ED' WILSON TAVARES FERREIRA

PROPOSTA DE UM SISTEMA DE DETECÇÃO E CLASSIFICAÇÃO DE INTRUSÃO EM REDES DE COMPUTADORES BASEADO EM TRANSFORMADAS WAVELETS E REDES NEURAIS ATIFICIAIS

Tese de Doutorado apresentado ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial para obtenção do título de Doutor em Ciências.

Orientador: Prof. Dr. Gilberto Arantes Carrijo

Uberlândia
2009

Ed' Wilson Tavares Ferreira

Proposta de um Sistema de Detecção e Classificação de Intrusão em Redes de Computadores Baseado em Transformadas Wavelets e Redes Neurais Artificiais

Tese de Doutorado apresentado ao Programa de Pós-Graduação em Engenharia Elétrica da Universidade Federal de Uberlândia, como requisito parcial para obtenção do título de Doutor em Ciências.

Uberlândia, 18 de Dezembro de 2009.

Banca Examinadora

_____

Prof. Dr. Gilberto Arantes Carrijo - UFU

_____

Prof. Dr. Keiji Yamanaka- UFU

_____

Prof. Dr. Paulo Roberto Guardieiro - UFU

_____

Prof. Dr. Ruy de Oliveira - IFMT

_____

Prof. Dr. João Paulo Ignácio Ferreira Ribas - UFMT

Ao meu pai, Wilson Ferreira (*in memoriam*).

AGRADECIMENTOS

Agradeço a Deus, o Grande Arquiteto do Universo.

Ao meu orientador, Prof. Dr. Gilberto Arantes Carrijo, por acreditar na minha capacidade.

Ao Prof. Dr. Ruy de Oliveira, pela valorosa co-orientação e participação ativa nos trabalhos de pesquisa.

Aos demais professores que constituíram a banca examinadora para avaliar esse trabalho de pesquisa.

Ao colega do Grupo de Pesquisa de Redes e Segurança – GPRS, Nelcileno Virgilio de Souza Araújo, pela cooperação nos trabalhos.

Aos colegas do Instituto Federal de Mato Grosso – IFMT.

A minha mãe, minhas irmãs, meu sogro e sogra, pelo incentivo, apoio e torcida pelo êxito desse trabalho.

Finalmente, a minha esposa, Patrícia, por estar sempre ao meu lado.

A todos que contribuíram pela realização desse projeto.

RESUMO

Como a internet tem proporcionado grande número de interconexões entre as redes, hoje a segurança da informação tornou-se muito importante para garantir a confidencialidade, integridade e disponibilidade dos recursos computacionais. Os Sistemas de Detecção de Intrusão (IDS) devem ser capazes de identificar ações maléficas que podem comprometer essas garantias tão rápido quanto possível, além disso, devem utilizar baixo poder computacional. Nessa tese, apresenta-se uma abordagem híbrida para construção de IDS, através do uso de duas técnicas distintas: transformadas wavelets e rede neural artificial. As transformadas wavelets são utilizadas para indicar detectar comportamentos anômalos na rede, enquanto que as redes neurais são empregadas para classificação dos ataques. Foi desenvolvido um protótipo e foram avaliados dados oriundos de simulação, testes em rede de laboratório e a base do KDD99. Além da análise de resultados de outras propostas, também foi realizado uma comparação com a técnica de aprendizado por quantização vetorial. Em todos os experimentos, bons resultados foram obtidos, demonstrando que a abordagem proposta é bastante promissora.

Palavras chaves: Sistema de detecção de intrusão IDS. Transformadas wavelets. Redes neurais artificiais. Aprendizagem por quantização vetorial. Segurança em redes de computadores.

ABSTRACT

As the Internet has become an enormous interconnected network, the information security today is very important to guarantee confidentiality, integrity and availability of computing resources. Advanced Intrusion Detections Systems (IDS) should be capable of identifying malicious actions that may compromise these guarantees, as quickly as possible. In this work, we present a hybrid approach for the IDS, with two different techniques: wavelets and artificial neural network. The wavelet is used to indicate to detect anomalous behavior on the network, while the neural networks are used to classify the attacks. A prototype was developed and evaluated data from simulation tests, on laboratory network and the KDD99 database. Besides the analysis of results of the other approaches, was also carried out a comparison with the learning vector quantization. Good results were obtained in all experiments, this demonstrating that the approach is very promising.

Keywords: Intrusion Detections Systems (IDS). Wavelets. Neural artificial networks. Learning vector quantization. Network security.

LISTA DE ILUSTRAÇÕES

Figura 2.1 - Rede de computadores *20*
Figura 2.2 - Redes Convergentes *21*
Figura 2.3 - Modelo de referência TCP/IP *22*
Figura 2.4 - Canais do padrão IEEE 802.11 ..... *25*
Figura 2.5 - Quadro da subcamada MAC, fonte (Kurose e Ross, 2006) *27*
Figura 2.6 - Esquema Básico de Acesso no DCF *28*
Figura 2.7 - Terminal Oculto, fonte (Kurose e Ross, 2006). *28*
Figura 2.8 - DCF com uso de RTS e CTS *29*
Figura 2.9 - Arquitetura IEEE 802.11, fonte (Walke, Mangold et al., 2006) *30*
Figura 2.10: Tipos de Redes, fonte (Mohammad, 2003). *32*
Figura 2.11 - Topologia WMN *34*
Figura 2.13 - Inundação de pacotes numa rede wireless *37*
Figura 2.12 - Alguns Protocolos de Roteamento Ad Hoc *37*
Figura 2.14 - Inundação de pacotes numa rede wireless com MPR *38*
Figura 2.15 - Rede sem fio com indicação de banda *39*
Figura 2.16 - Propagação de Mensagens RREQ e RREP do AODV *41*
Figura 2.17 - Protocolo DSR, fonte (Johnson, Hu et al., 2007). *42*
Figura 2.18 - Protocolo CEDAR, fonte (Kilinkaridis, 2007). *44*
Figura 2.19 – Protocolo ZRP, fonte (Kilinkaridis, 2007). *45*
Figura 3.1 - Modelo de um Neurônio *59*
Figura 4.1 - Incidentes de Segurança Reportados no Brasil, fonte: (Centro De Estudos Resposta E Tratamento De Incidentes De Segurança No Brasil, 2009) *63*
Figura 4.2 - Arquitetura Genérica de um IDS *68*
Figura 5.1 - Algoritmo da Primeira Camada *76*
Figura 5.2 - Topologia Simulada *78*
Figura 5.3 - Taxa de Recebimento dos Nós *80*
Figura 5.4 - Algumas Famílias de Wavelet *81*
Figura 5.5 - Topologia da Rede no Laboratório *82*
Figura 5.6 - Resultados experimentados pelo IDS híbrido proposto no cenário de ataque TCP/RPC. *84*
Figura 5.7 - Curva de Treinamento da Rede Neural *85*

Figura 5.8 - Topologia da Geração de Dados do KDD99 86
Figura 5.9 – Exemplos de Conexões na Base do KDD 86
Figura 5.10 - Algoritmo para Análise da Base do KDD através de Poisson 93
Figura 5.11 - Percentual de Amostras de Treinamento 94
Figura 5.12 - Histograma da Característica "duration" 98
Figura 5.13 - Histograma da Característica "source bytes" 99
Figura 5.14 - Histograma da Característica "is guest login" 100
Figura 5.15 - Precisão Global (1.000 amostras para treinamento e 5.000 para testes) 102
Figura 5.16 - Precisão Global (10.000 amostras para treinamento, 50.000 conexões para testes) 103
Figura 5.17 - Estrutura de Teste 105
Figura 5.18 - Algoritmo LVQ 108
Figura 5.19 - Quantidade de Conexões por Tipo no KDD10 110

LISTA DE TABELAS

Tabela 2.1 - Padrões IEEE 802.11 26
Tabela 2.2- Comparação entre Redes 36
Tabela 2.3 - Comparação entre Alguns Protocolos para Rede Ad Hoc Sem Fio 45
Tabela 2.4 - Principais Ataques às Redes Ad Hoc 51
Tabela 4.1 - Métricas de Avaliação de IDS 69
Tabela 5.1 - Classes de Reconhecimento do Tráfego 76
Tabela 5.2 - Parâmetros da Simulação 78
Tabela 5.3 - Rede em Laboratório 82
Tabela 5.4 - Períodos dos Ataques 83
Tabela 5.5- Características Básicas no KDD 99 87
Tabela 5.6- Características Sugeridas no KDD 99 88
Tabela 5.7- Características de Tráfego com Janela de 2s no KDD 99 88
Tabela 5.8- Características de Tráfego Calculadas Utilizando o Histórico das Últimas 100 Conexões no KDD 99 89
Tabela 5.9 - Classes das Conexões no KDD 99 90
Tabela 5.10 - Comparação com outras distribuições 94
Tabela 5.11 Comparação entre Conjunto Completo e Parcial do KDD 95
Tabela 5.12 Precisão Global Conforme Seleção de Algumas Características do KDD 96
Tabela 5.13 - Precisão Global (1.000 amostras para treinamento 5.000 para testes) 101
Tabela 5.14- Precisão Global (10.000 amostras para treinamento, 50.000 conexões para testes) 102
Tabela 5.15 – Precisão Global Conforme Seleção de Alguns Atributos 104
Tabela 5.16 - Comparação entre Diversos Trabalhos 106
Tabela 5.17 - Resultados com LVQ 110

LISTA DE ABREVIATURAS E SIGLAS

AODV: *Ad Hoc On-demand Distance Vector*.

AP: *Access Point*.

ARPANet: *Advanced Research Projects Agency Network*

ATM: *Asynchronous Transfer Mode*.

