Pentest para Quebra de Criptografia *Wireless*

Adrielle T. Q. Rocha, Bruno N. L. Costa, Kessius V. L. Giuzeppe, Henrique P. Martins

Faculdade de Tecnologia de Bauru – Redes de Computadores (FATEC-Bauru)
CEP 17.015-171 – Bauru, SP – Brasil

atqrocha@gmail.com, bru.87@hotmail.com, kevinicius@gmail.com, henrique.martins01@fatec.sp.gov.br

Abstract. *This paper aims to describe some types of encryption of wireless networks as well as the vulnerabilities that each can present. an experiment conducted in an academic environment, using three notebooks with wireless interface and a wireless router, aiming to demonstrate the possibility of decryption from commands using the terminal Kali Linux Operating System as methodology was used. As a result, was obtained that the WEP network password was broken in a few minutes while the WPA and WPA2 networks has been proven that it is possible to break the password, however, it may take a longer time, as the composition of the password. We conclude that the WEP fell into disuse since it was replaced by WPA2, and for more security for this type of network, we recommend preventive actions to be taken in order to increase network security.*

Resumo. *Esse trabalho pretende descrever alguns tipos de criptografia das redes wireless, bem como as vulnerabilidades que cada uma pode apresentar. Como metodologia foi utilizado um experimento realizado em ambiente acadêmico, utilizando-se três notebooks com interface wireless e um roteador sem fio, visando demonstrar a possibilidade de descriptografia a partir de comandos, utilizando o terminal do Sistema Operacional Kali Linux. Como resultado obteve-se que na rede WEP a senha foi quebrada em poucos minutos, enquanto que nas redes WPA e WPA2 foi comprovado que é possível quebrar a senha, entretanto, pode levar um tempo maior, conforme a composição da senha. Conclui-se que a WEP caiu em desuso desde que foi substituída pela WPA2, sendo que para ter mais segurança para este tipo de rede, indicamos ações preventivas a serem adotadas, afim de aumentar a segurança da rede.*

1. Introdução

Conforme a legislação brasileira, a invasão não autorizada de dispositivo informático está prevista na Lei Federal nº. 12.737/2012, a qual dispõe sobre a tipificação criminal de delitos informáticos e que alterou o artigo 154-A do Decreto-Lei no 2.848, de 7 de dezembro de 1940 - Código Penal, conforme cita o artigo 2º da referida Lei Federal:

> Art. 2º.: Art. 154-A. Invadir dispositivo informático alheio, conectado ou não à rede de computadores, mediante violação indevida de mecanismo de segurança e com o fim de obter, adulterar ou destruir dados ou informações sem autorização expressa ou tácita do titular do dispositivo ou instalar vulnerabilidades para obter vantagem ilícita: Pena - detenção, de 3 (três) meses a 1 (um) ano, e multa.

A rede sem fio mal protegida é um alvo fácil para o invasor; nota-se que é imprescindível a correta criação da senha e a escolha da ferramenta correta de criptografia, conforme a necessidade e disponibilidade, aumentando assim o nível de segurança da sua rede *wireless* (NAKAMURA & GEUS, 2010).

O *Penetration Testing (Pentest)* ou teste de intrusão é um método utilizado para avaliar o nível de segurança de uma determinada rede, ou seja, testar as vulnerabilidades da infraestrutura de uma rede ou sistemas operacionais. Conforme Giavaroto & Santos (2013) o *Pentest* possibilita analisar a real estrutura do sistema, que é diagnosticada em todas as áreas inerentes à estrutura de segurança por um auditor. São de suma importância os testes aplicados, pois através deles verificam-se falhas em hardware e software utilizados, criando opções de defesas ou ajustes adequados a tais ataques.

Segundo Nakamura & Geus (2010) o auditor explora brechas em um sistema, cuja segurança depende de todos os níveis de segurança, pois as aplicações podem estar interligadas. No caso de um ataque indevido de um *cracker*, basta apenas uma brecha em um dos níveis de segurança para que ele entre no sistema e tenha acesso à informação pretendida. Assim, a invasão do *cracker* torna-se mais fácil, pois basta encontrar uma falha, enquanto que o auditor tem que tentar identificar todas essas falhas e corrigi-las.

Segundo informações do Centro de Estudo, Resposta e Tratamento de Incidentes de Segurança no Brasil (CERT.br, 2016), foram registrados os seguintes ataques indevidos de *crackers* em computadores e redes nos últimos cinco anos: em 2010 foram 89 invasões; 2011, 106 invasões; 2012, 7.815 invasões; 2013, 11.207 invasões e em 2014, 6.509 invasões. No ano de 1999 a 2011 o número de ataques manteve-se baixo e na mesma média, notando-se um crescimento expressivo a partir do ano de 2012, conforme as estatísticas registradas pelo referido Centro de Estudos.

Cabe ressaltar ainda que existem incidentes que permanecem na “cifra negra”, ou seja, não são denunciados e por isso não chegam ao conhecimento das autoridades responsáveis, deixando no mínimo de serem incluídos nas estatísticas.

Um exemplo disso pode ser uma invasão a uma empresa que depende dos meios tecnológicos para funcionar, podendo ter perdas irreversíveis. Outro exemplo é uma invasão numa simples rede *wireless* residencial, em que exista automação residencial e a invasão indevida nesta rede, poderá fazer com que o invasor cause grandes prejuízos materiais e morais aos proprietários.

Diante de tais constatações, fica evidente que a segurança da informação é um requisito básico, tanto para as grandes corporações, como para os usuários em geral.

2. Fundamentação Teórica

Mediante os autores estudados e citados a seguir, fundamenta-se a teoria a respeito da conceituação da rede *wireless*, os benefícios adquiridos pelo uso, a segurança e possíveis ameaças devido ao crescimento tecnológico, a criptografia como um método de segurança para prevenir a invasão de *crackers*, os tipos de criptografia – WEP, WPA e WPA2 e o procedimento tipo *Pentest* para detectar a vulnerabilidade na infraestrutura da rede.

2.1. Conceito de rede *wireless*

Conforme Nakamura & Geus (2010, p.148) uma rede sem fio – Wi-Fi, *wireless* – utiliza frequência de rádio e ondas eletromagnéticas em infravermelho para a transferência dos dados.

A rede *wireless* possui órgãos que padronizam e regulamentam a utilização da frequência de rádio, ondas eletromagnéticas e protocolos para as suas interfaces.

A *Federal Communications Commision* (FCC) é o órgão regulamentador que padroniza as bandas de frequência dos Estados Unidos da América. Este órgão liberou a faixa de espectro de frequência Industrial, Científica e Médica.

Cada país possui seu órgão regulamentador. No Brasil o órgão responsável pelos limites de potência é a Agência Nacional de Telecomunicações (ANATEL).

