Um Estudo Sobre o *WEP* e Propostas de Alternativas Para a Melhoria da Segurança em Redes Sem Fio 802.11b*

Fernando C. A. Verissimo e Luís Felipe M. de Moraes

Laboratório de Redes de Alta Velocidade - RAVEL
COPPE/Programa de Engenharia de Sistemas e Computação
Universidade Federal do Rio de Janeiro - UFRJ
Caixa Postal: 68.511 - 21941-972 - Rio de Janeiro - RJ - Brasil
{verissimo,moraes}@ravel.ufrj.br

Resumo

Este artigo apresenta uma abordagem sobre as características do protocolo *WEP* (*Wired Equivalent Privacy*), utilizado para segurança em redes sem fio que aderem ao padrão *IEEE 802.11b*, enfatizando as suas principais deficiências. Com base nas vulnerabilidades existentes no *WEP* e em observações feitas através de um levantamento realizado em diversas redes da cidade do Rio de Janeiro, conclui-se sobre a importância em estabelecer regras de proteção que não levem em consideração a implementação atual do protocolo. Um conjunto de regras básicas, visando dar segurança às redes sem fio atuais é apresentado e comentado. Finalmente, são apresentadas soluções alternativas ao *WEP*, que visam eliminar os problemas atualmente existentes.

Palavras-chave: Redes sem Fio, *WEP*, Protolocos de Segurança, *IEEE 802.11b*.

Abstract

This article presents a view of the features of the WEP (Wired Equivalent Privacy) protocol, which is used in the security in wireless networks that adhere to the IEEE 802.11b standard, emphasizing its major flaws. The article is based on known WEP vulnerabilities and the observations made on a survey done in several networks of the city of Rio de Janeiro. It concludes on the importance in establishing basic rules of protection that ignore the current implementation of the protocol. A set of basic rules aiming at the improvement of the security of the current wireless networks is presented and commented. Concluding, it is presented alternative solutions to the WEP, that aim to eliminate the current existing problems.

Keywords: *Wireless Networks, WEP, Security Protocols, IEEE 802.11b.*

1 Introdução

A grande proliferação das tecnologias de comunicação em meios não confinados (sem fios) deverá resultar num uso cada vez maior de dispositivos sem fio (*wireless*), para as mais diversas aplicações e serviços. É fácil verificar o grande crescimento de dispositivos móveis sendo comercializados. Uma previsão feita neste cenário em abril de 2002, [1], indica que na metade daquele ano seria ultrapassada a barreira de 1 bilhão de celulares no mundo. Em novembro de 2002, a ANATEL (Agência Nacional de

*Esse trabalho teve o suporte da FAPERJ (Fundação Carlos Chagas Filho de Amparo à Pesquisa do Estado do Rio de Janeiro)

Telecomunicações) divulgou a existência de 32 milhões de celulares ativos no Brasil, [2]. No entanto, existe ainda um número pequeno de serviços oferecidos através de dispositivos sem fio, em relação ao enorme potencial existente. Acredita-se que com o aumento na oferta de produtos e aplicações, a demanda gerada pelos usuários será explosiva.

Neste cenário, as redes sem fio para comunicações móveis entre terminais de dados deverão ser responsáveis pelo suporte a uma grande parcela das aplicações utilizadas. Por exemplo, redes locais sem fio já se constituem numa excelente alternativa para integração de dispositivos diversos numa área restrita (*WLAN*s). Além disso, alguns dos padrões utilizados para interligar entre si equipamentos sem fio, com utilização de pontos de acesso para redes cabeadas (redes estruturadas), estão também sendo propostos para utilização em redes *ad hoc*.

Entre os padrões disponíveis, o *IEEE 802.11b* [3] é atualmente o mais utilizado pelo mercado. Principalmente a partir do ponto em que o *WECA*[1] (*Wireless Ethercnet Compatibility*) conseguiu acordar, entre os vários fabricantes de produtos que seguem o padrão citado, uma interoperabilidade envolvendo diferentes produtos. A grande maioria dos fabricantes de equipamentos de redes tem um leque de produtos baseados no padrão *IEEE 802.11b*. Outro importante fator que tornou o padrão *IEEE 802.11b* o mais utilizado pelo mercado, é que ele pode ser considerado extremamente amigável. Todos os dispositivos que seguem esse padrão possuem preços acessíveis e são fáceis de serem instalados.

No cenário de redes, de uma forma geral, um dos principais requisitos é a segurança dos diversos sistemas e dispositivos envolvidos. Assim, faz-se necessária a garantia de que as informações que circulam estejam protegidas, contra leituras/alterações indevidas, duplicações ou eliminações feitas por pessoas não autorizadas. Em linhas gerais, tanto os usuários quanto os responsáveis pela operação, gerenciamento e controle das redes estão interessados em garantir a privacidade, a autenticidade e outros requisitos básicos que envolvem aspectos de segurança.

Em redes cabeadas diversos aspectos de segurança têm sido exaustivamente abordados nos últimos anos. Entretanto, no cenário das redes sem fio, apesar dos aspectos mencionados serem análogos àqueles das redes cabeadas, as ameaças e os riscos trazidos pelo uso do meio físico não confinado, inerente à tecnologia utilizada, são motivos de maiores preocupações. Este trabalho está inserido exatamente nesse contexto.

Este artigo está organizado da seguinte forma. Na próxima seção, é feita uma apresentação concisa do protocolo de segurança *WEP*, utilizado pelo padrão *IEEE 802.11b*. A abordagem busca salientar as principais rotinas e procedimentos propostos para o uso de criptografia e autenticação. Em seguida, com base em resultados da literatura, são apresentadas as principais falhas de segurança encontradas no protocolo em questão. Na Seção 4, são apresentados os principais resultados obtidos a partir de medidas de campo realizadas na cidade do Rio de Janeiro, com relação à utilização do protocolo *WEP* em *WLAN*s. A partir desses resultados e das vulnerabilidades observadas no protocolo *WEP*, são apresentadas, na Seção 5, diversas recomendações que visam estabelecer um mínimo de segurança, tanto para os usuários quanto para os administradores de redes sem fio que utilizam o padrão *IEEE 802.11*. Na Seção 6 são apresentadas as conclusões do artigo e algumas propostas, em relação às pesquisas que atualmente estão sendo desenvolvidas no sentido de proporcionar soluções alternativas ao uso do *WEP*.

