WEP está tomando um café com leite usando seu *notebook*.

O conceito do ataque é bem simples. Os *notebooks* de clientes de redes que usam o WEP guardam em seu chaveiro a chave da rede. A chave não é enviada para o *access point*, porém ela é utilizada para encriptar o desafio enviado por ele.

O ataque então consiste de criar um *access point* falso com o endereço MAC e o nome ESSID do *access point* que o cliente conhece. O sistema operacional do *notebook* do cliente o reconhece e tenta conectar a ele automaticamente. Um desafio é enviado e o cliente o encripta utilizando a chave legítima (de onde vários bytes do RC4 podem ser obtidos pois o desafio foi criado pelo atacante). O *access point* falso aceita o cliente (mesmo sem verificar a chave) e utiliza pacotes ARP enviados pelo mesmo para realizar injeção e captura de pacotes como seria feito com um ataque convencional. Tendo os pacotes basta usar o mesmo aircrack-ng para obter a chave. Todo o ataque leva menos de 6 minutos.

3.6 Darknet – 2007

Um pouco antes, em janeiro de 2007, um grupo de *hackers* autodenominado Darknet divulgou um ataque semelhante ao Caffe Latte porém utilizando fragmentação de pacotes [12]. O princípio é o mesmo: simular um *access point* falso e obter pacotes do cliente. Eles implementaram o ataque em uma ferramenta chamada Wep0ff.

4 *Ferramentas*

Existem hoje várias ferramentas que tornam quase trivial o processo de obter a chave de uma rede sem fio "protegida" pelo protocolo WEP. Todas as ferramentas descritas abaixo são livres e de fácil acesso. As principais distribuições GNU/Linux possuem a maioria delas em seus repositórios de *software*.

4.1 aircrack-ng

O aircrack-ng é a principal coleção de ferramentas para o trabalho. Possui diversas ferramentas voltadas para ataques a redes sem fio IEEE 802.11. Disponível em `http://www.aircrack-ng.org/`.

4.1.1 História

De acordo com o diário de alterações do projeto [9]:

- Em julho de 2004 foi lançado sua primeira versão documentada no diário, o aircrack 1.0 (ainda sem o sufixo "-ng").
- Um mês depois, em agosto do mesmo ano, foi implementado o ataque de KoreK [8]. Ele permaneceu o ataque padrão ao WEP no aircrack e no aircrack-ng até outubro de 2007.
- Em março de 2006 o aircrack tornou-se aircrack-ng com uma reestruturação da organização dos aplicativos (que também ganharam o novo sufixo).
- Em abril de 2007 o aircrack-ptw foi criado pelos próprios autores do ataque PTW. Ele era uma ferramenta "prova de conceito" e "bem similar ao aircrack-ng" [7].
- Em maio de 2007 o código do aircrack-ptw foi incorporado ao aircrack-ng 0.9.

Porém apenas em outubro de 2007 o ataque PTW foi tornado padrão, enquanto o ataque de KoreK foi mantido como uma opção.

- Também em outubro de 2007 foram lançados o easside-ng e o wesside-ng, este para obter de chaves WEP automaticamente e aquele para comunicar com uma rede WEP sem saber a chave.
- Em junho de 2008 foi criado o airbase-ng, com ataques voltados aos clientes e não aos *access points* das redes sem fio.

4.1.2 Aplicações

O pacote do aircrack-ng contém todas as ferramentas necessárias para a realização de ataques ao WEP, desde alterar a placa de rede sem fio para o modo promíscuo e a obtenção dos pacotes até a realização dos cálculos para o ataque em si.

As principais ferramentas são:

airmon-ng Altera uma placa de rede sem fio para o modo promíscuo.

airodump-ng O *sniffer* de pacotes, capta-os e os salva.

aireplay-ng O injetor de pacotes, para realizar ataques ativos para gerar tráfico ou para o chopchop.

aircrack-ng Levando o nome do pacote, é o *cracker* que implementa os ataques descritos (cf. Capítulo 3). Pode funcionar em modo *on-line* (simultaneamente à captura) ou *off-line* (com um arquivo de captura salvo de outro momento).