Para evitar sobreposições no uso de ondas de rádio, foram criadas faixas de frequência para cada tipo de aplicação. A Figura 1 mostra as três faixas liberadas de frequência ISM, sendo de 900 Mhz, 2.4 Ghz e 5.8 Ghz para as redes *wireless*:

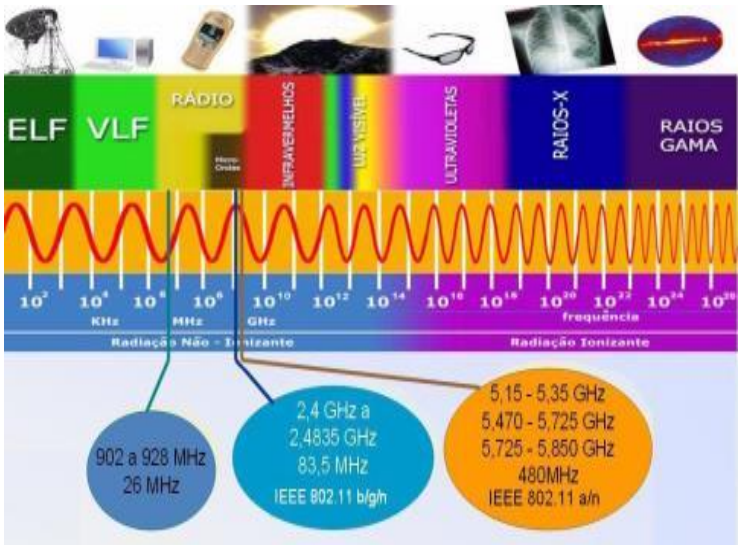

Figura 1: Faixa do espectro da Banda ISM

Fonte: Adaptado de SANTOS JUNIOR (2009)

O *Institute of Eletrical and Eletronic Enginners* (IEEE), fundado nos Estados Unidos, com filiais em muitas partes do mundo, é um órgão importante que gerencia os

padrões para formatos de computadores e dispositivos, sendo que para redes *wireless*, criou protocolos padrões, conforme descrito a seguir:

a) IEEE 802.15 para redes PAN (Rede de Área Pessoal): são dispositivos que se comunicam dentro de uma distância bastante limitada. Exemplo: redes *bluetooth* e UWD (Ultra Wide Band);

b) IEEE 802.11 para redes WLAN (Rede Local Sem Fio): interligam interfaces sem fio dentro do mesmo espaço físico. Exemplo: empresas, escolas e residências;

c) IEEE 802.16 para redes WMAN (Rede Metropolitana Sem Fio): foi projetado para acessos ponto-a-multiponto. Exemplo: interligação de dois escritórios que estão a quilômetros de distância;

d) IEE 802.20 para redes WWAN (Rede de Longa Distância Sem Fio): com o alcance ainda maior, alcança diversas partes do mundo.

2.2. Benefícios da rede *wireless*

Para Nakamura & Geus (2010) o uso de redes *wireless* possui uma série de benefícios, tais como:

a) Mobilidade dos usuários;

b) Instalação rápida: implementação física mais fácil que uma rede cabeada, evitando passagens de fios, otimizando o espaço físico;

c) Flexibilidade: facilidade de criar uma rede *wireless* permanente ou temporária para um evento temporário, além de alcançar lugares onde a rede cabeada não poderia chegar;

d) Escalabilidade: inserção rápida de novas interfaces na rede *wireless*, como *notebook*s, rede de dados e voz, *smartphones,* entre outros dispositivos móveis;

e) Capacidade: atualmente os pontos de acessos, podem chegar à velocidade de 1 Gbps de transmissão de dados, equiparando-se à capacidade de transferência da rede cabeada.

Dentre os benefícios da rede *wireless*, a mobilidade é o principal benefício, pois substitui os cabos de uma rede convencional.

2.3. Segurança da rede *wireless*

A segurança de redes *wireless* tem sido ameaçada devido ao crescimento tecnológico, pois os computadores, *notebooks*, *tablets*, celulares e outros dispositivos tecnológicos já possuem a interface de rede sem fio integrada. Com isso, vários estudos têm sido feitos para combater os possíveis ataques de *crackers,* visando a descoberta da senha dos roteadores para a utilização gratuita desse sinal de internet ou para a invasão desses dispositivos (Nakamura e Geus 2010).

Nakamura & Geus (2010, p.125) relatam que existem novos riscos para os usuários que utilizam a rede *wireless*, pois se antes um *cracker* tinha que ter no mínimo,

contato com um ponto de rede cabeada para ter acesso, hoje para acessar uma rede *wireless* basta apenas ele estar na área de cobertura da mesma.

Giavaroto & Santos (2013) afirmam que para manter seguros e protegidos os seus ativos, tangíveis ou intangíveis, muitas empresas utilizam-se de tecnologias avançadas com soluções que vão desde a instalação de um antivírus, de *firewall* e também complexos algoritmos de criptografia.

2.4. Tipos de criptografia

Atualmente o método de segurança mais utilizado é feito através de senhas, utilizando caracteres que têm por objetivo garantir apenas o acesso do proprietário. Além dos diferentes tipos de ferramentas de criptografia é preciso orientação ao usuário para que a senha tenha requisitos mínimos de segurança, como: troca da senha periódica e também não utilizar senhas simples, dificultando para que uma pessoa sem autorização consiga invadir.

O objetivo da criptografia é que apenas o emissor e o receptor sejam capazes de decifrar a mensagem. Utilizando ferramentas de criptografia, a mensagem permanecerá ilegível, tornando-se acessível através da chave ou senha do usuário.

A chave de segurança é a proteção utilizada nas estações *wireless*, que é compartilhada entre a estação e o ponto de acesso (roteador), fornecendo assim, acesso, desde que a mesma seja digitada corretamente.

Os tipos de chaves usualmente utilizados nos pontos de acesso:

a) Chave WEP (Wired Equivalent Privacy): em 1999, o WEP foi criado com o objetivo de tornar a comunicação sem fio equivalente a uma comunicação com fio. Este tipo de segurança tem sido duramente criticado, devido à sua vulnerabilidade, pois este foi um dos primeiros métodos de chaves de segurança criado para *Wireless* e, sendo assim, se torna mais inseguro à medida que o poder de processamento dos computadores aumenta.

A chave WEP utiliza o algoritmo criptográfico RC4, sendo que suas chaves variam de 40 a 128 bits, que podem ser decifradas em poucos minutos, por meio de um software de ataques (força bruta), porém, para quem utiliza WEP como sua chave padrão, é necessário adotar algumas medidas essenciais:

- Alterar a chave WEP frequentemente;

- Usar filtragem baseada em endereços MAC (*Media Access Control Address*): é feita uma autenticação através do endereço MAC, a partir de uma lista contendo todos os endereços MAC válidos nos pontos de acesso; pode-se impedir que outra interface que não está cadastrada na lista se conecte ao ponto de acesso, porém, é possível fazer a clonagem de endereço MAC, identificando um endereço MAC válido e clonar, obtendo acesso à rede;

- Manter um inventário de todos os equipamentos utilizados;

- Reforçar para os usuários a utilização de Antivírus, Firewall;

- Realizar verificações frequentemente na rede sem fio;

- Deixou de ser considerado um padrão desde 2004;

- Considerar a troca para chave WPA.

b) Chave WPA (*Wi-Fi Protected Access*): diferente da chave WEP, que faz uso de uma criptografia fraca, a chave WPA vem com novos padrões de segurança, corrigindo assim, esse problema.

Criado em 2004, a chave WPA utiliza criptografia feita pelo TKIP (*Temporal Key Integrity Protocol*), conjunto de técnicas para incrementar a segurança, destacando-se:

- Chaves de 256 bits;

- Função de mistura de chave por pacote;

- Nova função de derivação de chave para cada pacote de 48bits;

- Vetor de inicialização estendido com regras de sequência;

- Mecanismo de renovação de chaves.

Entretanto, uma série de elementos do protocolo antigo foi reaproveitada e com ela, diversos problemas do antecessor também acabaram presentes na nova versão.

c) Chave WPA2 (*Wi-Fi Protected Access II*): é o tipo de criptografia mais seguro e utilizado atualmente. Segundo especialistas, o risco de intrusões para usuários domésticos com WPA2 é praticamente zero e não pode ser decifrada por meio de software de ataques (força bruta).

Diferente do TKIP (*Temporal Key Integrity Protocol*), a chave WPA2 vem equipada com o AES (*Advanced Encryption Standard*), um novo padrão para a segurança das informações. É um mecanismo de encriptação que protege os dados que passam pela rede.