BPSK: *Binary Phase Shift Keying*

BSS: *Basic Service Set*

CCK: Complementary Code Keying

CEDDAR: Core Extraction Based Distributed Ad Hoc Routing

CSMA/CA: *Carrier Sense Multiple Access/Collision Avoidance*.

CTS: *Clear to Send*

CTWS: *Continue Time Wavelet Series*

CWT: *Continuous Wavelet Transform*

DIFS: *Distributed Inter Frame Spaces*

DNS: *Domain Name System*

DQPSK: *D Quadrature PSK*

DSDV: *Destination Sequenced Distance Vector*

DSR: Dynamic Source Routing

DSSS: *Direct Sequence Spread Spectrum.*

DTWS: *Discrete Time Wavelet Series*

DWT: *Discrete Transform Wavelet*

ETX: *Expected Transmission Count*

FDM: *Frequency-division multiplexing*

FEC: *Forward Error Corrector*

FHSS: *Frequency Hopping Spread Spectrum*.

FTP: *File Transfer Protocol*

GHz: Gigahertz

HIDS: *Host-based IDS*

HTTP: *Hyper Text Transfer Protocol*

HTTPS: *Hyper Text Transfer Protocol Secure*

IDS: *Intrusion Detection System*

IPSEC: *Internet Protocol Secure*

KDD: *Knowledge Discovery in Database*.

LVQ: *Learning Vector Quantization*

MAC: *Medium Access Control.*

Mbps: Mega bits por segundo

MHz: Mega hertz

MIT: *Massachusetts Institute of Technology*

MLP: *Multilayer Perceptron*

MPLS: *Multiple Protocol Label Switch*

MPR: Multipoint Relay

MSE: *Mean Squared Error*

NIDS: *Network-based IDS*

OFDM: *Orthogonal Frequency Division Multiplexing*

OLSR: *Optimized Link State Routing*

OLSR-ML: *OLSR Minimum Loss*

PDA: *Personal Digital Assistants*

QoS: *Quality of Service*

QPSK: Quadrature Phase Shift Keying

RFC: *Request for Comment*.

RM-OSI: *Reference Model - Open Systems Interconnection*

RN: Reported Node

RPC: *Remote Procedure Call*

RREP: *Route Reply*

RREQ: Route Request

RT: Reported Tree

RTS: *Request to Send*

s: *Medida de tempo expressa em segundos*.

SIFS: *Short Inter Frame Spaces*

SN: *Node Sequence Number*

SSH: *Secure Shell*

STFT: *Short Time Fourier Transform*

SVM: *Support Vector Machine*

TBRPF: Topology Broadcast Based Reverse Path Forwarding

TCP/IP: *Transmission Control Protocol/Internet Protocol*

TDM: *Time-division multiplexing*

UDP: *User Datagram Protocol*

VPN: *Virtual Private Network*

Wifi: Wireless Fidelity

WMN: *Wireless Mesh Network*

ZRP: Zone Routing Protocol

SUMÁRIO

1 INTRODUÇÃO ____ *15*

1.1 Objetivos ____ 16

1.2 Justificativa ____ 16

1.3 Metodologia ____ 17

1.4 Contribuições ____ 17

1.5 Estrutura da Tese ____ 17

2 REDES DE COMPUTADORES ____ *19*

2.1 Introdução ____ 19

2.2 Redes de Computadores ____ 19

2.3 Família de Protocolos TCP/IP ____ 22

2.4 Padrões IEEE 802.11 ____ 24

2.5 Redes Ad Hoc Sem Fio ____ 31

2.6 Redes em Malha Sem Fio ____ 33

2.7 Roteamento em Rede Ad Hoc Sem Fio ____ 35

2.8 Principais Ataques às Redes Ad Hoc Sem Fio ____ 46

2.9 Comentários Finais ____ 53

3 TRANSFORMADAS WAVELETS E REDES NEURAIS ARTIFICIAIS ____ *54*

3.1 Introdução ____ 54

3.2 Wavelet ____ 54

3.3 Redes Neurais Artificiais ____ 58

3.4 Comentários Finais ____ 61

4 SISTEMAS DE DETECÇÃO DE INTRUSÃO ____ *62*

4.1 Introdução ____ 62

4.2 Sistema de Detecção de Intrusos ____ 63

4.3 Algumas Abordagens Propostas para Construção de Sistemas de Detecção de Intrusão ____ 69

4.4 Comentários Finais 71

5 SISTEMA DE DETECÇÃO DE INTRUSÃO COM ABORDAGEM BASEADA EM PROCESSAMENTO DIGITAL DE SINAIS E REDES NEURAIS PARA REDES DE COMPUTADORES *73*

5.1 Introdução 73

5.2 Abordagem Proposta com Utilização das Transformadas de Wavelet e Redes Neurais Artificiais 73

5.3 Simulações e Resultados 77

5.3.1 Avaliação de Algumas Famílias de Wavelets 77

5.3.2 Emprego da Proposta em uma Rede em Laboratório 82

5.4 Estudo de Caso – Utilização do KDD 99 com Dados de Auditoria 85

5.4.1 Descrição da Base de Dados do KDD 99 85

5.4.2 Análise da Base do KDD 99 Através da Distribuição de Probabilidade de Poisson 91

5.4.3 Seleção de Características da Base do KDD 99 para o IDS 95

5.5 Resultados Obtidos 100

5.6 Comparação dos Resultados com Outras Abordagens Propostas 105

5.7 Comparação dos Resultados com a Aprendizagem por Quantização Vetorial 106

5.8 Comentários Finais 111

6 CONCLUSÕES E TRABALHOS FUTUROS *112*

REFERÊNCIAS BIBLIOGRÁFICAS *115*

1 INTRODUÇÃO

Com o crescimento dos serviços disponibilizados pelas redes de computadores, a facilidade em encontrar informações na internet, além do aumento do número de usuários, a segurança da informação tornou-se fundamental para garantir a integridade, disponibilidade e confidencialidade dos dados e da rede. Uma ação maléfica ou não intencional pode comprometer o sistema, caracterizando uma intrusão. O sistema de detecção de intrusão deve conseguir identificar essa ação, mas sem comprometer o funcionamento normal da rede. Os sistemas de detecção de intrusão são ferramentas de segurança que, como outras medidas, a exemplo de antivírus, destinam-se a reforçar a segurança da informação (Teodoro, Verdejo *et al.*, 2009).

Diferentes técnicas têm sido propostas para implementação de sistemas de detecção de intrusão. Geralmente as abordagens são específicas para determinados tipos de redes (redes ad hoc sem fio, redes de sensores, redes com fio, redes de alta velocidade), pois cada modelo possui características particulares que influenciam seu funcionamento. O grande número de propostas que são apresentadas atualmente permite concluir que se trata de uma área que está em desenvolvimento. À medida que novos tipos de redes ou tecnologia são desenvolvidos e disponibilizados, novas brechas de segurança são descobertas e também novos tipos de ataques exploram essas vulnerabilidades.

Um sistema de detecção de intrusão deve ser capaz de identificar ações maléficas, porem sem comprometer o funcionamento normal da rede. Além disso, os IDS não devem consumir recursos computacionais a ponto de prejudicar as aplicações dos usuários, como também não utilizar excessivamente a largura de banda da rede para comunicação.

Nessa tese, é apresentada uma proposta de IDS para rede de computadores. Essa abordagem faz uso das transformadas wavelets e de redes neurais artificiais. As wavelets são utilizadas para reconhecer anomalias na rede de computadores, enquanto que as redes neurais são utilizadas para classificação dos ataques. Os resultados de avaliações também são apresentados. Foram realizados experimentos com diversos cenários: dados oriundos de simulação, de uma rede em laboratório e por fim, uma base de testes. No experimento realizado a partir de uma simulação, foi

empregada uma topologia de rede sem fio, configurada no modo ad hoc. A avaliação em laboratório ocorreu com o uso de uma rede sem fio composta por três nós. Por fim, a outra avaliação ocorreu através de testes de detecção e classificação dos ataques em dados obtidos através de uma base de testes bastante utilizada em avaliações de IDS. Também foram avaliadas outras técnicas de reconhecimento e classificação de padrões, com objetivo de comparar a taxa de detecção entre a abordagem proposta nessa tese com outras propostas.

1.1 Objetivos

Este trabalho tem por objetivo geral apresentar uma abordagem híbrida, em duas camadas, de um sistema de detecção e classificação de intrusão para redes de computadores.

Os objetivos específicos são:

- estudar a fundamentação teórica sobre wavelets e redes neurais artificiais;
- estudar as principais propostas de IDS;
- elaborar uma proposta de IDS híbrido;
- avaliar a proposta para dados obtidos de um simulador;
- avaliar a proposta para dados obtidos em uma rede em laboratório;
- avaliar a proposta para uma base de teste comumente utilizada para análise de IDS.

1.2 Justificativa

O número de ataques às redes de computadores tem crescido consideravelmente, nos últimos três anos o número de incidentes reportados, no Brasil, cresceu quase 100% (Centro De Estudos Resposta E Tratamento De Incidentes De Segurança No Brasil, 2009). Neste cenário, é fundamental a disponibilização de recursos para garantir um nível mínimo de segurança.

Os sistemas de detecção de intrusão são ferramentas que contribuem para melhorar a segurança em sistemas ou redes de computadores. Sua implementação é parte de uma política de segurança e é fundamental para manter os serviços ativos providos pelas redes.

Parte das propostas de construção de IDS utiliza apenas uma única técnica para reconhecimento e a classificação dos ataques. A abordagem híbrida permite melhor aproveitamento dos recursos computacionais, pois uma camada mais simples poderá identificar uma situação de risco enquanto que a segunda, mais complexa, realizar sua classificação, contribuindo para escolha de ações para anular ou minimizar os danos causados por ataques.