Também estão inclusas:

packetforge-ng Cria pacotes encriptados para uso em ataques de injeção. Não requer a chave, apenas uma sequência de *bytes* do RC4 para um IV qualquer (visto que no protocolo WEP o cliente escolhe seus IVs).

airdecap-ng Decripta pacotes WEP e WPA salvos com uma chave conhecida. Após um ataque bem sucedido, pode ser usada para revelar o texto pleno que trafegava.

airtun-ng Cria uma interface virtual com um túnel para uma rede sem fio. Pode ser usado para decifrar uma rede protegida com WEP ou WPA (cuja chave seja conhecida) e direcionar o tráfego a um sistema de detector de intrusos (IDS); também pode ser usado para injetar tráfego a partir de ferramentas convencionais.

airolib-ng Gerencia um banco de dados de senhas e pré-computa chaves a partir delas. Usado para aumentar a eficiência de ataques de dicionário a redes WPA.

airbase-ng Ferramenta versátil contendo vários ataques a clientes de redes sem fio. Implementa ataques como o Caffe Latte [11] e o de fragmentação [12].

airdecloak-ng Usado para remover pacotes espúrios que podem ser injetados por equipamentos que tentam prevenir que a chave seja quebrada.

airdriver-ng Usado para gerenciar os *drivers* das places de rede sem fio do sistema.

airserv-ng Servidor TCP para que clientes possam acessar uma placa de rede sem fio remotamente.

easside-ng Usada para comunicar com uma rede sem fio com WEP sem saber a chave. Utiliza um servidor auxiliar disponível na Internet para "pedir" para o *access point* decriptar pacotes por ele.

wesside-ng Utiliza uma combinação de ataques para automaticamente obter uma chave WEP em poucos minutos. Os ataques podem ser feitos manualmente mas a ferramente os automatiza.

tkiptun-ng Uma prova de conceito de ataque ao WPA/TKIP.

4.2 Kismet

Um *sniffer* capaz de detectar redes sem fio, capturar pacotes e detectar certos tipos de intrusos. Funciona totalmente passivamente (como o airodump-ng). Também possui um modo para *wardriving*, podendo anunciar em voz alta as redes encontradas, e integração com GPS (para salvar a posição aproximada do *access point*). Disponível em `http://www.kismetwireless.net`.

4.3 Wireshark

Antigo Ethereal, o Wireshark é o venerável analisador de pacotes. Os formatos em disco usados para salvar e ler os pacotes utilizados por todas as ferramentas aqui descritas é o mesmo, o PCAP. Portanto o Wireshark pode ser usado em várias etapas para analisar o tráfego, especialmente em conjunto com suas centenas de ferramentas de análise dos mais diversos protocolos de rede (em várias camadas). Em

conjunto com o airdecap-ng ele pode ser usado para decriptar pacotes de redes cujas chaves foram obtidas utilizando qualquer dos ataques já descritos. Disponível em http://www.wireshark.org/.

4.4 BackTrack

Uma distribuição GNU/Linux distribuída como *Live CD* (ou *Live USB*) usada para testes de penetração. Contém centenas de ferramentas e *drivers* prontos para serem usados e que abrangem uma variedade de diferentes tipos de alvos e técnicas de ataque. Em particular, todas as ferramentas citadas aqui estão inclusas. Portanto mesmo um usuário que não deseje instalar uma distribuição GNU/Linux pode quebrar uma chave WEP tendo o BackTrack em um *pen drive*. Disponível em http://www.remote-exploit.org/.

5 *Soluções*

A única solução completa é o WPA2/CCMP, definido no padrão IEEE 802.11i-2004 [13]. Contudo nós veremos neste capítulo também o WPA/TKIP, uma solução temporária proposta pelo IEEE.

5.1 Temporal Key Integrity Protocol

Em vista do primeiro ataque prático público em 2001 por Fluhrer, Mantin e Shamir, o comitê padronizador IEEE 802 procurou construir rapidamente uma solução para o problema. Com apenas o WEP para a proteção a nível de enlace, e com ele estando exposto, as redes sem fio sofreram um grande golpe. Uma solução devia ser achada em pouco tempo e sem requerer uma troca massiva de *hardware*.

Enquanto o comitê trabalhava houveram ideias que acabaram não sendo padronizadas [14], como:

WEP2 Rascunhos iniciais do padrão IEEE 802.11i tinham uma "melhoria" ao WEP que expandia o tamanho das chaves e dos IVs para 128 *bits*. Como o maior problema do WEP é o próprio protocolo, e não os tamanhos, a idéia foi abandonada.

WEPplus Extensão proprietária da Agere Systems ao WEP que evita os IVs fracos. Só é útil se tanto o *access point* quanto os clientes usarem equipamentos compatíveis. Sua segurança não foi compravada, também porque seu uso não foi muito disseminado.