Contudo, mesmo com a segurança adicional na WPA2, é bom manter determinados cuidados:

- Esconder o nome da rede;

- Filtragem de MAC;

- Troca de senha periodicamente.

3. Fundamentos para a realização do *Pentest*

Silva & Pereira (2013), Verde (2012) e Vieira (2010), entre outros autores estabeleceram etapas para a realização dos procedimentos de *Pentest*; mesmo não sendo uma regra, esses procedimentos auxiliam na execução do *Pentest*, a partir de um protocolo a ser seguido com determinadas metodologias e procedimentos que podem ser seguidos por todos os profissionais da área.

Nos procedimentos de Vieira (2010), inicialmente é estabelecida uma conversa entre o auditor e o cliente, pautando as necessidades e objetivos do solicitante, sendo um dos procedimentos mais importantes, pois serão estabelecidos os limites permitidos que o teste poderá alcançar.

Nesta fase são coletadas as informações visando designar qual tipo e qual a forma de auditoria que será realizada, fazendo-se um levantamento e posterior

planejamento da modelagem dos testes, descrevendo a infraestrutura contemplada e os equipamentos e recursos tecnológicos necessários para os testes (VIEIRA, 2010).

Vieira (2010) refere ainda que nesta fase estabelecem-se os prazos para a execução dos testes, detalhando-se ainda as intercorrências que possam aparecer e o tipo de testes que será utilizado.

Após a entrevista, todo o procedimento solicitado pelo cliente é descrito em um contrato de prestação de serviço para ambos estarem cientes e assinarem, formalizando-se os serviços que serão realizados.

Na próxima etapa, Vieira (2010) explica que o auditor deve informar ao cliente que vai preparar e organizar os profissionais que serão envolvidos na elaboração da modelagem da estrutura do *Pentest*, montando o planejamento e agendamento dos testes. O próximo passo é a execução da auditoria de *Pentest*.

Giavaroto & Santos (2013) referem que durante a execução o auditor vai fazendo o reconhecimento do sistema, com a elaboração de relatórios técnicos detalhados, incluindo também as soluções pertinentes à avaliação e o *Pentest* vai fazendo a auditoria completa de segurança, explorando todos os aspectos que envolvem a segurança do sistema.

Visando proteger o maior patrimônio que existe para uma empresa – suas informações – o *Pentest* vai aplicar as melhores técnicas de segurança, "reparando hardwares com bug presentes, aplicando patches de segurança, otimizando softwares, políticas de senhas, entre outros, logo após o reconhecimento total do alvo analisado" (GIAVAROTO & SANTOS, 2013, p. 21).

As técnicas de reconhecimento dividem-se em reconhecimento passivo e ativo, sendo que no primeiro são reunidas as informações relativas ao sistema, através de ferramentas da Internet. No reconhecimento ativo são reunidas as informações *in loco*, através das visitas, entrevistas e preenchimento de questionários (GIAVAROTO & SANTOS, 2013).

Nas palavras de Giavaroto & Santos (2013, p. 21), nesta etapa tudo deve ser considerado, desde a estrutura física tecnológica até dados pessoais e da rotina dos funcionários "pessoas relacionadas à empresa, empresas terceirizadas, e-mails, MSN, telefones, tipo de informação que chega ao lixo", entre outras.

Por fim, analisam-se os resultados, contextualizando as informações num relatório final ao cliente, com todas as ocorrências levantadas durante a auditoria e apontando as devidas soluções, pois convém muito mais ao cliente investir na prevenção para diminuir as vulnerabilidades do que sujeitar-se a invasões no seu sistema, com perdas irreversíveis.

Destaca-se que o objetivo principal durante um *Pentest* é a simulação – de forma controlada e organizada – de um ataque real, da mesma forma que aconteceria se um *cracker* atacasse um sistema, demonstrando os danos reais causados por ele e apontando as possibilidades de uma estratégia de prevenção e correção (VIEIRA, 2010).

3.1. Metodologias e tipos de *Pentest*

Vieira (2010) descreve as principais metodologias utilizadas internacionalmente na padronização da auditoria de *Pentest*: *Pentest*ing Frameworks, OSSTMM (*Open Source Security Testing Methodology Manual*), ISSAF (*Information Systems Security Assessment Framework*), OWASP *Testing Guide* (*Open Web Application Security Project*) e NIST SP800-115 e SP800-042 (*National Institute of Standards and Technology*).

Conforme Vieira (2010) existem dois tipos de ataques ao sistema de uma empresa: o interno e o externo, sendo que o interno é aquele realizado por um funcionário da própria empresa auditada e que por isso, possui informações privilegiadas e o externo é aquele realizado por qualquer *cracker* com o intuito de causar danos. Assim, o auditor, através da aplicação do *Pentest* vai se adequando ao ambiente encontrado e através das metodologias, vai buscando as vulnerabilidades do sistema.

Verde (2012), explica que existem seis tipos de *Pentest*, sendo eles:

a) *Blind*: O auditor não conhece nada sobre o alvo, porém, o alvo sabe que será atacado e o que será feito durante a intrusão;

b) *Double Blind*: O auditor não conhece nada sobre o alvo, e o alvo não sabe que será atacado e não sabe quais testes serão efetuados;

c) *Gray Box*: O auditor conhece um pouco sobre o alvo, o alvo sabe que será atacado, e também sabe o que será feito durante a intrusão;

d) *Double Gray Box*: O auditor conhece algumas informações sobre o alvo, o alvo sabe que será atacado, mas não sabe quais ataques serão feitos;

e) *Tandem*: O auditor conhece totalmente o alvo, o alvo sabe que será atacado e o que será feito durante a intrusão;

f) *Reversal*: O auditor conhece totalmente o alvo, mas o alvo não sabe que será atacado e não sabe quais testes serão efetuados.

Neste projeto utilizou-se o tipo *Tandem*, visando demonstrar o uso da ferramenta *Pentest* num roteador *wireless* para descobrir a criptografia das chaves WEP e WPA2.

3.2. *Kali Linux*

O sistema operacional Linux utiliza uma licença livre, não necessitando de custos para adquiri-lo, salvo quando for requisitado um suporte técnico especializado. Existem diversos tipos de distribuições Linux, cada uma com suas características para atender aos diferentes perfis de usuários.

Para o *Pentest*, o Linux é o sistema operacional que mais atende às necessidades do auditor, pois cada distribuição do sistema operacional possui uma gama de ferramentas específicas para a atividade desejada. As distribuições mais utilizadas para o *Pentest* são: *Kali Linux, BackTrack, Matriux, Deft, Caine, BackBox*, entre outras.

De acordo com o *Offensive Security, Kali Linux* é uma distribuição atualizada, robusta e mais completa, pois possui ferramentas adequadas às necessidades do uso do auditor, além de ser gratuito e seguro. Sua versão anterior *Back Track* que faz testes de

penetração e auditoria de segurança *Linux* trabalha para que dados não possam ser acessadas ou vazados de um determinado local, sem o consentimento da empresa.

O *Kali Linux* já vem com ferramentas instaladas específicas para o uso de segurança em redes, possibilitando até mesmo a utilização das aplicações do *Pentest* no Live CD, ou seja, não havendo necessidade de instalar o sistema operacional diretamente no disco rígido.

4. Material e Métodos

No presente estudo foi testado e demonstrado apenas a descriptografia de chaves de rede sem fio, tais como: WEP, WPA e WPA2.

O procedimento com o uso do *Pentest* foi realizado em sala de aula, com a criação das redes sem fio, com criptografia para serem exploradas através da captura de pacotes, utilizando comandos no terminal do sistema operacional Kali Linux.