2 Segurança no padrão *IEEE 802.11b*

O protocolo *WEP* tem como objetivo oferecer segurança aos usuários de redes locais sem fio (*WLAN*s) que utilizam o padrão *IEEE 802.11b*. Conforme será descrito mais adiante, o *WEP* envolve um algoritmo simétrico[2] de criptografia dos dados que trafegam no meio físico da *WLAN*. Desta forma, o *WEP* utiliza uma mesma chave para criptografar e descriptografar os pacotes. O protocolo foi construído originalmente para atender as seguintes necessidades:

[1] *http://www.weca.net/*. É uma associação de fabricantes de produtos de redes para WLAN.

[2] Algoritmos simétricos são aqueles que se utilizam da mesma chave para cifrar e decifrar a mensagem, ou quando uma chave é facilmente derivável da outra.

- ***Grande confiabilidade:*** A segurança é baseada na dificuldade de se descobrir a chave através do uso da força bruta.
- ***Autosincronização:*** O *WEP* se autosincroniza a cada mensagem. Essa propriedade é crítica num algoritmo de camada de enlace, onde o algoritmo de entrega de pacotes *best effort* é empregado e a taxa de perda de pacotes pode ser alta.
- ***Eficiência computacional:*** O *WEP* foi construído para funcionar tanto em hardware quanto em software.
- ***Exportabilidade:*** Ele foi desenvolvido para que os produtos contendo o *WEP* possam ser exportados dos EUA.
- ***Opcionalidade:*** O uso do *WEP* deve ser opcional.

O *WEP* utiliza o algoritmo *RC4*[3], [4], que foi criado por Ronald Rivest em 1987, e foi mantido em sigilo. Sete anos depois, um algoritmo foi enviado para uma lista de discussão da Internet, um algoritmo. Foi confirmado empiricamente que ele se tratava do *RC4*, e o algoritmo deixou de ser segredo industrial e tornou-se de domínio público. Assim como todo bom algoritmo criptográfico, o *RC4* é seguro independente de seu algoritmo ser público ou não. O algoritmo é de propriedade da RSA Security.

O diagrama da Figura 1 tenta mostrar como o algoritmo funciona. Dois processos são aplicados sobre o texto puro. Um deles é o processo de criptografia e o outro é um processo que visa proteger quanto a uma alteração não autorizada no texto durante a transmissão. No processo de criptografia, a chave secreta de 40 bits é concatenada ao *IV* (vetor de inicialização), que tem 24 bits, e esse 64 bits são usados para embaralhar a seqüência de números de 0 à 255 (veja a Seção 2.1), que são misturados, antes de serem enviados para a próxima etapa. É aconselhável que o *IV* varie a cada pacote ou mensagem enviada.

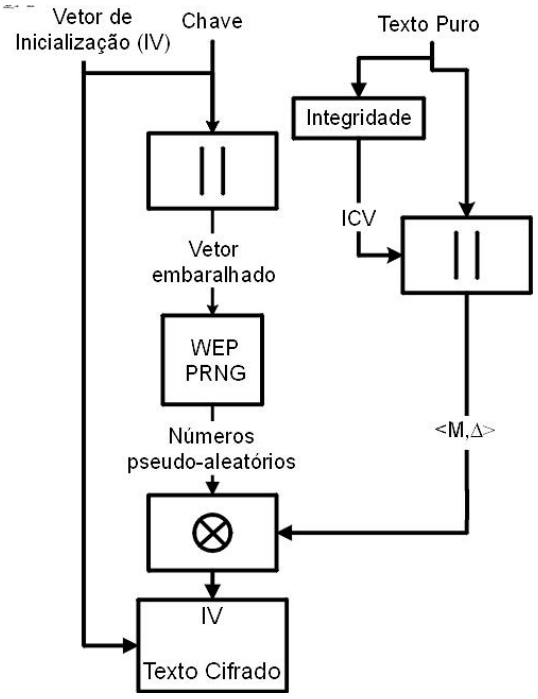

Figura 1: Esquema de criptografia do *WEP*

[3] *Ron's Code nº 4*

A próxima etapa é quando o vetor que foi misturado no processo anterior, é submetido ao algoritmo *PRGA* (veja a Seção 2.2). A saída deste algoritmo é utilizada para criptografar o texto puro, que já foi concatenado ao *ICV*, através de uma operação binária de `XOR`. O resultado da criptografia é exatamente do tamanho do texto puro somado ao tamanho do *ICV*.

O *ICV* (*integrity check value*) possui 4 bytes (32 bits) e é gerado pelo algoritmo chamado de *CRC32*. O *CRC32* é utilizado para proteger os dados contra uma modificação não autorizada, uma vez que o *ICV* é gerado pelo remetente utilizando-se do texto puro, e o destinatário pode gerar novamente o *ICV* a partir do texto puro que acabou de descriptografar e comparar com o *ICV* que foi gerado pelo remetente, se por qualquer motivo os dois *ICV*s não coincidirem, a mensagem pode ser descartada.

Observe ainda na Figura 1 que o *IV* é transmitido pelo canal inseguro sem nenhum tipo de proteção.

O *RC4* é uma maneira de se gerar bytes pseudo-aleatórios, a partir de uma chave de tamanho variável. O destinatário executará o *RC4* como o remetente, obtendo os mesmos bytes pseudo-aleatórios, podendo assim decifrar a mensagem.

A principal vantagem do *RC4* é que ele é um algoritmo de fluxo. O algoritmo de fluxo é chamado assim, pois cifra a mensagem pura bit a bit [5, pp. 95], permitindo que cada bit seja transmitido logo após ser cifrado.

2.1 A expansão da chave (*KSA*)

O *RC4* recebe uma chave *ch* de n_{ch} bits, onde $1 <= n_{ch} <= 2048$. Aqui se gera um vetor S de 256 bytes, a partir da chave: $S = (S_0, S_1, S_2, \ldots, S_{255})$

Para tanto, utiliza-se o seguinte algoritmo:

Para i de 0 a 255 faz-se

$S_i := i$

Seja o vetor de 256 bytes (2048 bits) $K = (k_0, k_1, \ldots, k_{255})$

Copia-se a chave ch para K bit a bit, repetindo-a quantas vezes forem necessárias para preencher K completamente. Por exemplo, se $n_{ch} = 100$ *copia-se a chave 20 vezes para K, e ainda se coloca os 48 primeiros bits de ch no fim de K para terminar de preenchê-lo.*

$j := 0$

Seja t um byte.

Para i de 0 *a* 255 *faz-se:*

$j := (j + S_i + k_i)\ mod\ 256$

$t := S_i$

$S_i := S_j$

$S_j := t$

Algoritmo retirado de [5, pp. 98-99].

Pode-se perceber que S é, de fato, uma permuta dos números de 0 a 255 determinada pela chave.