Dynamic WEP Outra extensão proprietária, usada por fabricantes como a 3Com, que troca as chaves dinamicamente. Também enfrentou problemas de adoção.

Em outubro de 2002, a Wi-Fi Alliance, detentora da marca "Wi-Fi", aprovou o uso de um rascunho do Temporal Key Integrity Protocol, ou TKIP, como o Wi-Fi Protected

Access, ou WPA (para evitar confusões nós escrevemos aqui "WPA/TKIP"). O WPA/TKIP procura dar uma solução a curto prazo para a falta de segurança. Ele pode ser implementado com uma atualização de *firmware* a grande parte dos *hardwares* que implementavam somente o protocolo WEP, o que facilitou sua adoção.

Em seu núcleo o WPA/TKIP ainda utiliza o RC4, porém ele implementa diversos recursos para tentar contornar as deficiências mais graves do protocolo WEP:

- Uma função de mistura é usada para combinar a chave com o IV antes de usá-lo com o RC4, ao invés de concatená-los. Vários ataques ao WEP tomavam partido desta vulnerabilidade.
- Um contador sequencial evita que ataques de *replay* sejam conduzidos. As técnicas de geração de tráfego geralmente se utilizam da facilidade de reinjetar os mesmos pacotes recebidos (ou com pequenas alterações).
- Um novo verificador de integridade de 64 *bits*, denominado MICHAEL, é usado.
- As chaves utilizadas são trocadas com frequência (por padrão, a cada 10.000 pacotes), dando o "temporal" ao nome do WPA/TKIP.

Apesar de até hoje a situação do WPA/TKIP estar muito melhor do que a do WEP, ele não deve ser usado sempre que for possível utilizar o WPA2/CCMP. Já existem ataques práticos ao protocolo que no mínimo abrem espaço para o atacante tentar conduzir um outro ataque, como um *man-in-the-middle* [15, 16].

O WPA/TKIP já chegou ao final de sua vida útil prevista e será retirado do próximo padrão IEEE 802.11.

5.2 Counter Mode with CBC-MAC Protocol

Em junho de 2004 o padrão IEEE 802.11i foi ratificado, padronizando o WPA/TKIP e o WPA2/CCMP.

O WPA2/CCMP é um protocolo novo, diferente do WEP. Ele utiliza o robusto AES (Advanced Encryption Standard, antigo Rijndael) como cifra interna. Nenhum dos ataques descritos aqui funciona com o WPA2/CCMP, nem mesmo os que se aplicam ao WPA/TKIP (apesar dos nomes "WPA" e "WPA2" serem semelhantes, os protocolos são bem distintos). Também não é conhecido nenhum ataque prático novo ao WPA2/CCMP.

Principalmente por usar uma cifra diferente da usada no WEP, o WPA2/CCMP em geral não pode ser implementado em *hardwares* feitos apenas com o *WEP* em mente. Com isso muitas organizações que procuravam sair da insegurança do WEP continuam utilizando o WPA/TKIP até hoje, apesar de seu uso não ser recomendável.

6 Conclusão

Atualmente manter uma rede sem fio aberta (i.e. sem encriptação) ou utilizar o WEP tem apenas uma única diferença: usuários honestos de outras redes sem fio não vão poder conectar-se à sua rede acidentalmente. Qualquer atacante, mesmo os mais inexperientes, podem realizar ataques bem sucedidos; para os que não gostam de ler manuais e *papers*, basta ver qualquer vídeo que explique o processo passo a passo.

O WPA/TKIP, criado para estancar emergencialmente o surgimento de ataques às redes sem fio IEEE 802.11, é suportado pela maioria dos dispositivos que também suporta WEP. Por isso é recomendável que todos os usuários do WEP utilizem o WPA/TKIP em seu lugar. Porém desde seu desenvolvimento já sabia-se que era uma questão de tempo até ataques práticos ao WPA/TKIP surgirem (além de ataques de dicionário). Hoje já há vários ataques que, apesar de não obterem a chave, podem ser utilizados como base para outros. Por fim, o WPA/TKIP também já foi descontinuado (*deprecated*) e será removido de versões futuras do padrão IEEE 802.11.

Portanto é recomendado que seja utilizado o WPA2/CCMP com senhas seguras sempre que possível. Além de força bruta (o que é inviável) e ataque de dicionário, ataques que sempre poderão ser realizados, não há nenhum ataque conhecido hoje ao WPA2/CCMP. Salvo sejam usadas senhas fracas, uma rede sem fio protegida por ele fica vulnerável apenas a ataques internos de seus próprios usuários legítimos.