Para a simulação do ambiente de *Pentest*, foram utilizados três *notebook*s com interface *wireless* e um roteador sem fio para conectar essas interfaces, que possuem as seguintes características:

a) *Notebook* Auditor, marca Acer, modelo aspire 5536, processador AMD Turion X2, dual core 2.10 Ghz, memória RAM 4GB DDR2, com o Sistema Operacional Kali Linux. Este *notebook* foi utilizado para executar a ferramenta *Pentest,* com a finalidade de adquirir a chave da rede *wireless*;

b) *Notebook* Servidor, marca Acer, modelo aspire 5750-6415, processador Intel core i5-2430M 2.4Ghz, memória RAM 6 GB DDR3, com o Sistema Operacional Windows 10. Este *notebook* foi configurado para ser um servidor de arquivos, pois sua finalidade era gerar tráfego na rede sem fio;

c) *Notebook* Cliente, Marca HP, modelo 1000-1460br, processador intel core i5-3230M 2.6Ghz, memória RAM 8 GB DDR3. Este *notebook* foi utilizado para fazer download de arquivos do *notebook* Servidor, a fim de gerar tráfego na rede *wireless*

d) Roteador Wifi, Marca *NetGear*, modelo WGR614 v9, 54Mbps de transferência. Este roteador sem fio foi utilizado para conectar as três interfaces. Primeiro foi configurado com a criptografia do tipo WEP e depois configurado com a chave tipo WPA eWPA2.

Inicialmente, após todas as interfaces ligadas, foi gerado tráfego na rede e executados alguns comandos no terminal do *notebook* auditor para a captura de pacotes da rede Wi-Fi alvo. Para gerar esse tráfego na rede foi compartilhado um arquivo de 1 Gb do *notebook* servidor, para que o *notebook* cliente realizasse o download e gerasse tráfego na rede por um período necessário à execução dos comandos no terminal do *notebook* auditor. Após o download ser iniciado, simultaneamente foram executados os comandos no terminal do *notebook* auditor.

Demonstra-se a seguir na Figura 2, o ambiente físico onde foi realizado o *Pentest*:

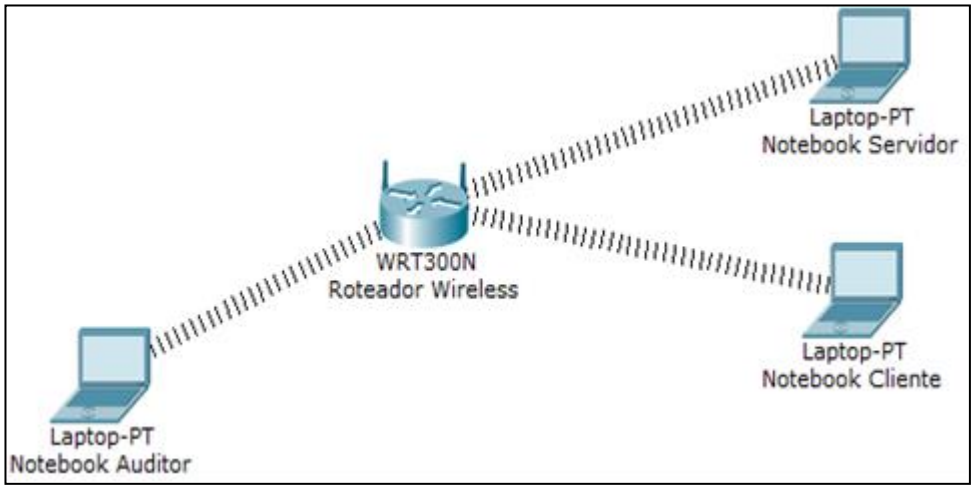

Figura 2: Ambiente Físico para realização do *Pentest*

Fonte: Elaborada pelos autores, 2016.

Existem diversas ferramentas de *Pentest*, sendo que foi utilizada a *aircrack-ng,* a qual trabalha principalmente com Linux, mas também pode ser utilizada nas plataformas *Windows, OS X, FreeBSD, OpenBSD, NetBSD, Solaris e eComStation* 2. A *aircrack-ng* é uma aplicação que utiliza o método de ataque de força bruta, este tipo de ataque, simplesmente tenta usar todas as combinações de caracteres possíveis. Sua licença é gratuita e de código aberto, além de ser nativo na distribuição Kali Linux, não necessitando fazer download.

Todos os procedimentos de simulação foram feitos no terminal do sistema Kali Linux do *notebook* auditor, através de comandos que serão abordados no próximo capítulo de resultados.

5. Resultados

Nesse capítulo são apresentados os resultados obtidos, onde serão demonstrados resultados como: *pentest* para criptografia WEP, *Pentest* para criptografia WPA e WPA2 e Tempo estimado para a descriptografia a partir da composição dos caracteres.

5.1. *Pentest* para criptografia WEP

Para o *Pentest* em criptografia WEP, não foi necessário a criação e execução da *wordlist*, pois, devido seu algoritmo ser mais simples, tornou-se mais fácil decifrá-lo. Utilizando o terminal do *Kali Linux* foram executados poucos comandos para a descoberta da senha de criptografia tipo WEP.

O primeiro comando executado no terminal identificava a interface Wi-Fi:

#airmon-ng

Encontrado a interface Wi-Fi, este comando habilitou a interface de rede *wireless* para o modo monitor, criando uma nova interface para o monitoramento, a mon0:

#airmon-ng start wlan0

Foram listadas todas as redes Wi-Fi que estavam no alcance da interface *wireless* do *notebook* auditor, além de algumas informações dessas redes que foram alcançadas:

#airodump-ng –ivs -w senha -c 6 wlan0mon0

--ivs: vetor de inicialização

-w: nome do arquivo gerado .ivs (senha.ivs)

-c: especifica o canal da rede Wi-Fi

No terminal, em específico na coluna “Data” mostra a quantidade de pacotes capturados. Foi necessário um tráfego na rede de no mínimo 55.000 pacotes. Não é um valor exato, podendo ser necessário um número maior ou menor de pacotes, dependendo da complexidade dos caracteres da senha.

Com os ivs capturados e salvo no arquivo gerado senha.ivs, foi dado o comando para revelar a chave:

#aircrack-ng -a 1 senha.ivs

-a 1: Define o modo de ataque, no caso 1 é para o modo WEP.