1.3 Metodologia

Na primeira fase deste trabalho foi desenvolvida a pesquisa teórica, onde são tratados os conceitos envolvidos em redes de computadores, segurança da informação e a implementação de sistemas de detecção de intrusão. Foram avaliados alguns trabalhos relacionados escolhidos de acordo com o número de citações em bases de pesquisas (periódicos e anais de eventos).

Na segunda fase foi desenvolvido um protótipo do sistema de detecção de intrusão em duas camadas, baseado em transformadas wavelets e redes neurais artificiais. O protótipo foi avaliado através de uma base de testes bastante utilizada em análise de IDS. Além disso, os resultados obtidos na avaliação da proposta aqui apresentada foram comparados com resultados de outras abordagens empregadas na construção de sistemas de detecção de intrusão.

1.4 Contribuições

A principal contribuição é a abordagem combinada de duas técnicas distintas para construção de um IDS. Com essa proposta, é possível utilizar uma metodologia que consome pouco poder computacional para avaliar a situação geral da rede e, se ocorrer alguma ação maléfica, um segundo método será utilizado para classificar a ação, neste caso, sendo necessário utilizar maior recurso computacional.

Algumas contribuições secundárias podem ser citadas, tais como revisão bibliográfica contendo o estado da arte sobre segurança em redes de computadores; revisão bibliográfica sobre os sistemas de detecção de intrusão; comparação entre algumas propostas de construção de IDS.

1.5 Estrutura da Tese

O restante desse texto está organizado da seguinte forma:

- capítulo 2: é realizada uma revisão sobre redes de computadores. São analisadas as principais arquiteturas, além da integração com redes de telefonia, através das redes convergentes. O capítulo ainda inclui uma descrição sobre os principais aspectos da família de protocolos TCP/IP;
- capítulo 3: é apresentado nesse capítulo os conceitos e definições matemáticas sobre o funcionamento das transformadas de wavelet. Também está presente no capítulo uma descrição sobre as redes neurais artificiais, bem como alguns aspectos relacionados ao seu treinamento e uso;
- capítulo 4: é descrito nesse capítulo os conceitos relacionados aos sistemas de detecção de intrusão. Contém ainda resultados sobre a taxa de detecção de algumas abordagens propostas;
- capítulo 5: é exposto nesse capítulo a proposta de implementação de um sistema de detecção de intrusão baseado em wavelets e redes neurais, como a principal contribuição desse trabalho. Também é apresentado nesse capítulo os resultados obtidos, bem como uma comparação com outras propostas e outro método de classificação.
- capítulo 6: é apresentado nesse capítulo as principais conclusões bem como sugestões de trabalhos futuros.

2 REDES DE COMPUTADORES

2.1 Introdução

A comunicação de dados é parte fundamental de um sistema de computação. As redes de computadores reúnem dados sobre os mais variados assuntos, desde condições atmosféricas a jogos de computadores. As empresas utilizam as redes para compartilhar dados de seus clientes, independente da localização geográfica de suas unidades. Com o desenvolvimento das fibras ópticas, houve contribuição para que as redes pudessem ser conectadas entre si, com apoio à criação de uma grande rede com abrangência mundial. As redes podem ser entidades autônomas, sem dependência de outras redes, com disponibilidade de serviços de comunicação para um determinado grupo de usuários.

A área de redes de computadores é uma das quais mais se desenvolvem tecnologias que visam, primordialmente, prover mecanismos e técnicas para comunicar e integrar as diversas regiões, empresas e cidadãos do mundo. Essas tecnologias utilizam vários meios de transmissão, também proporcionam formas de comunicação rápida, segura e eficiente.

2.2 Redes de Computadores

Uma rede de computadores é o conjunto formado por módulos processadores capazes de trocar informações e compartilhar recursos, interligados por um sistema de comunicação, como é mostrado na Figura 2.1. Módulo processador é qualquer dispositivo capaz de se comunicar através do sistema de troca de mensagens, como por exemplo, um microcomputador, uma máquina copiadora ou um computador de grande porte. O sistema de comunicação é constituído de um arranjo topológico com conexão dos módulos processadores através de enlaces físicos (meios de transmissão) e de um conjunto de regras, a fim de organizar a comunicação (protocolos) (Soares, Lemos *et al.*, 1995).

Em um sentido mais restrito, uma rede de computadores é um conjunto autônomo de equipamentos interconectados (Tanenbaum, 1997). Dois computadores estão interconectados quando podem trocar informações. A conexão pode ser feita através de fio de cobre, fibra óptica, enlaces de rádio, satélite ou por

qualquer outro meio de comunicação. O conceito de computadores autônomos exclui os sistemas em que existe a nítida relação mestre/escravo, como o uso de terminais remotos. Se um computador tiver o poder de iniciar, encerrar ou controlar outro computador haverá uma clara indicação de que não há autonomia entre eles. Um sistema com uma unidade de controle e muitos escravos ou um grande computador com terminais e impressoras não formam uma rede de computadores.

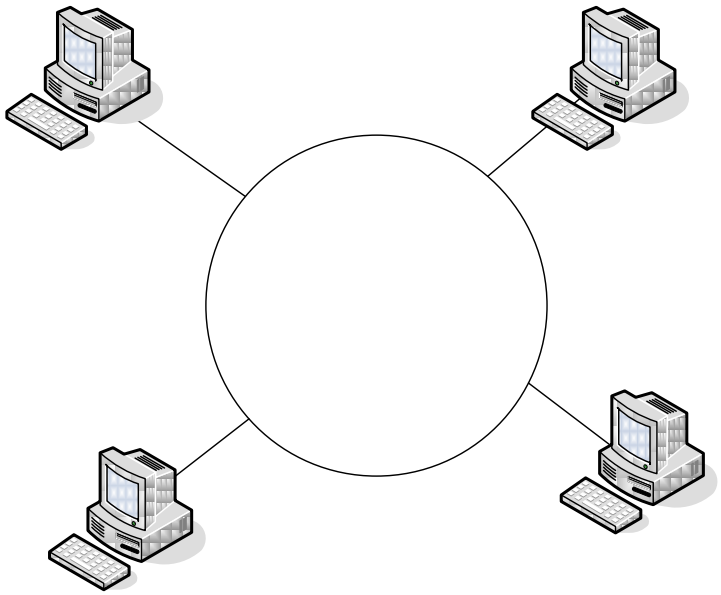


Figura 2.1 - Rede de computadores

O conceito de redes evoluiu muito desde as concepções iniciais. Tinha-se como modelo a separação física e lógica da rede de computadores, telefonia e vídeo. Neste momento, existe um movimento na direção da convergência entre as várias tecnologias de comunicação, bem como os serviços que estas propiciam para os usuários. É possível a utilização de uma única rede (considerando recursos de gerenciamento de serviços e prioridade de tráfego de determinadas aplicações), para transportar todos os dados da comunicação. Percebe-se também a existência de relativa facilidade no gerenciamento desta rede quando convergem as tecnologias de voz (seja através de PABX, sistemas híbridos ou totalmente IP), dados (transmissão de informações dos sistemas corporativos, e-mail, documentos e outros), vídeo (videoconferência) e sistemas de segurança (alarmes, sensores de portas e janelas e sensores de presença).

Existem várias tecnologias que podem ser utilizadas para agregar todas as comunicações. Estas podem ser através de uma rede privativa (geralmente fornecido por uma operadora de serviços, *Frame Relay*, ATM, VPN MPLS) ou pela Internet (com a formação de VPN). As operadoras de telecomunicações comercializam serviços para formação de redes através do aluguel de acessos e transporte de dados através de seus *backbones*.

A Internet é uma rede que permite que sua infra-estrutura seja utilizada para interligação de outras redes. Uma topologia simples é baseada na construção de túneis criptografados, com a formação de VPN, ou simplesmente disponibilizando o acesso a aplicações pela Internet, como ilustrado na Figura 2.2.

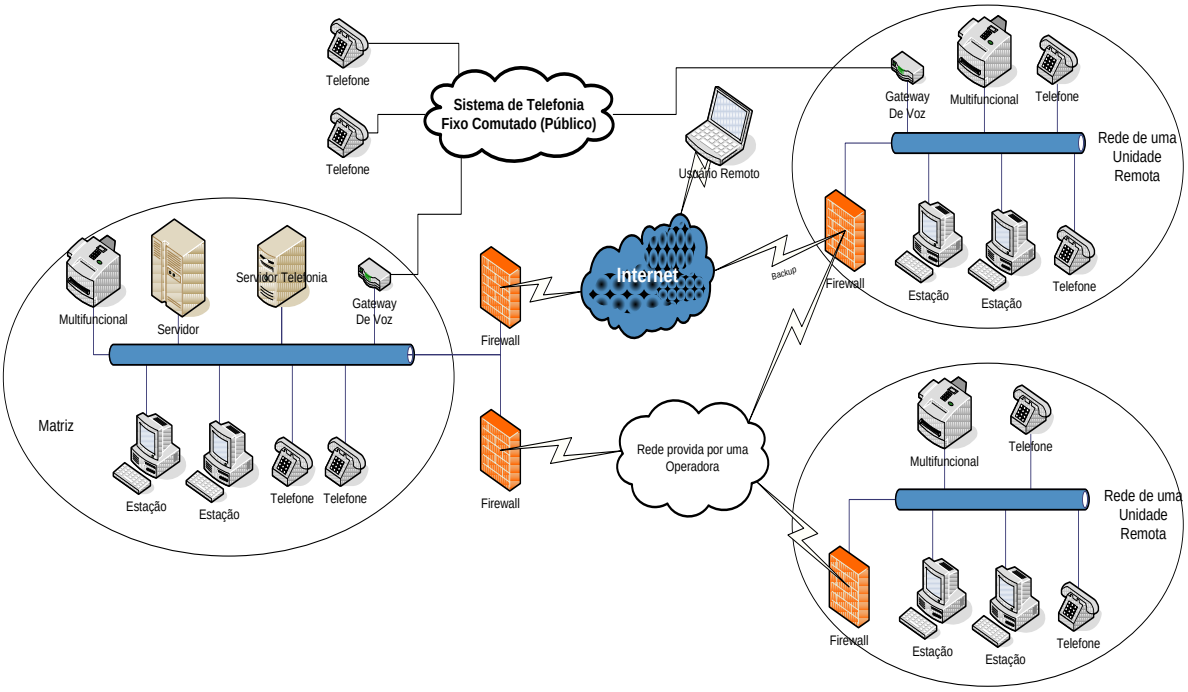


Figura 2.2 - Redes Convergentes

Os *backbones* de redes são um conjunto de equipamentos capazes de transmitir e processar dados em alta velocidade e é responsável pelo tráfego de informações entre redes menores. A interligação de redes locais aos backbones é realizada através de redes de acesso. As redes de acesso podem ser constituídas de meios guiados (baseados em fibra óptica ou fios de cobre) ou por meio de enlaces de rádio.