Referências Bibliográficas

[1] CAM-WINGET, N. et al. Security flaws in 802.11 data link protocols. *Communications of the ACM*, v. 46, n. 5, p. 35–39, 2003.

[2] IEEE Std 802.11-1997 Information Technology- telecommunications And Information exchange Between Systems-Local And Metropolitan Area Networks-specific Requirements-part 11: Wireless Lan Medium Access Control (MAC) And Physical Layer (PHY) Specifications. IEEE, 1997. 459 p. ISBN 1-55937-935-9. Disponível em: <http://ieeexplore.ieee.org/servlet/opac?punumber=5258>.

[3] WIKIPEDIA. *RC4 — Wikipedia, The Free Encyclopedia*. 2009. Disponível em: <http://en.wikipedia.org/w/index.php?title=RC4&oldid=330373630>. Acesso em: 16 dez. 2009.

[4] ROOS, A. A class of weak keys in the RC4 stream cipher. 1995. Dois posts em sci.crypt, message-id 43u1eh$1j3@hermes.is.co.za e 44ebge$llf@hermes.is.co.za. Disponível em: <http://groups.google.com/group/sci.crypt.research/msg/078aa9249d76eacc?dmode=source>. Acesso em: 14 dez. 2009.

[5] FLUHRER, S.; MANTIN, I.; SHAMIR, A. Weaknesses in the key scheduling algorithm of rc4. *Eighth Annual Workshop on Selected Areas in Cryptography*, 2001.

[6] WIKIPEDIA. *Fluhrer, Mantin, and Shamir attack — Wikipedia, The Free Encyclopedia*. 2009. Disponível em: <http://en.wikipedia.org/w/index.php?title=Fluhrer,_Mantin,_and_Shamir_attack&oldid=319703144>. Acesso em: 15 dez. 2009.

[7] TEWS, E.; PYCHKINE, A.; WEINMANN, R.-P. *aircrack-ptr*. 2005. Disponível em: <http://www.cdc.informatik.tu-darmstadt.de/aircrack-ptw/>. Acesso em: 15 dez. 2009.

[8] KOREK. *Need security pointers*. 2004. Resposta a uma pergunta em um fórum. Disponível em: <http://www.netstumbler.org/f49/need-security-pointers-11869/>. Acesso em: 15 dez. 2009.

[9] CHANGELOG do aircrack-ng. Distribuído com o código-fonte do projeto. Disponível em: <http://trac.aircrack-ng.org/svn/trunk/ChangeLog>. Acesso em: 17 dez. 2009.

[10] KLEIN, A. *Attacks on the RC4 stream cipher*. 2006. Disponível em: <http://cage.ugent.be/~klein/RC4/>. Acesso em: 16 dez. 2009.

[11] RAMACHANDRAN, V.; AHMAD, M. D. S. *Caffé Latte with a Free Topping of Cracked WEP: Retrieving WEP Keys From Road-Warriors*. Disponível em:

<http://www.airtightnetworks.com/home/resources/knowledge-center/caffe-latte.html>. Acesso em: 18 dez. 2009.

[12] DARKNET. *Wep0ff – Wireless WEP Key Cracker Tool*. Disponível em: <http://www.darknet.org.uk/2007/01/wep0ff-wireless-wep-key-cracker-tool/>. Acesso em: 18 dez. 2009.

[13] IEEE 802.11i-2004: Amendment 6: Medium Access Control (MAC) Security Enhancements. IEEE, 2004. Disponível em: <http://standards.ieee.org/getieee802/download/802.11i-2004.pdf>.

[14] WIKIPEDIA. *Wired Equivalent Privacy — Wikipedia, The Free Encyclopedia*. 2009. Disponível em: <http://en.wikipedia.org/w/index.php?title=Wired_Equivalent_Privacy&oldid=332281037>. Acesso em: 18 dez. 2009.

[15] TEWS, E.; BECK, M. Practical attacks against wep and wpa. 2008. Disponível em: <http://dl.aircrack-ng.org/breakingwepandwpa.pdf>. Acesso em: 18 dez. 2009.

[16] OHIGASHI, T.; MORII, M. A practical message falsification attack on WPA. 2009. Disponível em: <http://www.packetstormsecurity.org/papers/wireless/A_Practical_Message_Falsification_Attack_On_WPA.pdf>. Acesso em: 18 dez. 2009.