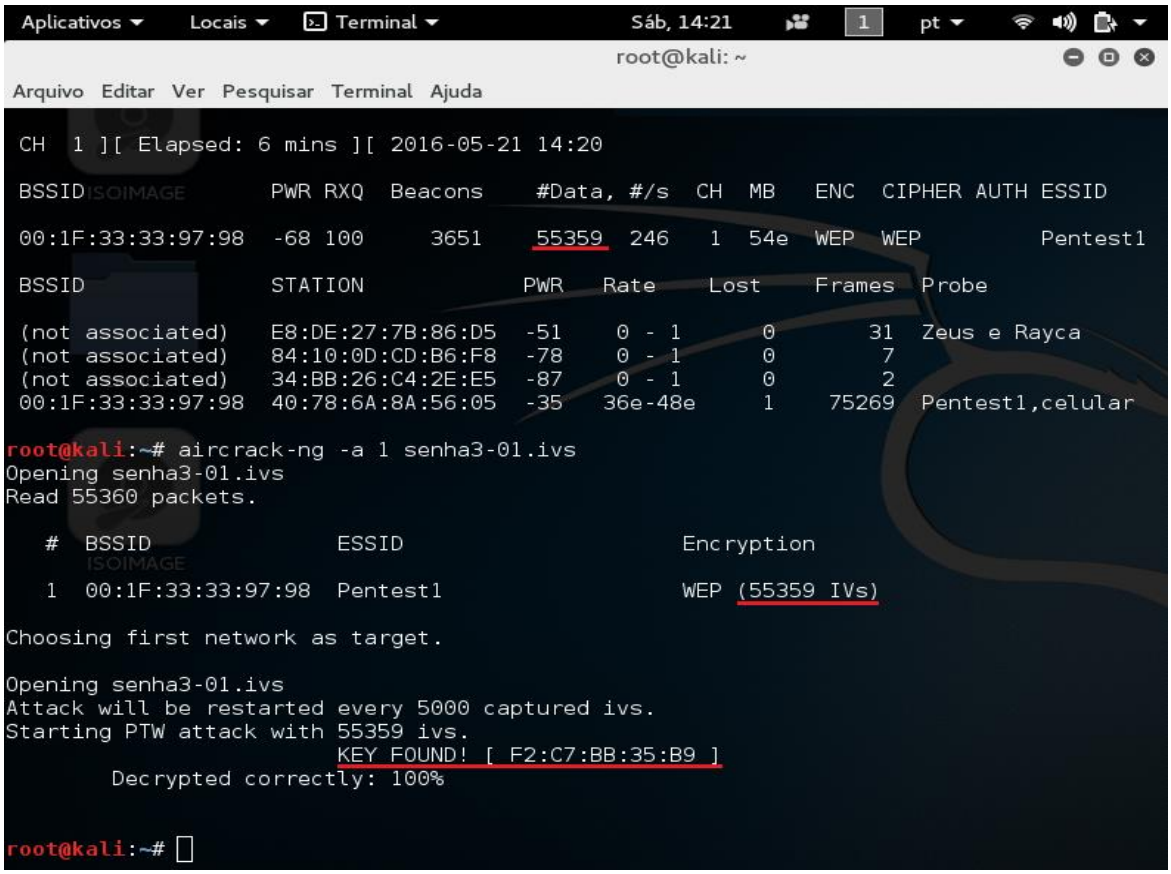

Figura 3: Exemplo da chave da criptografia tipo WEP encontrada

Fonte: Elaborada pelos autores, 2016.

5.2. *Pentest* para criptografia WPA e WPA2

Todo o processo de *Pentest* para descoberta da senha de criptografia tipo WPA ou WPA2, foram executados através de comandos no terminal do Kali Linux.

Este comando foi utilizado para identificar as interfaces de redes:

#airmon-ng

Este comando serve para especificar a interface Wifi. Esta ferramenta habilitou a interface de rede *wireless* para o modo monitor, criando uma nova interface de monitoramento, a mon0:

#airmon-ng start wlan0

Este comando listou todas redes Wi-Fi que estavam no alcance da interface *wireless* do *notebook* auditor, além de algumas informações dessas redes que foram alcançadas:

#airodump-ng wlan0mon0

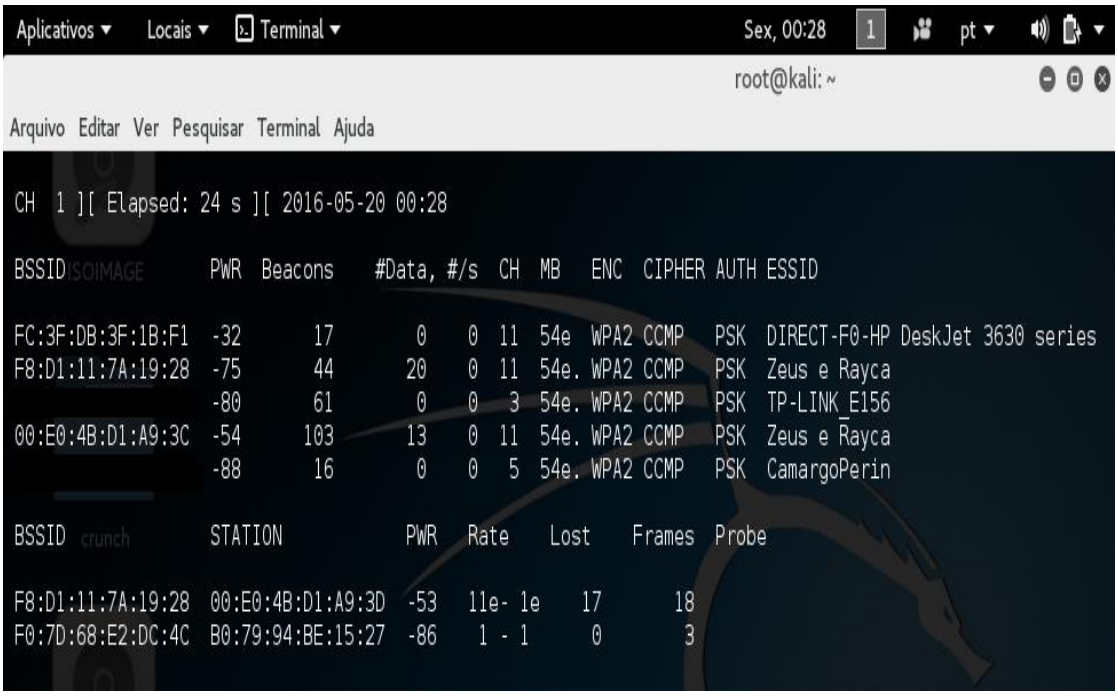

Figura 4: Exemplo de redes Wi-Fi em alcance com o comando airodump-ng

Fonte: Elaborada pelos autores, 2016.

Na segunda linha da Figura 4, foram expostas informações das redes alcançadas pela interface auditor. Descreve-se a seguir o significado de cada item do cabeçalho:

a) BSSID: é a identificação física da interface Wi-Fi;

b) PWR: é o nível do sinal reportado pela interface auditor, sendo que quanto menor esse sinal, mais próximo está a rede *wireless* alcançada;

c) Beacons: são utilizados para demonstrar a existência de uma rede *wireless*, os beacons frames enviam pacotes do ponto de acesso para que os clientes o localizem e iniciem a comunicação para o acesso – cada

ponto de acesso envia cerca de dez beacons por segundo na menor taxa (NAKAMURA & GEUS, 2010);

d) #Data: número de pacotes de dados capturados;

e) #/s: número de pacotes de dados por segundo, medida nos últimos dez segundos;

f) CH: número do canal do ponto de acesso. Além da frequência determinada para a rede Wi-Fi, a configuração do canal auxilia para que não haja interferências entre as redes *wireless.*

g) MB: é a taxa de transferência máxima suportada pelo ponto de acesso;

h) ENC: é o tipo de criptografia utilizado no ponto de acesso;

i) CIPHER: é detectado qual algoritmo utilizado pela chave, não é obrigatório, mas o TKIP é tipicamente usado com a chave WPA e o CCMP é rotineiramente usado para a chave WPA2;

j) AUTH: é o tipo de protocolo usado na autenticação. A criptografia WEP utiliza a chave *SKA (shared key for WEP),* ou seja, chave compartilhada para WEP – A WPA/WPA2 utiliza a chave PSK (*pre-shared key)*, traduzindo, chave pré-compartilhada para WPA/WPA2;

k) ESSID: Mostra o nome da rede sem fio.

Este comando faz parte do processo de *Pentest*, executados determinados comandos em outro terminal:

#airodump-ng- -bssid xx:xx:xx:xx:xx:xx -w pacotes -c 1 wlan0mon0

- w, cria um arquivo, onde os pacotes capturados foram salvos neste arquivo.

- c é o canal da rede alvo, é verificado qual o canal e especificado.

A partir disso ele começou a capturar pacotes da rede selecionada e a salvar no arquivo especificado (no nosso teste o -w pacotes). Automaticamente foi criado um arquivo com o nome escolhido e o formato *01.cap* (exemplo pacotes01.cap).