2.3 Família de Protocolos TCP/IP

No final dos anos 60, o Departamento de Defesa dos Estados Unidos, autorizou a pesquisa e o desenvolvimento de uma rede com objetivo de interligar os vários centros de pesquisa. Na ocasião, com a guerra fria, existia interesse no desenvolvimento de uma infra-estrutura com funcionamento independente de nós, mesmo se alguns centros sofressem ataques e fossem destruídos. O protótipo chamado ARPANet foi desenvolvido, porém apresentava problemas de estabilidade com constantes quedas. Assim, foi iniciada a pesquisa para criar um conjunto de protocolos mais confiáveis; o trabalho terminou em meados da década de 1970 com o desenvolvimento do TCP/IP (Comer, 1995).

O modelo de referência TCP/IP não especifica as camadas mais baixas de rede (física e enlace), uma visão geral desta família de protocolos é apresentada na Figura 2.3. É responsabilidade da camada inter-rede a interligação dos diferentes tipos de redes. Quando ocorre a transmissão de pacotes entre redes, o protocolo IP, é responsável pela transmissão. Deve ainda garantir que os dados sejam transmitidos, independente do destino. O protocolo IP pode entregar os pacotes fora da ordem original em que eles foram criados pelo emissor. A rede pode utilizar-se de diferentes caminhos para entregar os pacotes a um mesmo destinatário, dessa forma, as camadas de níveis superiores no destinatário, são responsáveis por colocar em ordem os pacotes recebidos. A tarefa principal da camada de rede é realizar o roteamento para entrega dos pacotes.

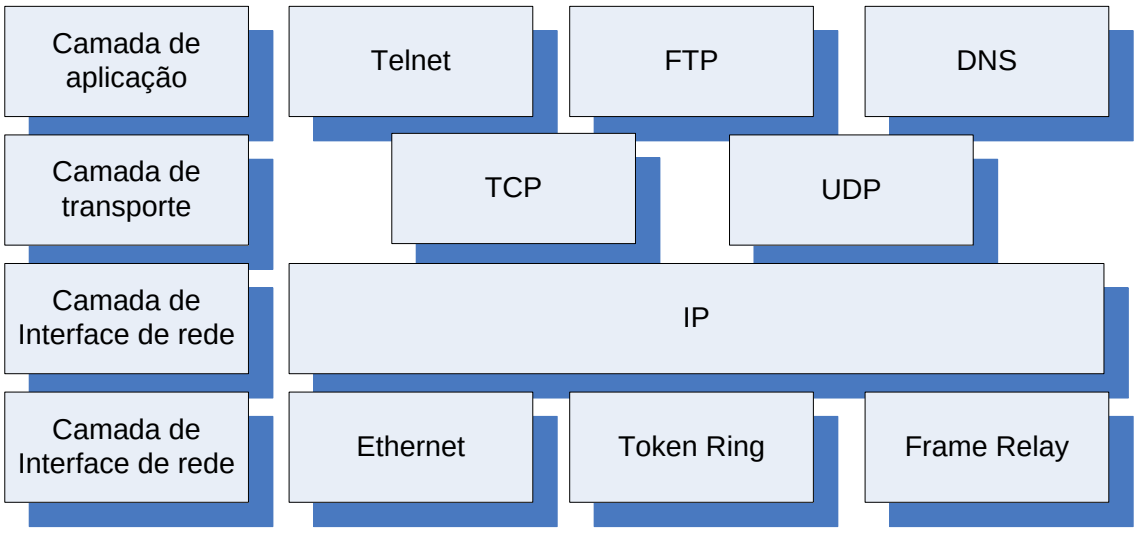


Figura 2.3 - Modelo de referência TCP/IP

As funções da camada de transporte no TCP/IP são semelhantes às funções apresentadas pelo nível de transporte do RM-OSI. Esta camada é definida pela utilização de dois protocolos: TCP e UDP.

O TCP permite a entrega dos pacotes sem erros, através de um fluxo de bytes em um canal originado no transmissor para o receptor. É um protocolo orientado à conexão. Possui funções de controle de erro, controle de fluxo, sequenciamento e multiplexação do acesso. Antes de realizar a transmissão de dados, recebidos da camada de aplicação, o TCP precisa estabelecer uma conexão entre os dois hosts envolvidos na comunicação. A conexão TCP não é um circuito TDM ou FDM fim a fim, e também não é um circuito virtual, pois o estado de conexão reside inteiramente nos dois sistemas finais (Kurose e Ross, 2006).

O cabeçalho do segmento TCP é constituído pelos campos: números de porta origem e destino, número de sequencia, número de reconhecimento, comprimento do cabeçalho, *flags*, janela de recepção, valor para checagem de dados, identificação para dados urgentes e um campo opcional de opções.

Os flags são constituídos de seis bits assim distribuídos:

- ACK: indica a confirmação de recebimento de um, ou vários, segmentos;
- RST: indica a intenção do emissor em abortar, de forma abrupta, a conexão;
- SYN: utilizado para estabelecer uma conexão;
- FIN: indica a intenção do emissor em finalizar, de forma normal, a conexão;
- PSH: indica que o destinatário deve enviar imediatamente os dados para a camada superior; e
- URG: indica que existem dados no segmento que foram marcados, pela camada superior, como urgentes.

O estabelecimento de uma conexão TCP inicia-se com a solicitação pelo cliente, através de um segmento especial, com o *flag* SYN ativado, para o servidor. Se o servidor aceitar a conexão, após fazer a alocação de *buffers* e variáveis do TCP, ele também retorna para o cliente um segmento com o *flag* SYN ativado. Por fim, o cliente confirma para o servidor, com um segmento com o *flag* ACK ativado. Foram omitidas desta explicação itens como número de sequencia e número de reconhecimento. Este procedimento utilizado pelo TCP é denominado de “aperto de

mão” de três vias. Ao término, os hosts possuem uma conexão e estão prontos para transmissão de dados oriundos da camada superior.

O protocolo UDP não é orientado à conexão, também não possui garantia de entrega de datagrama, portanto é mais rápido do que o TCP. É utilizado em aplicações que necessitam de entrega imediata de dados, não necessariamente de entrega precisa, como transmissão de voz e de vídeo (Tanenbaum, 1997).

A camada de aplicação possui protocolos de alto nível para permitir a utilização pelos usuários. Algumas aplicações utilizam o mesmo nome dos protocolos (HTTP, HTTPS, FTP, DNS e SMTP).

2.4 Padrões IEEE 802.11

As redes tradicionais, baseadas em cabos, restringem a mobilidade de seus usuários, é impossível tentativa de movimentação de equipamentos, quando estes estão presos pela infraestrutura física. As redes sem fio propiciam mobilidade para os usuários de notebook, ou PDA, que não poderia existir caso eles estivessem fazendo uso de uma rede com fios. As redes sem fio podem ser divididas em dois grupos: com infraestrutura e sem infraestrutura. No primeiro grupo, toda comunicação é feito através de um ponto concentrador, como acontece com as redes de comunicação celular. No segundo grupo, os nós da rede comunicam-se diretamente, sem a presença de um ponto concentrador; estas redes são chamadas de ad hoc. A comunicação pode ainda ser direta entre os nós vizinhos, ou por múltiplos saltos, neste caso, os nós também funcionam como roteadores na rede.

Os padrões para as redes sem fio são especificados pelo IEEE. Os trabalhos da primeira versão da família 802.11, definido em (ANSI/IEEE Std 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (R2003), 1999), iniciou-se em 1990. Este modelo especifica a camada física com três alternativas de transmissão: FHSS, DSSS e infravermelho.

A faixa de frequência, alocada para os canais, sobrepõem canais próximos. Como mostrada na Figura 2.4, a especificação do padrão define até treze canais, porém, somente três podem ser utilizados simultaneamente, sem interferência entre eles (canais um, seis e onze). O canal seis sobrepõe os canais cinco, quatro, três e dois; além dos canais sete, oito, nove e dez.

Na literatura, encontram-se vários trabalhos relacionados à alocação de canais, como mostrado em (Luo e Shankaranarayanan, 2004), (Kauffmann, Baccelli *et al.*, 2005), (Nie e Comaniciu, 2006) e (Akl e Arepally, 2007).