2.2 O algoritmo do *RC4* (*PRGA*)

Para gerar os bytes pseudo-aleatórios tem-se o seguinte algoritmo, onde S é o vetor gerado pelo *KSA*:

i:=0

j:=0

Seja t um byte.

Enquanto forem necessários bytes b aleatórios faz-se:

$i := (i + 1)\ mod\ 256$

$j := (j + S_i)\ mod\ 256$

$t := S_i$

$S_i := S_j$

$S_j := t$

$t := (S_i + S_j)\ mod\ 256$

$b := S_i$

O byte aleatório será o b

Algoritmo retirado de [5, pp. 99-100].

Note que o vetor S muda à medida que se vão gerando bytes aleatórios. Isto contribui para a força do algoritmo.

3 Falhas no *WEP*

As vulnerabilidades do padrão *IEEE 802.11b* começaram a ser notadas e divulgadas no fim de 2000. Jesse Walker, da Intel, foi pioneiro, ele escreveu em 2000 um relatório para o *IEEE* [6]. Em março de 2001, Arbaugh, Shankar e Wan escreveram um artigo intitulado *"Your 802.11 network has no clothes"* [7]. Depois desses, vários outros importantes artigos foram publicados divulgando falhas no protocolo de segurança desse padrão. Mas no segundo semestre do 2001, dois artigos, [8, 9], foram considerados os mais importantes, pois puseram por terra quaisquer esperanças do *WEP* ser considerado confiável quanto ao aspecto de segurança, da forma original como foi concebido.

O protocolo *WEP*, que é utilizado para a criptografia dos dados transmitidos, certificação da integridade dos mesmos e autenticação de um cliente autorizado para estabelecer conexões, foi duramente criticado em todos os artigos, no geral.

No final desse mesmo ano, partiu-se para um estudo cuidadoso desse protocolo, aprendendo sobre os algoritmos de embaralhamento da chave e de criptografia dos dados, respectivamente o *KSA* e o *PRNG*[4] [9].

O *RC4*, especificamente, foi duramente combatido nos artigos [8, 9], principalmente no segundo, sendo descrito como um algoritmo frágil. No final do mesmo ano, o seu criador veio à público defender o algoritmo. Em [10], o Prof. Rivest chama a atenção para o fato de que o *WEP* é um dos muitos protocolos baseados no algoritmo *RC4*. Os ataques implementados em [11] são especificamente voltados para o *WEP*, e não necessariamente afeta outros protocolos baseados no *RC4*.

Esse trabalho resume, a seguir, a resposta do Prof. Rivest, e depois os trabalhos produzidos pelo grupo de pesquisadores da Universidade da Califórnia (Berkeley)[5], [8], e pelo artigo polêmico de Fluhrer, Mantin e Shamir, [9].

3.1 Defesa de Rivest

O artigo do Prof. Rivest, [10] diz que em protocolos como o *WEP*, geralmente é necessário criar uma chave, a ser utilizada no algoritmo *RC4*, distinta para cada mensagem ou pacote criptografado. Todas as chaves são baseadas numa chave-base fornecida pelo administrador da rede. O método empregado pelo *WEP* para conseguir essas chaves é concatenar à chave-base um contador, que no caso específico do *WEP*, é o *IV*. No entanto, o autor e a RSA Security, empresa a qual o autor dirige, desaconselham esse método de derivação de chaves. Em substituição, eles recomendam que a chave-base e o contador passem por uma função hash. Eles sugerem também que se dispense os 256 primeiros bytes calculados pelo gerador pseudo-aleatório, antes de se começar a criptografia[6].

[4] *Key Scheduling Algorithm e Pseudo-Random Number Generation*

[5] *http://www.isaac.cs.berkeley.edu*

[6] Aliás, essa é a primeira solução a ser tomada, aclamada pela unanimidade das bibliografias especializadas, apesar de alguns autores ressalvarem que a medida pode ser custosa ou impossível para algumas implementações.

O Prof. Rivest ainda diz no seu artigo que o *RC4* é utilizado para proteger o tráfego na internet através do protocolo *SSL* (*Secure Sockets Layer*). Deste modo, resume ele, o *RC4* passa a ser o algoritmo de criptografia mais utilizado no mundo, por causa da grande utilização do protocolo *HTTPS*.

O *SSL* gera as chaves de criptografia que são utilizadas no *RC4* através de funções hash (usando tanto o *MD5*, como o *SHA1*), logo, diferentes sessões possuem chaves que não podem ser relacionadas. O *SSL* não troca de chave-base para o *RC4* para cada pacote, mas utiliza o estado final do *RC4* ao fim da transmissão de um pacote para criptografar o próximo pacote, sem reinicializar as variáveis do algoritmo.

Finalizando, o Prof. Rivest se diz otimista que novos estudos criarão funções hash que consumam quase tão pouco recurso de processamento e memória quanto a simples concatenação da chave-base e o contador.

3.2 Ferramentas de ataque

A partir de [8, 9] e de um terceiro artigo, [11], lançado um pouco mais tarde, foram criados os primeiros softwares que são capazes de quebrar o protocolo de segurança. Como, por exemplo, o AirSnort [12], o mais utilizado. Esse programa é capaz de "escutar"a comunicação entre dispositivos móveis[7] e um ponto de acesso[8], e em algum tempo descobrir a chave que serve tanto para o algoritmo de criptografia, como para o algoritmo de autenticação.

O mais incrível é que, segundo [11], o AirSnort pode quebrar o *WEP*, com chave de 64 bits, em 15 minutos, e quando a chave usada é de tamanho de 128 bits, o tempo necessário para quebrar o algoritmo e descobrir a chave é de, aproximadamente, 40 minutos.

No *WEP*, a mesma chave que é utilizada para criptografar e descriptografar é também utilizada para autenticar uma estação. Ter a mesma chave para criptografar e autenticar é considerado um risco de segurança. Se o intruso conseguir, de algum modo, quebrar o controle de autenticação e descobrir a sua chave, automaticamente estará apto a "ouvir"tudo o que se é transmitido naquela rede.