No terminal, a coluna “Data” mostrou a quantidade de pacotes capturados, sendo necessário o tráfego na rede de no mínimo 500 pacotes. Ainda neste comando apareceu *as Station*, ou seja, as interfaces clientes que estavam conectadas na rede Wi-Fi alvo.

Foi necessário abrir outro terminal para executar o comando:

#aireplay-ng -0 -a xx:xx:xx:xx:xx:xx -c yy:yy:yy:yy:yy:yy wlan0mon0

- 0 fez o usuário conectado ao roteador se desconectar e só voltar a conectar novamente quando o comando foi encerrado. Neste momento os dados que seriam enviados ao cliente que foi desconectado, foram enviados para a interface auditor em que os dados desta comunicação temporária foram salvos no arquivo pacotes-01.cap

-a foi utilizado para especificar o endereço MAC da rede Wi-Fi alvo.

-c o endereço MAC da estação cliente que estava conectado à rede Wi-Fi alvo, sendo escolhido a estação cliente que estava com maior tráfego de dados.

Após esse comando foram enviadas várias requisições entre a rede alvo e a estação cliente. A estação cliente teve a conexo interrompida com a rede Wi-Fi, tanto que na coluna "Data" permaneceu parado o tráfego de dados, e o *frame* de comunicação com o cliente estava aumentado.

Feito este procedimento por poucos segundos, assim que parou, a estação cliente já voltou a ter conexão com a internet. Com isso foi obtido o *handshake*[1], que é o necessário para concluir a descoberta da senha Wi-Fi. A partir deste momento o auditor de *Pentest* já capturou os dados necessários para realizar os procedimentos de descriptografia, não sendo mais necessário o acesso com a rede Wi-Fi alvo.

Para decifrar a criptografia foi utilizado dois métodos que auxiliaram o ataque de força bruta com a ferramenta *aircrack-ng*, ambos os métodos utilizam a aplicação *crunch* que já vem instalada no Kali Linux.

O primeiro método foi necessário a criação de um *wordlist*[2]. Integrado ao *crunch* estava o arquivo de nome *charset.lst* que auxiliou a criação da *wordlist*, contendo listas de caracteres pré-definidas, facilitando a criação da *wordlist*. O *charset.lst* disponibilizou os seguintes conjuntos de caracteres:

- **lalpha:** apenas letras minúsculas;
- **ualpha:** apenas letras maiúsculas;
- **lalpha-numeric:** letras minúsculas e números;
- **ualpha-numeric:** letras maiúsculas e números;
- **lalpha-numeric-all-space:** letras minúsculas, números e caracteres especiais;
- **ualpha-numeric-all-space:** letras maiúsculas, números e caracteres especiais;
- **mixalpha:** letras minúsculas e maiúsculas;
- **mixalpha-numeric-all-space:** letras maiúsculas, minúsculas, números, caracteres especiais e espaço.

Assim, foi utilizado para a criação da *wordlist* o *charset.lst numeric*. Significa que foi criado uma *wordlist,* contendo 88.888.888 milhões de *strings* num arquivo de formato txt de 900 megabytes, para descriptografar uma senha com no máximo 8 dígitos. Foi feita a tentativa da criação de uma *wordlist mixalpha-numeric-all-space,* ou seja, com todos os caracteres possíveis, números, caracteres especiais, espaços, letras maiúsculas e minúsculas, o que geraria uma *wordlist* de aproximadamente 56 Terabytes, sendo necessário abortar essa criação devido a falta de meios. Criamos a *wordlist* numérica para testar senhas com no máximo oito dígitos, para delimitar a criação da lista, foi utilizado o seguinte comando:

---

[1] *Handshake*, é uma palavra que, segundo a tradução do dicionário, quer dizer "aperto de mão". Em terminologia de tecnologia significa quando dois dispositivos se comunicam, eles realizam esse aperto de mão, sendo uma espécie de autenticação para a troca de dados continuar.

[2] *Wordlist* é um dicionário contendo os possíveis caracteres da criptografia. Existem diversos *wordlists* que são disponibilizados para download na internet, porém, podem conter *trojan*s embutido na *wordlist*. O mais indicado e seguro é criar a *wordlist* utilizando softwares que auxiliam na criação.

#crunch 8 8 –f charset.lst numeric –o /root/worlist/numeric8-8.txt

Quanto mais dígitos a senha tiver, maior será o tamanho da *wordlist* e a complexidade para descobrir a chave criptográfica, pois além do tamanho da *wordlist*, o tempo de execução do ataque de força bruta seria de meses ou anos.

Com a *wordlist* criada, foi usado um comando para apontar a *wordlist* e o arquivo criado, contendo os pacotes e o *handshake* capturados:

#aircrack-ng -w '/root/wordlist/numeric8-8.txt (caminho da *wordlist)*' **'/root/pacotes01.cap (caminho dos pacotes obtidos)'**

O segundo método não é necessário a criação de *wordlist,* bastou descrever quais caracteres foram utilizados para o ataque de força bruta e o tamanho mínimo e máximo da senha a ser descriptografada, entre vários testes, foi utilizado o seguinte comando para descriptografar uma senha numérica de oito dígitos:

#crunch 8 8 0123456789 | aircrack-ng -a /root/pacotes01.cap

Outra opção neste método sem *wordlist* foi a possibilidade que a aplicação *crunch* disponibilizou, pois, utilizando o comando "-t" na execução do *crunch*, foi possível indicar caracteres da senha que já eram de conhecimento, facilitando os testes, pois a descriptografia completa dependendo da senha, demorariam meses para ser descoberta. Está opção de indicar parte da senha, o método com *wordlist* não dispõe desta opção. Para indicar parte da senha, foi utilizado o seguinte comando:

#crunch 8 8 0123456789 -t 1234@@@@ | aircrack-ng -a /root/pacotes01.cap

-t: indica os números que são de conhecimento e os caracteres a serem descobertos são marcados com o caractere @.

5.3 Tempo estimado para a descriptografia a partir da composição dos caracteres

A LastBit *Software* (2016), empresa especializada em recuperação de senhas, cita como é realizado o cálculo matemático linear para saber o tempo necessário que demanda para a descriptografia da senha utilizando o ataque de força bruta. A formula é: **(C ^ L) / S / N,** em que **C** é o comprimento do conjunto de caracteres – exemplo o numérico contém 10 caracteres, elevado à potência de **L,** que é o comprimento da senha, com o resultado dessa potência, dividimos por **S,** que é o número de senhas testadas por segundo pelo computador auditor e dividimos por **N,** que é a quantidade de computadores que estão executando este processo, no caso, foi somente o *notebook* auditor. O resultado final obtido é dado em segundos, sendo necessária a conversão em minutos, dias, meses, anos, conforme a composição da senha.