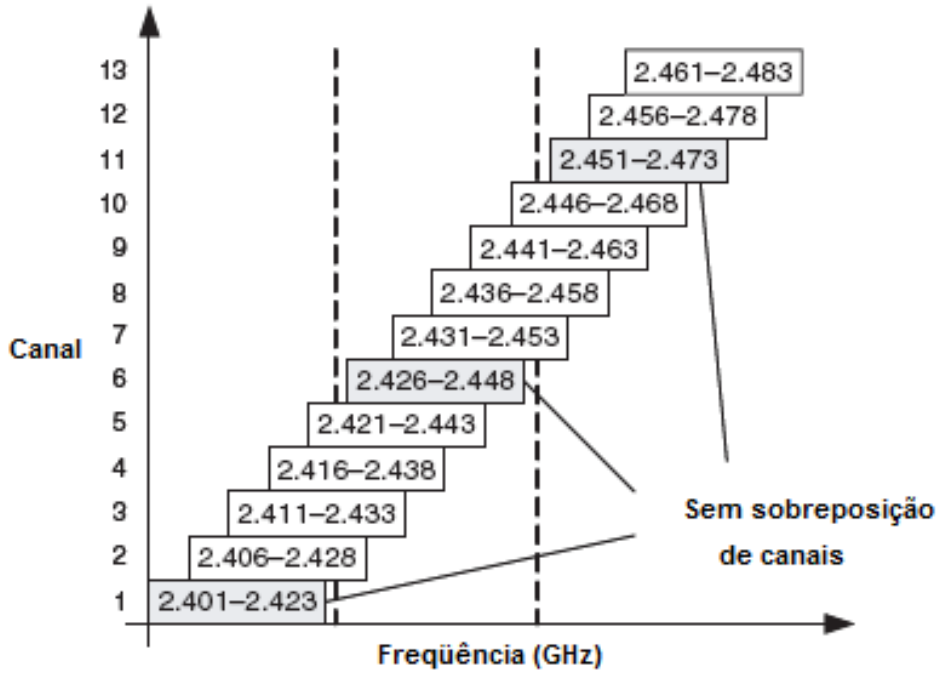


Figura 2.4 - Canais do padrão IEEE 802.11

De acordo com (Chen e Guizani, 2006), o padrão IEEE 802.11b é um suplemento do padrão original IEEE 802.11, também conhecido como Wifi ou “802.11 alta taxa” e possui suporte a taxa de transmissão de até 11Mbps.

Com as revisões do padrão IEEE 802.11, outros padrões foram desenvolvidos, entre eles IEEE 802.11b (ANSI/IEEE Std 802.11b, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the2.4 GHz Band (R2003), 1999), IEEE 802.11a (ANSI/IEEE Std 802.11a, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (R2003), 1999) e IEEE 802.11g (ANSI/IEEE Std 802.11g, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band (R2003), 1999).

A tabela 2.1 exibe as variações de padrões do IEEE, com as respectivas frequências e taxa máxima de transmissão. Alguns fabricantes comercializam equipamentos com capacidade maior de transmissão, porém esses valores não são

definidos pelo IEEE, logo são equipamentos que não seguem a padronização e não existe garantia de interoperabilidade com outras marcas.

Tabela 2.1 - Padrões IEEE 802.11

Padrão	***Faixa de Frequência em GHz***	***Transmissão Máxima em Mbps***	**Modulação**
IEEE 802.11	2,4 – 2,483	2	FHSS/DSSS
IEEE 802.11a	5,1 - 5,8	54	OFDM
IEEE 802.11b	2,4 – 2,485	11	DQPSK
IEEE 802.11g	2,4 – 2,485	54	OFDM

Os padrões definidos pelo IEEE especificam a taxa máxima de transferência, este valor é obtido em laboratório e em condições ideais. Porém, alguns itens podem reduzir consideravelmente este valor:

- obstáculos que degradam o sinal (paredes, campos eletromagnéticos, construções prediais);
- saturação do espectro (este tipo de acesso é compartilhado, com o aumento de número de usuários, a probabilidade de colisão na transferência de dados cresce); e
- interferência por outras redes (a existência de outras redes, na mesma faixa de frequência, pode degradar o sinal, até mesmo anulá-lo).

Os padrões 802.11 compartilham muitas características. Todos usam o mesmo protocolo de acesso ao meio, CSMA/CA; utilizam a mesma estrutura de quadros na camada de enlace; possuem capacidade de reduzir a taxa de transmissão para alcançar distâncias maiores, e também permitem modo de infraestrutura e ad hoc (Kurose e Ross, 2006).

A transmissão de dados, entre os nós, faz uso de um meio compartilhado. A camada MAC tem a função principal de controlar este acesso compartilhado pelos nós. O formato do quadro utilizado na subcamada MAC é um conjunto de octetos como mostrado na Figura 2.5. A descrição de cada campo é detalhada em (Kurose e Ross, 2006).

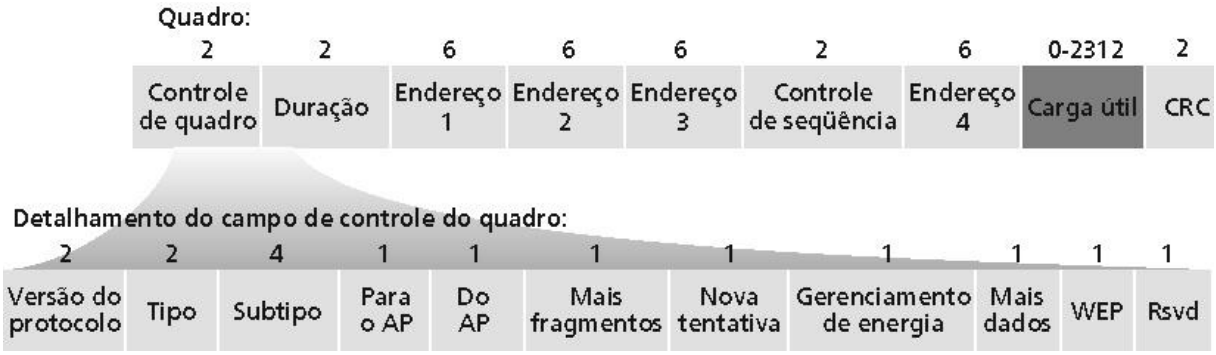


Figura 2.5 - Quadro da subcamada MAC, fonte (Kurose e Ross, 2006)

O mecanismo de controle provido pela subcamada MAC suporta dois métodos de acesso: distribuído e centralizado, com a possibilidade de ambos coexistirem. Os métodos de acesso determinam quando uma estação da rede tem permissão para utilizá-la.

Quando a decisão de transmissão é tomada de forma individual pela estação, a coordenação é distribuída. Isso poderá resultar em transmissões simultâneas, onde será necessária a retransmissão pelos nós, neste caso, o usuário poderá sofrer uma sensação de lentidão. O modelo centralizado para decisão de transmissão poderá reduzir a ocorrência das colisões. Ambos os métodos, a estação, quando desejar transmitir na rede, ela deverá primeiro ouvir o meio, e se estiver livre (sem outra estação transmitindo), o nó poderá iniciar o uso da rede.

A função de coordenação distribuída baseia-se no protocolo CSMA/CA para controle do acesso ao meio. As redes sem fio, em modo ad hoc, devem obrigatoriamente utilizar este método. Se o meio estiver ocupado, será necessário aguardar pela duração de DIFS, que é o tempo entre a transmissão dos quadros. Deverá então entrar numa fase de contenção, e tentar acessar o meio novamente após este intervalo aleatório de tempo. Caso o meio continue ocupado, todo este processo será repetido, inclusive a espera aleatória, como mostrada no esquema básico na Figura 2.6.

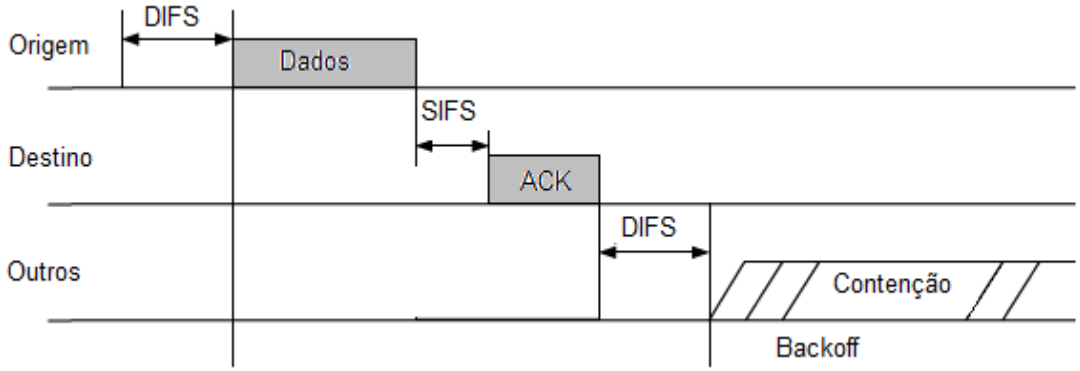


Figura 2.6 - Esquema Básico de Acesso no DCF

A estação destino confirma o recebimento de quadros sem erros através de ACK, após intervalo de tempo, chamado SIFS. Se o nó origem não receber o ACK, deduzirá que houve colisão, então iniciará a retransmissão e entrará no processo de retenção. Neste caso, aguardará um tempo aleatório, uniformemente distribuído entre zero e o tamanho da janela de contenção, com objetivo de evitar novas colisões. O uso apenas destes procedimentos não permite resolver o problema de terminais ocultos. De acordo com (Kurose e Ross, 2006), e como mostrado na Figura 2.7, este problema acontece porque o nós B e A ouvem um ao outro, assim como os nós B e C. Mas os terminais A e C não podem ouvir um ao outro, isto implica que não se dão conta da sua interferência em B. Se o nó A iniciar uma transmissão para B e no mesmo instante o nó C também iniciar uma transmissão para o nó B (pois o nó C não conseguiu identificar que o nó A está utilizando o meio) haverá colisão nas transmissões.