3.3 Reutilização do vetor de inicialização

O vetor de inicialização no *WEP* tem 24 bits, e junto com a chave, é responsável por gerar a cadeia pseudo-aleatória que criptografa o texto puro. O primeiro problema no *WEP* é justamente o tamanho desse *IV* que é muito pequeno. No caso extremo, esse *IV* é alterado a cada pacote enviado, começando no zero e indo até o valor máximo $2^{24} - 1$. Podemos calcular quanto tempo vai demorar para esse *IV* voltar a assumir o valor 0 novamente: imagine uma conexão cuja banda seja de 5 $Mbits/s$ (o máximo no *IEEE 802.11b* é 11 $Mbits/s$ [3]), com o tamanho médio dos pacotes de 1500 bytes.

$$(\frac{5\ Mbits/s}{8\ bits})/1500\ bytes \cong 416\ pac/s$$
$$\frac{2^{24}\ pac}{416\ pac/s} \cong 40.329s\ ou\ 11h12m$$

Em suma, no caso mais extremo, numa conexão de 5 Mbits/seg, o *IV* voltará a assumir o mesmo valor em menos de meio dia. Se a implementação assumir que o *IV* terá valores aleatórios teremos a repetição de um *IV* em menos tempo. E é a partir dessa repetição de *IV* que o *WEP* pode ser quebrado. A chave K é fixa, e foi configurada nos clientes que estão se comunicando, logo o par $< K, IV >$ repetir-se-á sempre que o *IV* se repetir. E sempre que eles se repetirem, gerarão a mesma string pseudo-aleatória, que iremos referenciar como *RC4(K,IV)*, e é a isso que chamamos de colisão de chaves.

Imagine dois textos legíveis distintos P_1 e P_2, que são criptografados através da mesma cadeia pseudo-aleatória *RC4(K,IV)* em C_1 e C_2.

$$C_1 = P_1 \otimes RC4(K, IV)$$
$$C_2 = P_2 \otimes RC4(K, IV)$$

[7] No contexto desse artigo, dispositivo móvel é qualquer aparelho portátil com processamento próprio que é capaz de transmitir dados no padrão *IEEE 802.11b*

[8] É a infra-estrutura da rede WLAN, normalmente formada por uma antena omni-direcional e uma porta de acesso a uma rede cabeada. Uma rede WLAN pode ter mais de um ponto de acesso.

$$C_1 \otimes C_2 = (P_1 \otimes RC4(K, IV)) \otimes (P_2 \otimes RC4(K, IV))$$
$$C_1 \otimes C_2 = (P_1 \otimes P_2) \otimes (RC4(K, IV) \otimes RC4(K, IV))$$
$$C_1 \otimes C_2 = P_1 \otimes P_2$$

Pelas propriedades do `XOR` (ou-exclusivo), podemos dizer que de posse de dois textos criptografados e um texto legível é possível descobrir o outro texto legível, pois:

$$C_1 \otimes C_2 \otimes P_2 = P_1 \otimes P_2 \otimes P_2 = P_1$$

Observa-se que a padronização do *WEP* não estipula a forma como se varia o *IV*, deixando isso aos cuidados dos implementadores. Algumas implementações não variam o *IV*, ou faz com que a variação seja aleatória.

3.4 Gerenciamento de chaves

O padrão *IEEE 802.11b* não especifica como deve ser a distribuição das chaves. Ele é baseado num mecanismo externo de distribuição global da chave em um vetor de 4 chaves. Cada mensagem contém um campo de identificação de chave para especificar o índice do vetor de chaves que está sendo usada. Na prática, a maioria das instalações utiliza a mesma chave para todos os dispositivos.

Isso traz problemas profundos à segurança dessas instalações, uma vez que a chave é compartilhada com vários usuários, fica muito complicado manter o segredo. Alguns administradores de rede tentam amenizar o problema não revelando a chave secreta ao usuário final, configurando, eles mesmos, os dispositivos. Mas isso não traz a solução, pois as chaves continuam guardadas nos dispositivos remotos.

A reutilização de uma única chave por vários usuários também aumenta as chances da colisão[9] do *IV*. A chance de uma colisão aleatória aumenta proporcionalmente ao número de usuários.

Uma vez que a troca de chaves requer que cada usuário reconfigure o seu dispositivo, as atualizações dos drivers controladores dos cartões de rede (*NIC*) serão cada vez menos freqüentes. Na prática, a troca demorará meses ou anos para acontecer, dando mais tempo para os intrusos analisarem o tráfego.

3.5 O *CRC32* é *linear*

Outra grande fraqueza do *WEP* é o seu algoritmo de garantia da integridade (*ICV - integrity check value*), que é o *CRC32*.

O *CRC32* é linear, isto é, $c(x \otimes y) = c(x) \otimes c(y)$ para qualquer valor de x e y. Essa propriedade serve para qualquer tipo de algoritmo *CRC*.

Uma conseqüência dessa propriedade é a possibilidade de se fazer modificações controladas no pacote, sem que sejam detectadas por qualquer um dos dispositivos transmissores ou receptores. Veremos que é possível alterar o conteúdo dos pacotes apenas com o conhecimento da string de valores pseudo-aleatórios.

Vamos lembrar como é formado o texto criptografado C, que corresponde ao texto legível P. O *RC4* gera uma cadeia de bits aleatórios que são formados dependentemente da chave e do vetor de inicialização, essa cadeia é operada, através de um *ou-exclusivo* com outra cadeia de bits formada pelo texto puro concatenado com o *ICV*. Assim como vemos abaixo:

$C = RC4(IV, K) \otimes < P, c(P) >$

Vamos imaginar um outro texto criptografado, C', que seja a imagem da criptografia de um outro texto legível, P', onde $P' = P \otimes D$, onde D é a alteração controlada que se deseja fazer. Veja só o desenvolvimento da fórmula a seguir:

$C' = RC4(IV, K) \otimes < P', c(P') >$
$C' = RC4(IV, K) \otimes < P \otimes D, c(P \otimes D) >$
$C' = RC4(IV, K) \otimes < P \otimes D, c(P) \otimes c(D) >$
$C' = RC4(IV, K) \otimes < P, c(P) > \otimes < D, c(D) >$
$C' = C \otimes < D, c(D) >$

[9] Colisão, nesse contexto, significa a captura de dois pacotes que utilizaram a mesma string pseudo-aleatória para a criptografia.

Ou seja, pode-se interceptar o pacote, fazer a alteração, corrigir o *ICV*, e a alteração não será detectada, pois o sistema de manutenção de integridade foi perfeitamente burlado.

3.6 Correlação dos bytes da chave

Em [9], Scott Fluhrer, Itsik Mantin e Adi Shamir mostram o mais frágil dos problemas do *WEP*, a correlação dos bytes que saem do algoritmo *RC4* e a chave. Eles enfatizam o fato de que o *RC4* ser um algoritmo de criptografia de fluxo, faz com que o byte a ser utilizado na operação de `XOR` somente dependa das iterações anteriores, o resultado conquistado daqui por diante não tem mais efeito sobre esse byte.