Com base na formula matemática, demonstra-se na Tabela 1, o tempo necessário para descriptografia de diversas senhas, conforme sua composição e quantidade de caracteres vale ressaltar que os valores citados na Tabela 1 foram arredondados para melhor compreensão do leitor. Foi utilizado somente o *notebook* auditor para execução da ferramenta *aircrack-ng:*

Tabela 1: Tempo Estimado de acordo com a quantidade e composição dos caracteres

Conjunto de Caracteres	Tamanho da senha					
	3	4	5	6	7	8
Números (10 caracteres)	1 segundos	10 segundos	2 minutos	17 minutos	3 horas	28 horas
Letras Minúsculas ou Maiúsculas (26 caracteres)	18 segundos	8 min	3 horas	4 dias	4 meses	8 anos
Letras Minúsculas e Maiúsculas (52 caracteres)	3 minutos	2 horas	4 dias	10 meses	41 anos	2148 anos
Letras Minúsculas, Maiúsculas, Números (62 caracteres)	4 min	4 horas	11 dias	28 meses	141 anos	8774 anos
Letras Minúsculas, Maiúsculas, Números e Pontuações (77 caracteres)	8 min	10 horas	31 dias	8 anos	645 anos	49661 anos
Letras Minúsculas, Maiúsculas, Números, Pontuações e Caracteres Especiais (93 caracteres)	14 min	21 horas	80 meses	26 anos	2418 anos	224883 anos

No presente estudo e conforme orientação bibliográfica consultada o ataque de força bruta utilizado com a ferramenta *aircrack-ng* no *notebook* auditor, juntamente com os dois métodos da aplicação *crunch* com e sem as *wordlists*, utilizou todos os caracteres possíveis, ou seja, uma espécie de permutação de cada conjunto de dígito, com o objetivo de descobrir as diferentes composições de senhas com diversos tamanhos de dígitos da rede Wi-Fi alvo com criptografia WPA e WPA2, conforme apresentado na Tabela 2, em vermelho estão os caracteres descobertos nos testes:

Tabela 2: Tempo gasto de acordo com a quantidade e composição dos caracteres

Senha	Quantidade de caracteres descriptografados	Caracteres Descobertos	Média de chaves testadas por segundos	Total de chaves testadas	Utilizou *wordlist*	Tempo gasto
12345678	8	13456789	1.000	12.345.676	Sim	2 horas e 38 minutos
99999999	8	99999999	1.700	88.888.888	Não	14 horas e 41 minutos
pentest	5	ntest	1.700	8.765.532	Não	1 hora e 27 Minutos
PeNtEsTs	4	EsTs	1.700	4.269.276	Não	42 minutos
A1b2C3d4	4	C3d4	1.700	6.881.016	Não	1 hora e 8 minutos
172839Br!	4	3Br!	1.700	15.111.124	Não	2 horas e 29 minutos

Utilizando a *wordlist* notou-se a oscilação da média de chaves testadas por segundo do ataque de força bruta, testando em média mil chaves por segundo e este valor variou durante o teste, entre mil chaves a mil e quinhentas chaves por segundos, está variância ocorre, pois, esse tipo de ataque utiliza diretamente a capacidade do processador e o desempenho do disco rígido, pois foi necessário acessar a *wordlist* salva no disco rígido.

Já no método sem utilizar a *wordlist,* ficou evidente a superioridade, pois na quantidade de chaves testadas por segundo, a média foi de mil e setecentas chaves por segundo e essa média foi constante durante o ataque de força bruta, pois esse tipo de

ataque utilizou exclusivamente a capacidade do processador. Outro fator positivo nesse método foi a possibilidade de indicar caracteres da senha que já eram de conhecimento, através do comando "-t", facilitando os testes, pois com o método de *wordlist* não foi encontrada está opção. Devido a eficiência na quantidade de chaves testadas por segundos e a indicação de caracteres da senha já descoberto no método sem *wordlist,* tornou-se mais viável a realização dos testes de ataque de força bruta sem *wordlist*, utilizando a aplicação *crunch*.

A quantidade de memória RAM e outros hardwares não afetaram a velocidade dos ataques de força bruta dos dois métodos.

Na Figura 5, utilizando o ataque de força bruta com a ferramenta *aircrack-ng* com uso da *wordlist,* demonstra o tempo gasto para o *notebook* auditor descriptografar uma senha numérica de oito dígitos de uma rede Wi-Fi configurada com a criptografia WPA2.

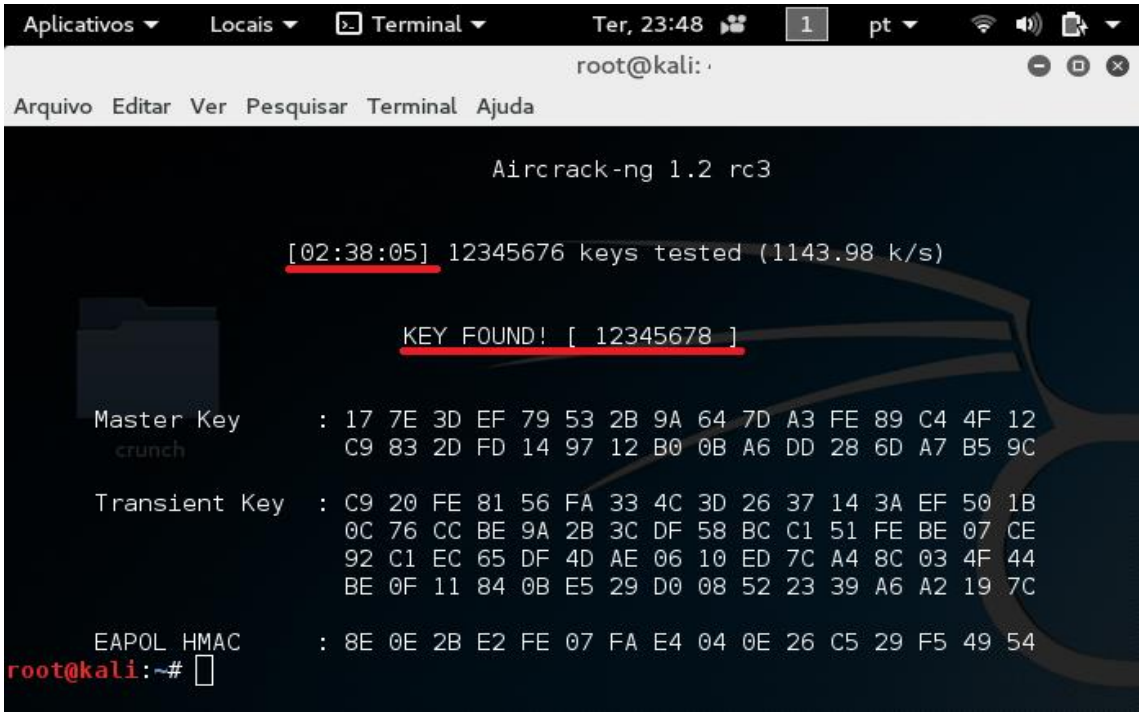

Figura 5: Exemplo de descriptografia de uma rede Wi-Fi WPA2 com utilização de *wordlist*

Fonte: Elaborada pelos autores, 2016.

Nota-se ainda na Figura 5, que para encontrar uma simples senha numérica "12345678" utilizando o método com *wordlist*, foi necessário testar 12.345.676 milhões de chaves, em uma média de mil chaves testadas por segundos, o que demandou 2 horas e 38 minutos para encontrar está senha numérica.

Na Figura 6, para encontrar a senha numérica "99999999" utilizando o método *crunch* sem wordlist, foi necessário testar 88.888.888 milhões de chaves, em uma média de mil e setecentas chaves por segundos, foi necessário 14 horas e 41 minutos para encontrar está senha numérica.