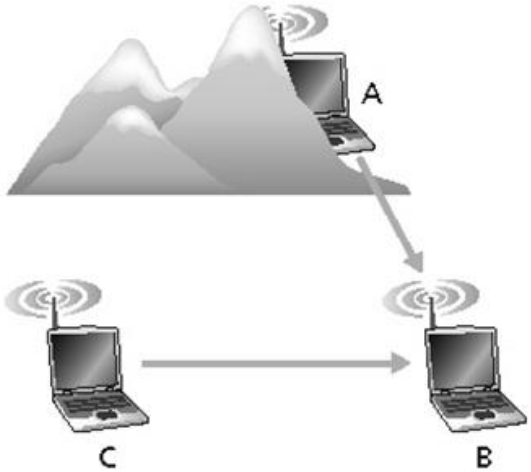


Figura 2.7 - Terminal Oculto, fonte (Kurose e Ross, 2006).

Com o objetivo de melhorar este cenário, foi desenvolvido um mecanismo opcional que envolve a troca de quadros de controle RTS *e CTS*. Uma estação, quando deseja transmitir na rede, envia um quadro de controle RTS, que informa uma estimativa de tempo da futura transmissão. A estação destino, em resposta ao quadro RTS recebido, envia um quadro de controle CTS com a indicação de que está pronta para receber os quadros de dados. Somente após a confirmação pelo nó destino, o emissor inicia a transmissão O quadro RTS possui funcionalidade de reservar o meio para transmissão e verificar se o destinatário está pronto para recebimento. Este processo é mostrado na Figura 2.8.

O padrão IEEE 802.11 também inclui uma função opcional, chamada Função de Coordenação Centralizada, que, diferentemente da DCF, é um esquema MAC centralizado onde um ponto de acesso elege, de acordo com suas regras, um terminal wireless para que este possa transmitir seu pacote.

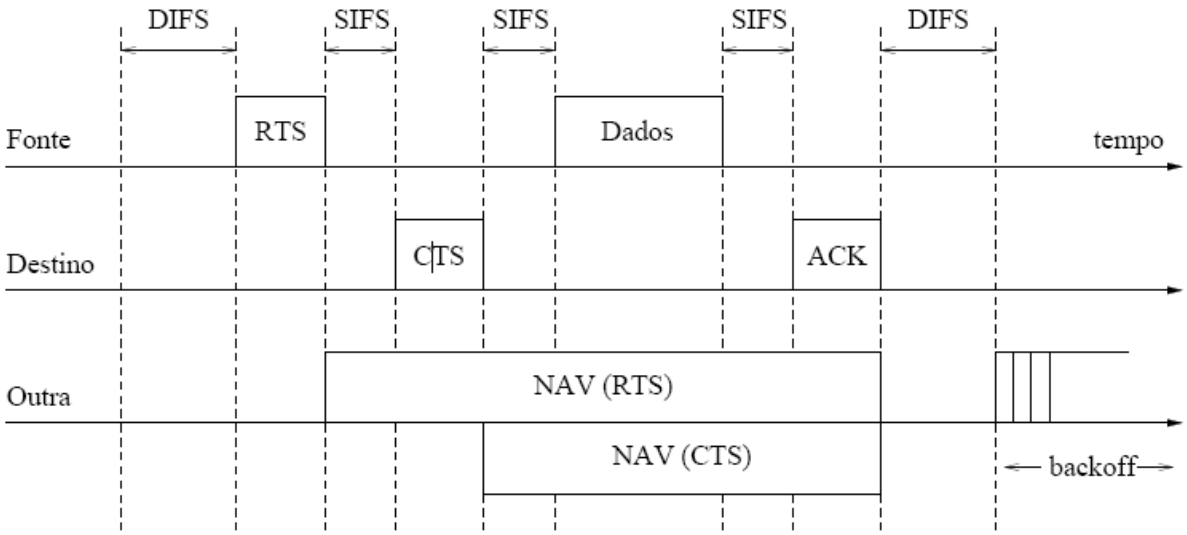


Figura 2.8 - DCF com uso de RTS e CTS

De acordo com a arquitetura apresentada em (Walke, Mangold *et al.*, 2006) e como mostrado na Figura 2.9, no modo de infraestrutura, o elemento BSS é responsável pelo gerenciamento dos nós da rede. Se um nó da rede encaminhar pacotes para seu vizinho, necessariamente, a transmissão passará e será controlada pelo BSS. No modo ad hoc, as estações são independentes e não possui infraestrutura centralizada, assim, em comunicações entre nós vizinhos, não será

necessário um elemento central fazer o encaminhamento de pacotes, desde que ambos os nós estejam acessíveis.

O padrão IEEE 802.11a provê melhorias para a camada física, mantendo as outras camadas intactas (Papadimitriou, Pomportsis *et al.*, 2003). As principais vantagens deste padrão é a diminuição de possibilidades de interferência, se comparado com o modelo 802.11. Outro avanço foi o aumento da taxa de transferência para o limite teórico de 54Mbps. Os equipamentos podem suportar taxa variável de transmissão (6, 9, 12, 18, 24,36 e 48Mbps) por fazer uso de técnicas diferentes de modulação. Este padrão utiliza 300MHz de largura de banda, na faixa de 5,4GHz. É variável também a potência máxima que pode ser utilizada. Por exemplo, os canais baixos (5,15 até 5,25GHz) operam com até 50mW, enquanto que os canais intermediários (5,25 até 5,35GHz) operam com potência até 250mW e os canais mais altos (5,725 até 5,825GHz) podem utilizar 1W de potência.

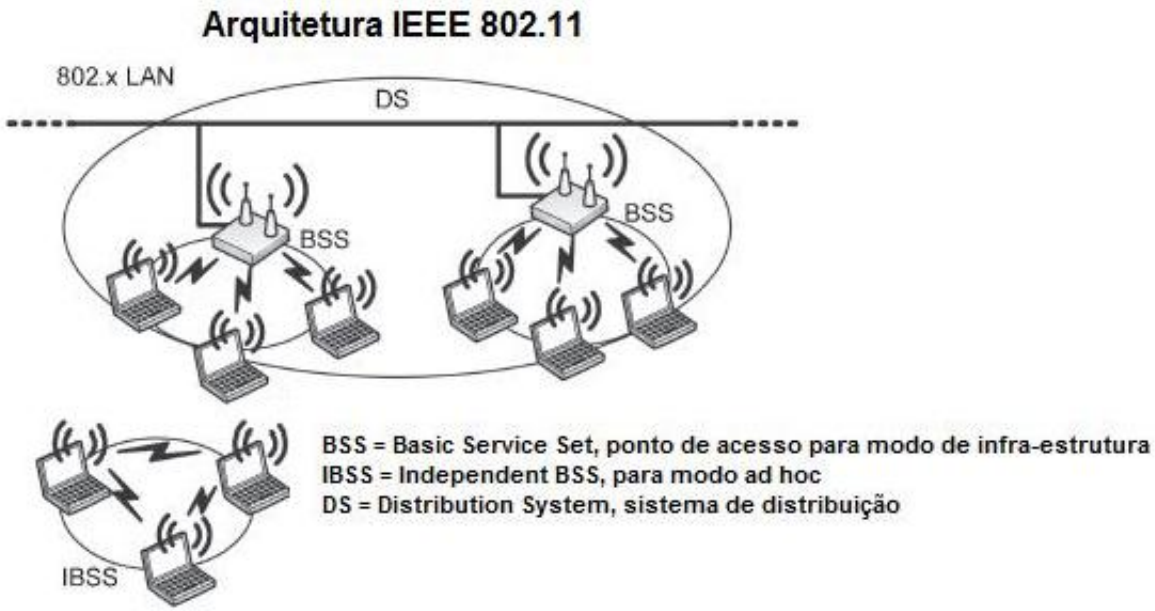


Figura 2.9 - Arquitetura IEEE 802.11, fonte (Walke, Mangold *et al.*, 2006)

O conjunto de várias portadoras de baixa velocidade forma um canal de alta-velocidade. O OFDM define oito canais (sem sobreposição) de 20MHz. Cada um destes canais é dividido em 52 subportadoras (aproximadamente 300KHz cada) que são transmitidas em paralelo. É utilizado o FEC, pois o overhead produzido não é expressivo se comparado com a banda disponível para transmissão. Como o OFDM

especifica baixa velocidade de transmissão de símbolos, a chance de interferência por propagação de múltiplos percursos também é pequena.

A especificação do IEEE 802.11b, como acontece com o IEEE 802.11a, também modifica apenas a camada física, quando comparado à especificação inicial. Permite taxas de 1; 2; 5,5 e 11Mbps e possui 14 canais (ou 11 canais em alguns países) com sobreposição. As taxas especificadas no IEEE 802.11b são variantes do IEEE 802.11 que faz uso do CCK, método de modulação baseado em códigos complementares binários. A taxa de 1Mbps é codificada em BPSK, enquanto que as outras três maiores em QPSK. O padrão IEEE 802.11g também opera na faixa de 2,4GHz e possui taxa de transmissão de até 54Mbps (Papadimitriou, Pomportsis *et al.*, 2003).

2.5 Redes Ad Hoc Sem Fio

Uma rede ad hoc é formada em situações quando nós móveis necessitam de comunicação enquanto uma infraestrutura fixa não é disponível ou não se deseja utilizar (Ramin, 2006). Neste caso, os nós móveis formam uma rede para uso temporário com objetivo de suprir as necessidades de comunicação naquele momento, ou ad hoc.

Uma rede móvel ad hoc (MANET) é um sistema de rede sem fio com nós móveis que podem mover-se livremente e são auto-organizáveis com topologia dinâmica e permite que equipamentos possam utilizar a rede sem comunicação pré-existente, diferente de uma rede com infraestrutura (Mohammad, 2003).