Observando a segunda parte do *RC4*, o *PRGA* (veja a seção 2.2), é possível ver que o primeiro byte gerado é formado pelo byte $S_{S_1+S_{S_1}}$, onde S é o vetor de bytes numerados de 0 até 255 que foi permutado pela primeira parte do algoritmo *RC4*, o *KSA*.

4 *Wardriving*

Em setembro de 2001 foi feita nos Estados Unidos uma experiência de *wardriving*. No contexto do estudo de redes sem fio, o termo *wardriving* determina uma experiência em que o pesquisador, munido de um dispositivo de rede sem fio, softwares de pesquisa e uma antena, anda por um percurso aleatório ou pré-estabelecido, fazendo observações sobre a segurança das redes sem fio instaladas.

A experiência de Ellison *et al* [13] ficou conhecida como Experiência de Manhattan. A equipe utilizou o AirSnort e o NetStumbler. O NetStumbler [14] é um software que registra todos os pontos de acesso encontrados na área de cobertura do dispositivo ao qual está associado, e marca aqueles que têm o protocolo *WEP* habilitado. Esse software foi desenvolvido a partir da constatação de que muitas das redes instaladas não tinham o protocolo *WEP* instalado. Ele é utilizado principalmente por pessoas que querem encontrar pontos de acesso desprotegidos para poderem ter acesso às redes, na maioria das vezes para poder ter acesso livre à Internet.

Daí surgiu o termo *warchalking* que é o ato de andar com um dispositivo de rede sem fio e o Netstumber, ou outro software que tenha o mesmo fim, e marcar com um giz (daí o nome) os locais na cidade, onde qualquer um pode ter acesso à um ponto da rede. Aqui no Brasil já há vários grupos que se unem para mapear todas as grandes metrópoles.

Durante o período desse trabalho, também foi feito um *wardriving* na cidade do Rio de Janeiro, onde está a UFRJ. Medidas análogas vêm sendo colhidas em várias outras cidades do Brasil e do mundo.

4.1 Requisitos de hardware e software

Foi usado um computador notebook, com dual boot: Windows XP e Red Hat Linux 7.0. Enquanto o NetStumbler é um software para plataforma Windows, o AirSnort foi desenvolvido para plataforma Linux. Há também programas similares ao NetStumbler que roda sobre o Linux, que se fosse utilizado, evitaria assim a necessidade do uso de dois sistemas operacionais.

Nesse computador foi conectado um cartão *PCMCIA* de rede sem fio, padrão *IEEE 802.11b*, da ORiNOCO[10], e a ele foi ligada uma antena omni-direcional de 15 dBi para melhorar a recepção do sinal.

Existem alguns procedimentos que se deve adotar na hora de compilar o AirSnort, mas todos eles estão bem descritos na sua *Home Page* [12].

4.2 Pontos de coleta

A colheita dos dados foi executada em dois dias não consecutivos do mês de julho de 2002, e resumidamente atuamos em dois bairros da cidade do Rio de Janeiro: Centro e Barra da Tijuca. Esses dois bairros foram escolhidos por serem os que concentram as maiores e mais importantes empresas da iniciativa privada nesta cidade.

[10] *http://www.orinocowireless.com*

4.3 Resultados

Do ponto de vista de segurança, foi encontrado um cenário bastante insatisfatório. Poucas redes têm o protocolo *WEP* habilitado, como pode se ver no gráfico da Figura 2.

Não é interesse desse trabalho citar o nome das empresas cujas redes são convidativas, mas é alarmante saber que entre essas empresas há uma grande empresa do mercado de telecomunicação, que entre outros serviços que comercializa, está o de garantir segurança nas comunicações de seus clientes.

O estacionamento de um grande shopping na Barra da Tijuca foi visitado e notou-se que em vários pontos é possível ter acesso livre à internet. Em um desses pontos, pode-se sentar em um banco confortável e ligar o computador, e mesmo que não se queira entrar na rede na empresa, o computador acaba sendo convidado pelo servidor do protocolo *DHCP* (*Dynamic Host Configuration Protocol*) dessa rede. Como se fosse um *hot spot*. Não pode-se nem classificar isso como uma invasão na rede, uma vez que o dispositivo móvel foi convidado a associar-se à rede quando recebeu os números *IP*s sem requisitar[11].

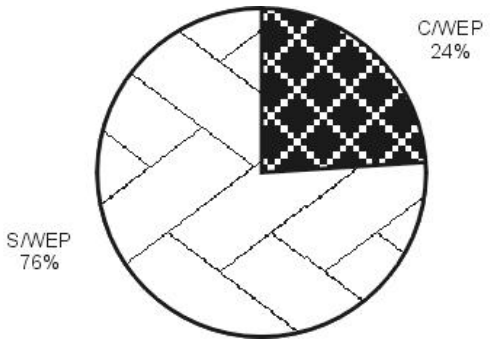

Figura 2: Relação das redes sem fio que possuem o *WEP* habilitado e desabilitado

Das 63 redes mapeadas, somente 15 estavam com o protocolo *WEP* habilitado, ou 24%. Esse número é muito preocupante, visto que o Rio de Janeiro, uma das grandes capitais do país, deveria ser um dos mercados consumidores mais bem informados. Nota-se uma pressa muito grande em adquirir a tecnologia, sem se preocupar com a sua viabilidade.

5 Recomendações

Existem três caminhos que podemos seguir. O primeiro, e mais óbvio, é não utilizar redes sem fio até que o padrão esteja totalmente confiável. Esse é o caso do governo norte-americano que não utiliza essa tecnologia em suas agências militares e aconselha veementemente o não uso dela nas agências civis federais. O segundo é para quem não confia que se possa fazer um protocolo de segurança em camada de enlace. Esse usuário pode esquecer o *WEP* e investir em outros produtos já existentes no mercado.

Sugere-se a instalação de *VPN*s (*Virtual Private Network*), [15], rodando sobre o protocolo de *IPSec*, [16]. Sugere-se também o uso de um algoritmo forte de autenticação, como exemplo os baseados no protocolo *RADIUS* [17]. Com essas soluções, o sistema é capaz de prover autenticação, privacidade, integridade, irrevogabilidade e controle de acesso. No entanto, esses produtos trazem um custo financeiro adicional, e tornam a conexão mais lenta, além de que existem problemas de configuração e compatibilidade entre os diferentes fabricantes e plataformas.

Acredita-se que se o usuário confiar que o *WEP* pode evoluir, se tornando um protocolo seguro, ele será recompensado, pois estará economizando dinheiro e recursos de processamento. Esse artigo mostrará adiante como a Indústria e a Acad2emia vêm trabalhando para resolver as falhas do *WEP*.