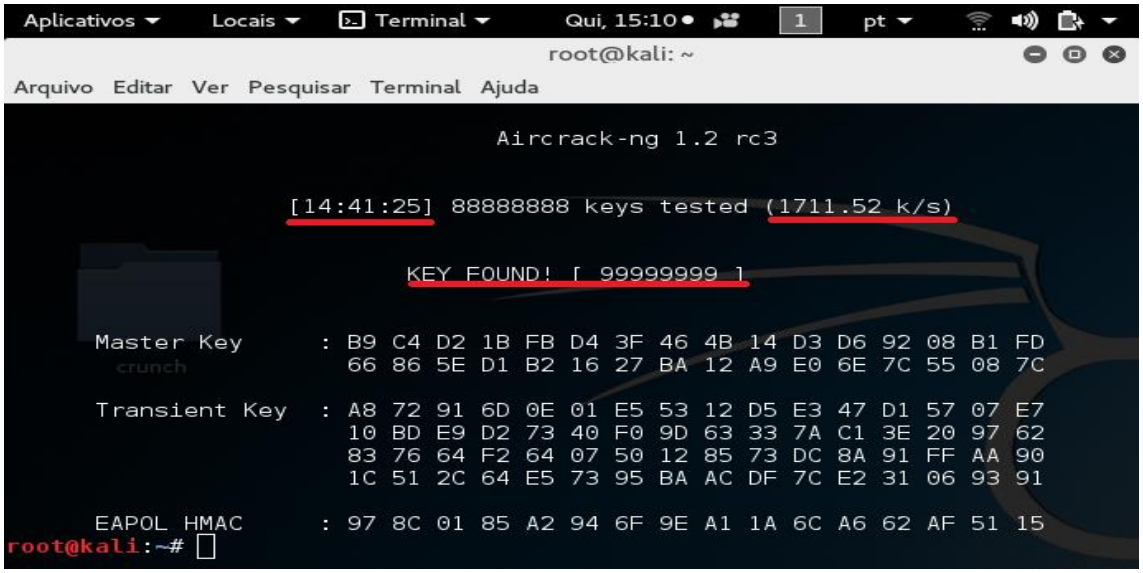

Figura 6: Exemplo de descriptografia de uma rede Wi-Fi WPA2 sem utilização de *wordlist*

Fonte: Elaborada pelos autores, 2016.

Percebeu-se que mesmo as duas senhas numéricas “12345678” e “99999999” tendo oito dígitos, a quantidade de chaves testadas e consequentemente o tempo gasto para descobrir a senha foram diferentes, devido a quantidade de permutação que o ataque de força bruta realizou para encontrar as senhas.

Na rede WEP a senha foi quebrada com muita facilidade, mesmo aquelas que continham números, letras e caracteres especiais. Não foi necessário o uso da *wordlist* ou de especificar os caracteres que teriam que ser testados.

Para quebra de senha em redes WPA e WPA2, foi possível quebrá-las, porém, foi necessário um ataque de força bruta, que funciona como uma espécie de permutação dos dígitos, testando todos as possíveis chaves da senha até encontrá-la, o tempo gasto nos testes para as redes WPA e WPA2 foram praticamente iguais, pois ambos necessitam do ataque de força bruta. Foi utilizado para o ataque a ferramenta *aircrack-ng* que com o auxílio da aplicação *crunch* possibilitou dois métodos para a descriptografia, sendo o primeiro método, foi criado uma *wordlist* numérica para descobrir senhas de oito dígitos e já no segundo método, não criamos *wordlist*, pois especificamos os caracteres que seriam utilizadas no ataque com a ferramenta *crunch,* o que resultou uma eficiência na quantidade de chaves testadas por segundo, diminuindo assim o tempo em aproximadamente 60% para a descriptografia, isentando também a necessidade de criar *wordlists* que ocupam uma certa quantidade de espaço no disco rígido. Devido aos algoritmos de segurança desse tipo de criptografia, o tempo na execução do *Pentest* pode demorar meses, de acordo com a quantidade e composição dos caracteres.

As utilizações desses comandos no terminal se mostraram muito simples e podem ser usados por usuários com nível de conhecimento básico no sistema operacional Linux.

6. Conclusão

Como resultado obteve-se que para capturar pacotes nas redes WEP, WPA e WPA2, não há dificuldades para tal procedimento, entretanto, após a captura desses pacotes a descriptografia para a descoberta da senha WPA e WPA2, torna-se algo mais criterioso e de difícil a descoberta conforme a composição da senha e a quantidade de caracteres, devido ao tempo gasto para obter a chave criptográfica.

Mediante os autores estudados e após os experimentos com o *Pentest* é possível afirmar que não existe criptografia inquebrável, pois sempre existirá um meio capaz de desvendá-la, desde que se tenha tempo e recursos para tanto.

Por outro lado, o objetivo do presente estudo foi atingido, uma vez que pretendeu-se descrever alguns tipos de criptografia das redes *wireless*, bem como as falhas que pode apresentar. Com isso, define-se a força de um algoritmo de criptografia e da capacidade que o mesmo tenha em suportar a um ataque de crackers por um tempo suficiente para que a senha utilizada seja descriptografada e o ataque seja descoberto, ou o especialista em ataque desista da operação.

Algumas técnicas de descriptografia utilizadas tentam encontrar pontos fracos no algoritmo, de forma que um determinado padrão no texto criptografado possa ser associado ao texto aberto correto, ou à chave integral ou parcial, que seria a ferramenta *Crunch*.

Em análise das criptografias utilizadas nas redes wireless, as mais antigas, baseadas em WEP, eram muito simples, pois deixavam vazar dados que podiam ser utilizados para recuperar o texto original. Já o método seguinte, WPA e WPA2, também possui falhas, apesar de ser muito mais resistente a ataques de força bruta.

Portanto, conclui-se que a rede WEP caiu em desuso desde que foi substituída pela rede WPA2, pois independentemente da quantidade de caracteres, seu algoritmo é fraco, tornando-se fácil a sua descoberta.

No caso da WPA e da WPA2, se a senha que estiver sendo utilizada contiver letras, números e caracteres especiais e no mínimo com oito dígitos – conforme requisito mínimo dos pontos de acessos – torna-se mais difícil a quebra, pois a sua descoberta com apenas um computador auditor executando o ataque de força bruta demoraria décadas para a sua descoberta.

Referências

BRASIL. Lei nº. 12.737 de 30/11/2012. Dispõe sobre a tipificação criminal de delitos informáticos; altera o Decreto-Lei nº 2.848, de 7 de dezembro de 1940 - Código Penal; e dá outras providências. Disponível em: <http://www.planalto.gov.br/ccivil_03/ _ato2011-2014/2012/lei/l12737.htm>. Acesso em: 17 MAI 2016.

CERT.br. Centro de Estudos, Resposta e Tratamento de Incidentes de Segurança no Brasil. Disponível em: <http://www.cert.br/>. Acesso em: 17 maio 2016.

GIAVAROTO, S; SANTOS, G. Backtrack Linux: auditoria e teste de invasão em redes de computadores. 1ª Ed. São Paulo: Moderna, 2013.

IEE, Institute of Eletrical and Eletronics Enginners. Disponível em: <http://grouper.ieee.org/groups/802/> Acesso em 10 ABR 2016.

LAST BIT SOFTWARE. Disponível em: < http://lastbit.com/password-recovery-methods.asp#Brute Force Attack> Acesso em 01 JUN 2016.

NAKAMURA, E. T.; GEUS, P. L. de. Segurança de Redes em Ambientes Cooperativos. Rio de Janeiro: Novatec, 2010.

OFFENSIVE SECURITY. Kali Linux. Disponível em:<http://www.kali.org/official-documentation/>. Acesso em: 03 ABR. 2016.

SANTOS JUNIOR, Arthur R. dos. Palestra segurança *wireless* Workshop Camehil e Instituto Online, 2009. Disponível em: <http://institutoonline.com.br/downloads.php> Acesso 11/03/2016.

SILVA, R; PEREIRA, J. Identificando vulnerabilidades de segurança computacional. Disponível em <http://web.unipar.br/~seinpar/2013/ artigos/Raquel%20 Fonseca% 20da%20Silva.pdf> Acesso em 30 MAI 2015.

VERDE, Evandro Villa. Mini Curso: PentestUnivem Disponível em <http://aberto.univem.edu.br/bitstream/handle/11077/704/mini_pen_univem.pdf?sequence=1> Acesso em 30 MAI 2015.

VIEIRA, Luiz. Pentest Curso Teste de Invasão em Redes e Sistemas. Local: OYS, Niterói Rio de Janeiro, 2010, 261p. Apostila.