Como mostrado na Figura 2.10, as redes móveis ad hoc sem fio são comparadas, conceitualmente com redes de telefonia celular (se for excluída a estação rádio base). A rede de celulares, composta por nós móveis, permite que os usuários se movimentem livremente na área de cobertura. Nas redes ad hoc, também existe a mobilidade dos usuários, porém não é constituída por infraestrutura fixa. Cada elemento da rede também é responsável pelo encaminhamento de pacotes. Cada nó da rede é equipado com uma (ou mais) interface de rádio, e a cobertura da rede depende diretamente do alcance destas interfaces. As interfaces de rádio são responsáveis pela comunicação entre os nós.

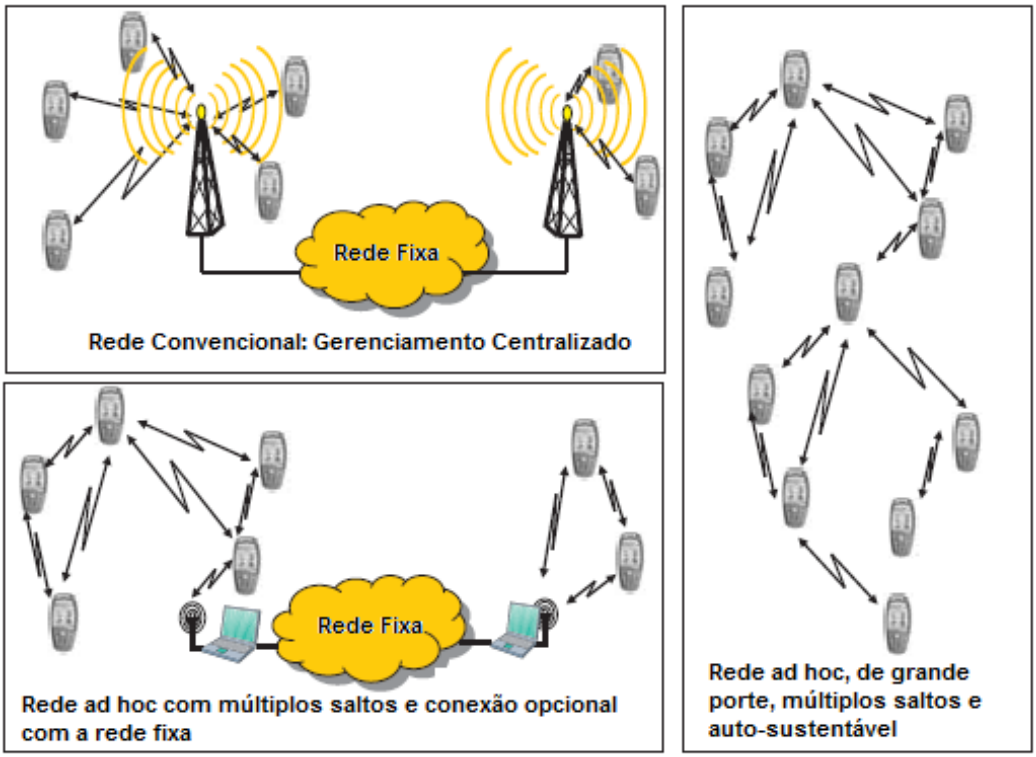


Figura 2.10: Tipos de Redes, fonte (Mohammad, 2003).

Os nós da rede podem funcionar, se necessário, como roteadores para outros nós, com o encaminhamento de pacotes para o destinatário final. Esta rede pode possuir conexão com rede fixa tradicional, através de gateways. De acordo com (Xiao, Hong *et al.*, 2006) as redes *ad hoc* consistem de uma coleção de pares de nós móveis que são capazes de se comunicarem sem a necessidade de uma infraestrutura fixa. As aplicações tradicionais deste modelo de rede são campos de batalhas militares, locais onde existe necessidade de formação rápida de redes devidas às catástrofes, missões de resgate, redes de sensores para automação e aplicações em eventos.

Existem, de acordo com (Prasant e Srikanth, 2005), certas vantagens quanto à utilização de redes ad hoc sem fio quando comparadas com as redes tradicionais: mobilidade dos nós, economia pela ausência de infraestrutura e menor consumo de energia devido aos enlaces curtos.

As desvantagens dessas redes é a complexidade para as estações, além do roteamento dos pacotes, os nós devem conter mecanismos de controle de acesso ao meio e resolver os problemas inerentes da tecnologia de rede sem fio: baixa taxa de transmissão, probabilidade de erro e grande variação das condições do meio de transmissão.

2.6 Redes em Malha Sem Fio

As Redes em Malha Sem Fio ou WMN é uma rede ad hoc com certas particularidades. Os nós que formam esta rede não possuem mobilidade, geralmente são instalados nos topos de edifícios ou nos telhados de residências. Como estes equipamentos são fixos, não são observadas restrições quanto ao fornecimento de energia aos nós. A rede possui capacidade de auto-organização, onde os equipamentos podem realizar roteamento, e teoricamente sem restrição de escalabilidade. Este modelo de rede é similar às redes "*peer to peer*" de compartilhamento (gnutella, napster, kazaa, freenet, metanet, waste).

De acordo com (Akyildiz e Wang, 2005), as WMN´s possuem capacidade de autoconfiguração, onde os nós da rede automaticamente estabelecem a conexão na forma de ad hoc e mantém a conectividade com a rede. Como na rede ad hoc não existe um elemento central e controlador, todos os nós podem trocar informações diretamente entre si, ponto a ponto. A topologia é constituída por três níveis ou camadas:

- internet ou rede externa: conexão com a Internet ou outra rede externa, exterior ao ambiente dos usuários. Contém roteadores de borda, ou servidores que realizam esta tarefa, podem-se utilizar enlaces de rede sem fio. Poderá haver mais de um circuito para conexão externa;
- *backbone*: conjunto que forma a infraestrutura principal da rede, ou espinha dorsal. Esta camada comporta todos os equipamentos que realizam a função de gateway ou roteamento e forma uma malha com topologia parcialmente ligada ou completa. Geralmente estes elementos da rede possuem pouca ou nenhuma mobilidade, fixados em determinados locais;
- clientes: conjunto de equipamentos utilizados pelo usuário final. Podem possuir mais de uma interface de rádio, e estas podem conectar-se a mais de um elemento concentrador. Não é necessário capacidade de roteamento. Permite uma variedade de equipamentos, desde sensores, estação móvel celular, PDA, notebook, roteador, AP, ou seja, qualquer equipamento de rede.

A Figura 2.11 apresenta um modelo de topologia para uma rede em malha sem fio, exibindo os principais componentes e os níveis de equipamentos. Em

termos práticos, uma rede em malha é híbrida, que permite a utilização em conjunto, de várias tecnologias de rede: fixa ou sem fio.

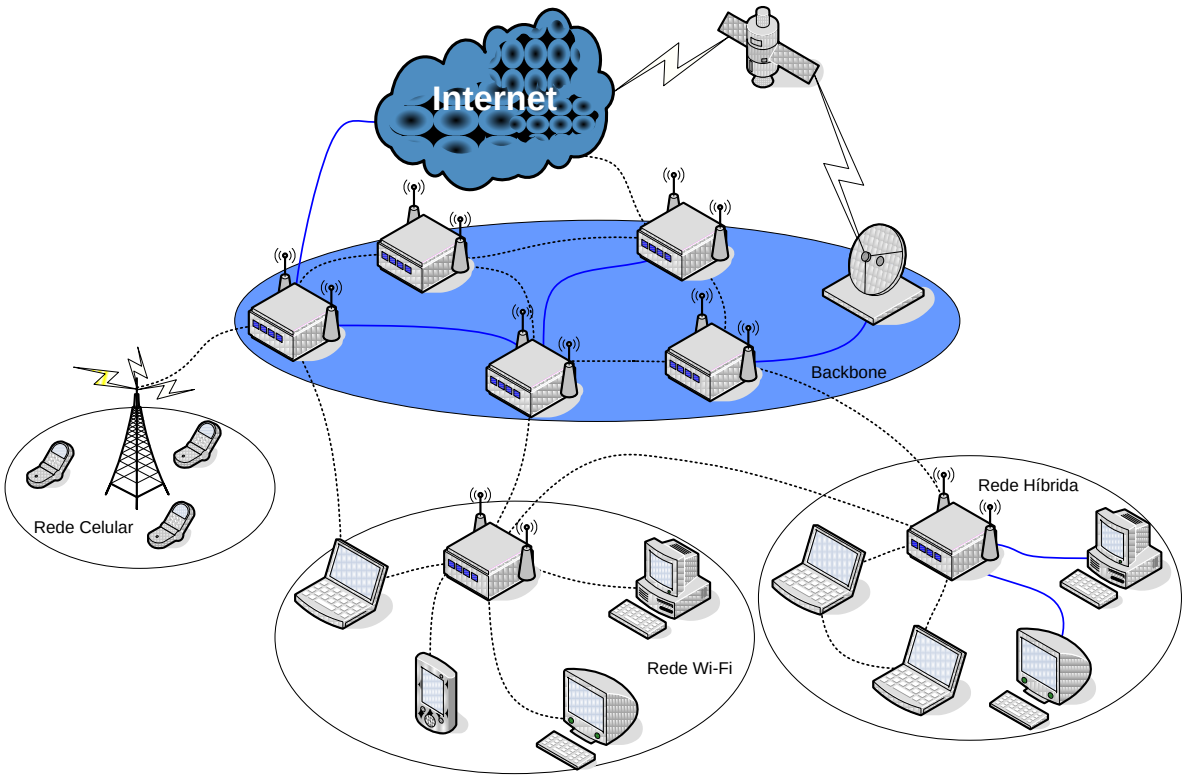


Figura 2.11 - Topologia WMN

Esta rede é baseada em enlaces de rádio. Assim, o desempenho WMN está baseado no desempenho das redes sem fio. De forma análoga, os problemas existentes na topologia de rede sem fio também são herdados pela rede em malha. Existem várias razões para o uso de uma rede em malha sem fio. De acordo com (Held, 2005), as principais vantagens são:

- confiabilidade: cada nó pode funcionar como um transmissor intermediário que encaminha os pacotes até seu destino final. Como o nós possuem capacidade de entrar e sair da rede, cada nó é capaz de, dinamicamente, mudar seu padrão de encaminhamento baseado em sua vizinhança. Assim, a topologia em malha aperfeiçoa a confiabilidade porque resultará na transmissão dos pacotes via enlace alternativo até seu destino;
- auto-organização: com a aprendizagem dinâmica de rotas e atribuição de endereços IP, os nós possuem capacidade para conectar a nós vizinhos.