[11] Formalmente devemos dizer que o servidor de *DHCP* enviou os dados depois que o dispositivo móvel se anunciou.

Antes, acredita-se que existem atitudes que o administrador de uma rede pode e deve tomar visando aumentar a segurança da sua rede, quando ela tem um dispositivo sem fio. Esse trabalho lista as mais importantes aqui, mas somada a elas devem ser seguidas todas aquelas que provêm segurança em um sistema de informação genérico. Recomenda-se uma olhada na vasta literatura.

- Habilite o *WEP*. Como falamos, o *WEP* não é seguro, mas é fato que ele dificulta o acesso de curiosos, e evita que alguém acesse a sua rede por acaso.
- Altere o *SSID* (*Service Set Identifier*) do seu produto quando ele vier de loja. *SSID* é um identificador que associa os dispositivos que podem acessar o ponto de acesso da sua rede. O valor do *SSID* de cada fabricante é conhecido, ou é facilmente encontrado na Internet. Com o *SSID* diferente daquele que vem da fábrica você estará dificultando o trabalho do invasor.
- Não troque o *SSID* para algo que possa ser facilmente derivado do nome da sua empresa, do seu departamento, ou de algum produto. Também não utilize nomes de rua, bairro ou cidade onde a sua rede está instalada. Esses nomes são os primeiros a serem tentados pelos invasores.
- Desabilite a opção *broadcast SSID*. Se essa opção estiver habilitada isso significa que o ponto de acesso da sua rede aceitará qualquer dispositivo com qualquer *SSID*, igual ao do ponto de acesso ou não. Isso torna a sua rede mais frágil que se estivesse com o *SSID* de fábrica.
- Altere a senha inicial do seu ponto de acesso e do seu roteador. Assim como o *SSID*, as senhas que os roteadores e pontos de acesso trazem de fábrica são conhecidos pelos invasores.
- Peça ao administrador que, de posse do NetStumbler ou de outro software com o mesmo fim, faça periodicamente um *wardriving* ao redor das suas instalações, para ver se não há algum novo ponto de acesso instalado na sua companhia que não tenha sido do conhecimento da administração da rede. Há casos em que um departamento da empresa, independentemente, instala pontos de acesso sem informar a uma coordenação central. Isto é agravado pelo fácil processo de instalação de um ponto de acesso, e o seu relativo baixo custo.
- Se o seu ponto de acesso tiver o recurso da criação de uma tabela de endereço *MAC* (*Media Access Control*), utilize-o. Os novos dispositivos de rede não permitem que os seus endereços *MAC* sejam alterados, então essa medida pode ser bem eficaz para evitar uma invasão. Desta forma você tem como controlar através do endereço *MAC* quem pode se autenticar no seu ponto de acesso.
- Use um algoritmo de autenticação seguro, como por exemplo, o *RADIUS* [17], para que um cliente possa se autenticar antes de acessar a sua rede.
- Desabilite o *DHCP* para os dispositivos que acessam o seu ponto de acesso. *DHCP* é um protocolo que fornece informações sobre os números de configuração do protocolo *IP* aos dispositivos que se autenticam na rede. Não é impossível descobrir a configuração de números *IP* da sua rede, mas, com certeza, você vai estar criando uma dificuldade além.
- Compre pontos de acessos ou dispositivos de acesso que permitam suporte para a implementação do *WEP* que funciona com chave de tamanho 128 bits. A chave de 128 bits também se mostrou que não deixa o protocolo seguro, mas dificulta um pouco mais a sua quebra.

5.1 O que vem sendo feito no TGi

TGi é o grupo de trabalho que visa projetar os mecanismos de segurança do padrão *IEEE 802.11*. É de responsabilidade deles a viabilização do padrão no aspecto de segurança. E muitos passos importantes vêm sendo dados na direção de se dar credibilidade ao protocolo *WEP* e na direção de se criar um outro protocolo mais robusto que substituirá o *WEP*.

O grupo encontra alguns problemas na hora de dar melhores soluções para o padrão:

- ***Baixo poder computacional dos chips :*** Os algoritmos devem ser leves o suficiente para que possam ser executados nos cartões existentes hoje.

- ***Manter compatibilidade :*** Deve-se manter a compatibilidade com o padrão que a *WECA* chamou de *Wi-Fi*.

A melhor proposta para trazer segurança ao *WEP* de aplicação imediata, ou seja, que pode ser implementada nos equipamentos já instalados, desde que eles tenham suporte à atualização de software ou firmware, é o *TKIP* (*Temporal Key Integrity Protocol*)[12]. O algoritmo de escalonamento de chaves do *TKIP* surgiu a partir da idéia proposta ao *IEEE por Russ Housley* (RSA Security) e Doug Whiting (Hifn), chamada de *Temporal Key Hash*[13], [18]. *TKH* é uma função hash geradora de chaves para o *WEP*. A utilização de uma função hash para derivar uma outra chave a partir da chave-base foi sugestão de Ron Rivest em [10], que citou como exemplo o *MD5*. Entretanto os autores do *TKH* preferiram não utilizar o *MD5* ou *SHA1*, por serem muito custosos. No lugar disso eles propuseram um algoritmo muito mais simples e que exige menos processamento. Segundo [19] submeter a chave à função hash resolve parte do problema. O resultado da função hash deve ser combinado ao resultado da função de integridade para prevenir a alteração e o reenvio de mensagens, e o gerenciamento de chaves ainda precisa ser implementada. Em 2002, Niels Ferguson juntou-se à Housley e Whiting, e propuseram um alternativa ao *TKH*, o *ATKH*(*Alternate Temporal Key Hash*), [20], que já foi aceito pelo TGi.

No *TKIP*, também conhecido como *WPA v1* (*WiFi Protected Access, version one*), o dispositivo começa com uma chave-base secreta de 128 bits, chamada de *TK* (*Temporal Key*), então ela é combinada com o *TA* (*Transmitter Address*), o endereço *MAC* do transmissor, criando a chave chamada de *TTAK* (*Temporal and Transmitter Address Key*), ou a "Chave da Fase 1". A *TTAK* é então combinada com o *IV* para criar as chaves que variam a cada pacote, chamadas de *RC4KEY*. Cada chave é utilizada pelo *RC4* para criptografar somente um pacote.

O *TKIP* faz com que cada estação da mesma rede utilize uma chave diferente para se comunicar com o ponto de acesso. O problema da colisão de chaves (veja a seção 3.3) do *RC4* é resolvido com a substituição da *TK* antes que o *IV* assuma novamente um valor que já assumiu, ou seja a cada vez que o *IV* assuma o seu valor inicial, o *TK* deve assumir um valor distinto. A forma como é gerenciada essa troca de *TK*s não foi padronizada. Cabe ao próprio TGi o processo de validação dos novos protocolos, o que, acredita-se, não ser o sistema mais confiável.

No que tange a autenticação, esse grupo vem trabalhando num novo algoritmo de autenticação chamado *ULA* (*Upper Layer Athentication*). O *ULA* é baseado no processo de autenticação do padrão *IEEE 802.1X*[14]. O *IEEE 802.1X* utiliza um protocolo de autenticação do tipo *EAP* (*Extensible Authentication Protocol*)[21].

Agere e Cisco anunciaram que serão os primeiros fabricantes a lançar linha de produtos que já contêm o *TKIP*. Está previsto[15] que essa linha de produtos estará no varejo no início de 2003.

5.2 Um protocolo robusto

Mas o TGi também vem trabalhando no intuito de fazer um novo padrão de segurança que demandará a troca dos chips existentes nos cartões de rede, o *WPA v2* (*WiFi Protected Access, version two*). A idéia é que o novo padrão ainda mantenha compatibilidade com o padrão atual, para que os usuários possam fazer a mudança de seus equipamentos gradualmente. Em [22] podemos ler que o grupo vem desenvolvendo o que chamaram de *ESN* (*Enhanced Security Network*) que será o embrião do *WPA v2*, e trará novos mecanismos mais fortes de criptografia e autenticação.

Para a criptografia está sendo sugerido o uso do algoritmo *AES* (*Advanced Encryption Standard*), que é o novo padrão norte-americano, baseado no algoritmo de Rijndael[23]. O TGi ainda não especificou

[12] Em novembro de 2001, o TGi aprovou uma moção para substituir todas as referências ao *WEP2* por *TKIP*

[13] A RSA implementou o *TKH* com o nome de *Fast-Packet Keying*.

[14] *IEEE 802.1X* é um padrão voltado para a autenticação, e que não se preocupa com a privacidade, integridade e a irrevogabilidade.

[15] Segundo Donald Eastlake III, da Motorola, publicou no *The IEEE Boston Section Techsite*

qual tipo de protocolo *EAP* deverá ser utilizado junto com o *AES*. Isso certamente será fonte de problemas no futuro. Vêm-se falando na utilização dos protocolos de criptografia chamados *CCMP* (*Counter Mode CBC-MAC Protocol*) e *WRAP* (*Wireless Robust Authenticated Protocol*), ambos baseados no *AES*, entretanto os planos para o publicação desses protocolos são, inicialmente, para 2004.

5.3 Trabalhos em paralelo

O TGi do *IEEE 802.11* não são os únicos a trabalhar em prol da melhoria de segurança do padrão *IEEE 802.11*. Espalhados pelo mundo existem vários grupos voltados para esse objetivo. Principalmente dentro dos laboratórios das indústrias de dispositivos de rede e nas universidades.

O Laboratório de Redes de Alta Velocidade da COPPE/UFRJ possui um grupo de trabalho voltado para a pesquisa em redes sem fio, o GARF[16]. Desse grupo sairão trabalhos úteis para melhorar a segurança em redes sem fio.

- ***Escolha aleatória do algoritmo de criptografia :*** Esse trabalho tentará, usando ferramentas matemáticas e computacionais, provar que pode-se implementar um protocolo de segurança que escolherá de forma "aleatória"um algoritmo de criptografia de fluxo, entre eles o próprio *RC4*, tornando assim mais difícil a criptoanálise.

- ***Wstrike :*** Esse é um trabalho que já está sendo desenvolvido pelo Demetrio Carrión e visa a criação e a disponibilização de uma ferramenta de instalação dos aplicativos de segurança num ponto de acesso de uso genérico[17]. O processo de autenticação e autorização para associação de uma estação a um ponto de acesso passa por duas partes: StrikeIN e *VPN*. O *StrikeIN* inicialmente tem somente a função de filtrar os pacotes transmitidos da estação para o ponto de acesso, permitindo que somente as portas destinadas à requisição de *IP* (*DHCP*) e a resposta de requisições de páginas WEB (*HTTPS*) possam receber os pacotes. Quando a estação acessa a página WEB segura, via *SSL*, certificados digitais são trocados de forma a se fazer uma autenticação mútua entre a estação e o ponto de acesso. Após a validação do usuário, o ponto de acesso modifica dinamicamente suas regras de firewall permitindo que a estação, com o *IP* fornecido, posa acessar as portas destinadas ao estabelecimento de um *VPN*.

6 Conclusão

Segurança em redes sem fio ainda é um assunto muito pouco explorado pelas pessoas responsáveis pelo dimensionamento e instalação de redes. Esse trabalho mostrou isso na pesquisa que revelou que poucas redes preocupam-se com a implementação do WEP. Apesar do que foi visto a respeito do *WEP* e suas falhas, o uso de nenhum mecanismo de segurança é preocupante. Outro fato preocupante é a falsa segurança que o *WEP* pode prover. Os usuários que acham que estão seguros, sem realmente estarem, são mais vulneráveis do que aqueles que sabem que estão totalmente desprotegidos. Como conseqüência imediata, os usuários ainda sofrem muitos prejuízos com perdas, adulterações e roubos dos seus dados.

Com uma ferramenta como o AirSnort, e sem necessitar de muito conhecimento na área, é possível penetrar uma *WLAN* que utilize somente o *WEP* como barreira. Essa ferramenta é gratuita e de relativa fácil intalação.

Foi visto que existem pequenos cuidados que, se tomados, podem dificultar o acontecimento de invasões e acidentes. Além disso, foi mostrado que o TGi vem pensando em soluções de médio e longo prazo para solução das falhas levantadas nesse artigo. O *WEP* mostrou-se frágil demais para aplicações comerciais. O *WPA v1* também tem as suas vulnerabilidades, mas a melhoria na segurança foi notada. Espera-se muito do *WPA v2*, principalmente que o seu custo final não invibialize o seu uso.

Devemos ressaltar que a ausência do *WEP* numa rede é um fato, que, se observado isoladamente, não pode decretar que uma rede é insegura. O *WEP* provê autenticação e criptografia numa camada baixa

[16] *http://www.garf.coppe.ufrj.br*. Grupo de Atuação em Redes sem Fio.
[17] Está sendo utilizado um PC com o S.O. OpenBSD

de rede[18], porém existem outros métodos de garantir segurança em camadas mais altas, como exemplo, as *VPN*s. Um dos grandes objetivos deste trabalho é alertar que pior do que estar desprotegido é ter a falsa impressão de estar protegido.

WPA v1 não será a solução definitiva e o *WPA v2* ainda demorará muito para ser visto no mercado, por demandar recompra de equipamentos e ainda não dar total garantia da compatibilização com as versões anteriores.

Na continuação desse trabalho pretende-se achar e testar ferramentas matemáticas e computacionais que possam medir desempenho e confiabilidade dos algoritmos de criptografia de fluxo e dos algoritmos de autenticação.

7 Agradecimentos

Agradecemos à Academia Brasileira de Ciências pelo apoio ao nosso trabalho.

Agradecemos ao BNDES, na pessoa do Sr. Oliveira, e a Cifmoney, na pessoa do Sr. Vitorio Mele, que nos permitiu ter acesso a dois pontos priviligiados no centro da cidade.

Agradecemos especialmente a ajuda dos colegas Alexandre Pinaffi Andrucioli e Luís Rodrigo de Oliveira Gonçalves, que durante o processo nos deram suporte na configuração dos equipamentos e softwares.

Referências

[1] J. de Vriendt, P. Lainé, C. Lerouge, and X. Xiaofeng, "Mobile network evolution: A revolution on the move. Broadband access series," *IEEE Communications Magazine*, abril 2002.

[2] R. Mesquita, "32 milhões de celulares no país, diz Anatel." Info on-line, São Paulo: Editora Abril, 21 novembro 2002. Disponível em: http://www2.uol.com.br/info/aberto/infonews/112002/21112002-19.shl. Acesso em: 23 dez. 2002.

[3] IEEE Std. 802.11b, *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications*. IEEE, 1999.

[4] R. L. Rivest, "The RC4 encryption algorithm," tech. rep., RSA Data Security, Inc., março 1992.

[5] D. B. de Carvalho, *Segurança de Dados com Criptografia: Métodos e Algoritmos*. Rio de Janeiro: Book Express, segunda ed., 2000. 252p.

[6] J. Walker, "Unsafe at any key size: An analysis of the WEP encapsulation," tech. rep., IEEE Standards 802.11 Committee, março 2000. Technical Report 03628E.

[7] W. A. Arbaugh, N. Shankar, and Y. C. Justin Wan, "Your 802.11 wireless network has no clothes," in *Proceedings of the First IEEE International Conference on Wireless LANs and Home Networks*, pp. 131–144, dezembro 2001.

[8] N. Borisov, I. Goldberg, and D. Wagner, "Intercepting mobile communications: The insecurity of 802.11," in *Proceedings of The Seventh Annual International Conference on Mobile Computing and Networking*, julho 2001.

[9] S. Fluhrer, I. Mantin, and A. Shamir, "Weakness in the key scheduling algorithm of RC4," in *Eighth Annual Workshop on Selected Areas in Cryptography*, Agosto 2001.

[10] R. Rivest, "RSA security response to weakness in Key Scheduling Algorithm of RC4." http://www.rsasecurity.com/. Acesso em: 13 dez. 2001, 2001.

[18] No caso do padrão *IEEE 802.11b*, o *WEP* fica numa camada que, de acordo com o modelo OSI, da ISO, se situa dentro da camada de enlace.

[11] A. Stubblefield, J. Ioannidis, and A. Rubin, "Using the Fluhrer, Mantin, and Shamir attack to break WEP," tech. rep., ATT&T, agosto 2002. TD-4ZCPZZ.

[12] SOURCEFORGE.NET, "AirSnort homepage." http://airsnort.shmoo.com/. Acesso em: 29 out. 2002, 2001.

[13] C. Ellison, "Exploiting and protecting 802.11b wireless networks," *ExtremeTech*, setembro 2001.

[14] W. Slavin, "Net Stumbler Dot Com." http://www.netstumbler.com Acesso em: 30 out. 2002, 2001.

[15] D. Kosiur, *Building and Managing Virtual Private Networks*. Wiley, 1st ed., 1998.

[16] R. Thayer, N. Doraswamy, and R. Glenn, "IP Security document roadmap," *RFC 2411*, 1998.

[17] C. Rigney, A. Rubens, W. Simpson, and S. Willens, "Remote Authentication Dial In User Service," *RFC 2138*, 1997.

[18] R. Housley and D. Whiting, "Temporal Key Hash," tech. rep., IEEE 802.11 WG, TGi, 2001. IEEE 802.11-01/550r3.

[19] L. Phifer, "Better than WEP." http://www.isp-planet.com/fixed_wireless/technology/2002/better_than_wep.html. Acesso em: 27 dez. 2002, 2002.

[20] R. Housley, D. Whiting, and N. Ferguson, "Alternate Temporal Key Hash," tech. rep., IEEE 802.11 WG, TGi, 2002. IEEE 802.11-02/282r2.

[21] L. Blunk and J. Vollbrecht, "PPP Extensible Authentication protocol (EAP)," *RFC 22284*, março 1998.

[22] M. Casole, "WLAN security - Status, problems and perspective," in *Proceedings of European Wireless 2002*, fevereiro 2002.

[23] J. Daemen and V. Rumen, *The Design of Rijndael : AES - The Advanced Encryption Standard*. Nova Iorque: Springer, 2002.

[24] T. Dismukes, "Wireless security blackpaper." http://www.arstechnica.com/. Acesso em: 01 nov. 2002, 2002.

[25] G. R. Mateus and A. A. F. Loureiro, *Introdução à Computação Móvel*. Rio de Janeiro: DCC/IM, COPPE/Sistemas ,NCE/UFRJ, 1998.

[26] G. Meredith, "Securing the Wireless LAN," *Packet magazine*, vol. 13, no. 3, pp. 74–77, 2001.

[27] L. D. Paulson, "Exploring the Wireless LANscape," *Computer Magazine*, outubro 2000.

[28] B. Schneier, *Applied Cryptography*. Nova Iorque: John Wiley & Sons, Inc., 1996.

[29] C. E. Shannon, "A mathematical theory of communication," *The Bell System Technical Journal*, vol. 27, pp. 379–423, julho 1948.

[30] W. Stallings, *Cryptography and Network Security*. Nova Jersey: Prantice-Hall, segunda ed., 1998. 569 p.

[31] A. S. Tanenbaum, *Redes de Computadores*. Rio de Janeiro: Campus, 1944.

[32] R. Terada, *Segurança de Dados: Criptografia em Redes de Computadores*. São Paulo: Edgard Bluscher, 2000.