Caso ocorra falha ou remoção de um nó, a topologia poderá ser rearranjada de tal forma que os serviços não sejam interrompidos;

- escalabilidade: os nós podem entrar e sair da rede enquanto executam um software compatível com outros nós da rede. Esta característica permite estender a área de cobertura de uma rede em malha, com a alocação de novos nós em localizações apropriadas onde eles possam se comunicar com os nós existentes da rede. Entretanto, a quantidade de nós possíveis de ser inserida é o limite máximo referente ao número total de nós que se possa ter numa rede sem degradar os serviços em função do número de saltos.

Existem também desvantagens deste tipo de rede. Esta rede herda quase todos os problemas intrínsecos das redes ad hoc sem fio, exceto os relacionados ao fornecimento de energia, visto que os nós geralmente são fixos.

2.7 Roteamento em Rede Ad Hoc Sem Fio

O roteamento é o processo que determina os caminhos ou rotas tomados pelos pacotes ao fluírem entre o remetente e o destinatário (Kurose e Ross, 2006). Os algoritmos que calculam esses caminhos são denominados algoritmos de roteamento. De acordo com (Prasant e Srikanth, 2005), o suporte para roteamento é um dos mais significativos desafios em redes ad hoc sem fio, é um fator crítico para as operações básicas de uma rede. As combinações das características dos nós da rede contribuem para a dificuldade em prover o roteamento. Os nós de uma rede ad hoc sem fio podem mover-se pela rede, sem qualquer tipo de controle; esta mobilidade resulta numa topologia dinâmica com rápidas mudanças com a causa frequente de falhas nas rotas. O protocolo de roteamento para esta rede deve adaptar-se dinamicamente a estas mudanças topológicas.

As transmissões em redes sem fio possuem banda inferior, se comparada com as redes fixas, o ambiente sem fio funciona como meio compartilhado. O protocolo de roteamento deve utilizar *overhead* mínimo no cálculo das rotas. Nas redes de sensores, os nós possuem bateria com energia limitada, o protocolo de roteamento também deverá considerar a limitação quanto à disponibilidade de energia.

De acordo com (Waharte, Boutaba *et al.*, 2006), as redes em malha possuem características que as diferencia de outros modelos, mesmo sendo considerada uma particularidade de uma rede ad hoc sem fio:

- topologia: as redes em malha possuem *backbone* sem mobilidade, ou nós com pouca mobilidade;
- tráfego: a predominância do tráfego é entre os nós móveis e o gateway da rede;
- interferência entre caminhos: existe a possibilidade de interferência de sinais entre os nós que fazem parte da rede e outras redes em funcionamento;
- diversidade de canais: a rede beneficia-se da possibilidade de utilização de diversidade de canais no processo de comunicação com outros nós.

Apresenta-se na Tabela 2.2 uma comparação das principais características entre as redes fixa e sem fio.

Tabela 2.2- Comparação entre Redes

Característica	***Rede Fixa***	***MANET***	***Rede de Sensores***	**Rede em Malha**
Topologia	Estática	Móvel	Estática	Estática
Tendência de Tráfego	Qualquer par de nós	Qualquer par de nós	Sensor ao sink	Nó móvel ao gateway
Interferências entre caminhos	Não	Sim	Sim	Sim
Capacidade do Enlace	Fixa	Variável	Variável	Variável
Diversidade de Canais	Não Aplicável	Não	Não	Sim

Existe grande número de publicações com propostas para protocolos de roteamento em redes ad hoc sem fio, alguns são mostrados na Figura 2.12. Os protocolos OLSR (Clausen e Jacquet, 2003), TBRPF (Ogier, Templin *et al.*, 2004), AODV (Perkins, Belding-Royer *et al.*, 2003) e DSR (Johnson, Hu *et al.*, 2007) são exibidos por já possuírem RFC.

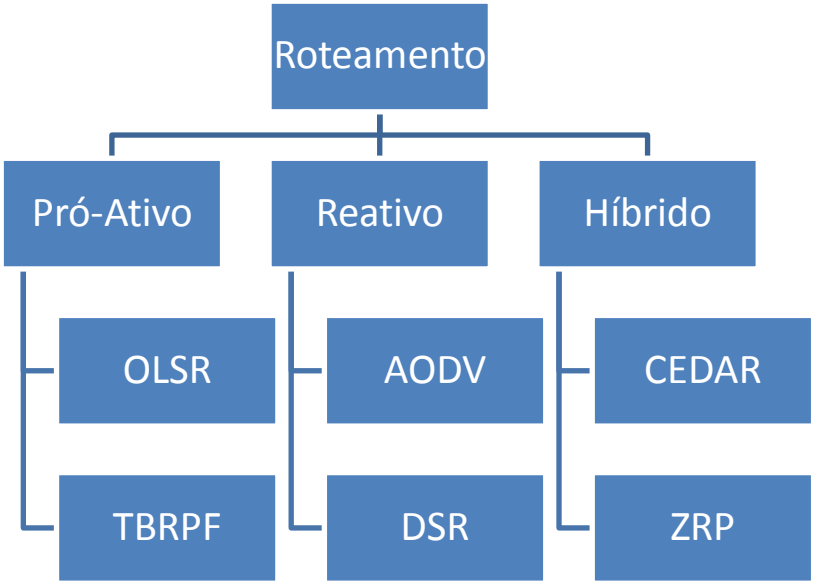


Figura 2.12 - Alguns Protocolos de Roteamento Ad Hoc

O protocolo OLSR é um protocolo de roteamento pró-ativo para redes ad hoc sem fio. Cada nó seleciona um conjunto de nós vizinhos como MPR e somente os nós MPR são responsáveis pelo encaminhamento de pacotes e controle de tráfego.

Os MPR's proveem mecanismo eficiente contra inundação, com redução do número de transmissões necessárias. A Figura 2.12 mostra inundação de pacotes numa rede wireless com múltiplos saltos, enquanto que a Figura 2.13 mostra a mesmo cenário com o uso de nós MPR.

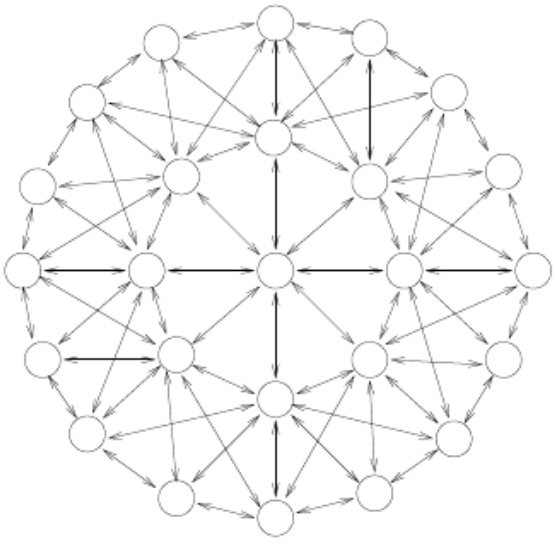


Figura 2.13 - Inundação de pacotes numa rede wireless

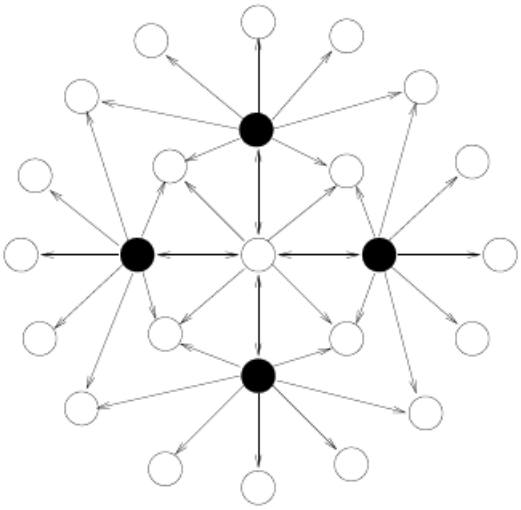


Figura 2.14 - Inundação de pacotes numa rede wireless com MPR

Os nós selecionados como MPR anunciam, periodicamente, mensagens de controles. Estas informações são utilizadas por outros nós MPR's para cálculo de rotas. A escolha de nós MPR's é baseada na premissa que ele consiga alcançar todos os nós de segunda origem, através do menor número de outros nós MPR's. Assim, o nó de origem deve alcançar qualquer nó a dois enlaces de distância.

A implementação padrão do OLSR cria rotas com métrica baseada em menor número de saltos. Esta alternativa não é a mais adequada para redes ad hoc sem fio. Mesmo com número reduzido de saltos, é possível existir enlaces congestionados, ou com banda disponível inferior a outros com número mais elevado de saltos. É mostrado na Figura 2.15 nós MPR e é indicada a banda do enlace e a utilização em percentual. Através do OLSR pode-se escolher a rota ADE, devido ao menor número de saltos, porém, a rota ABCE poderá ser mais eficiente.

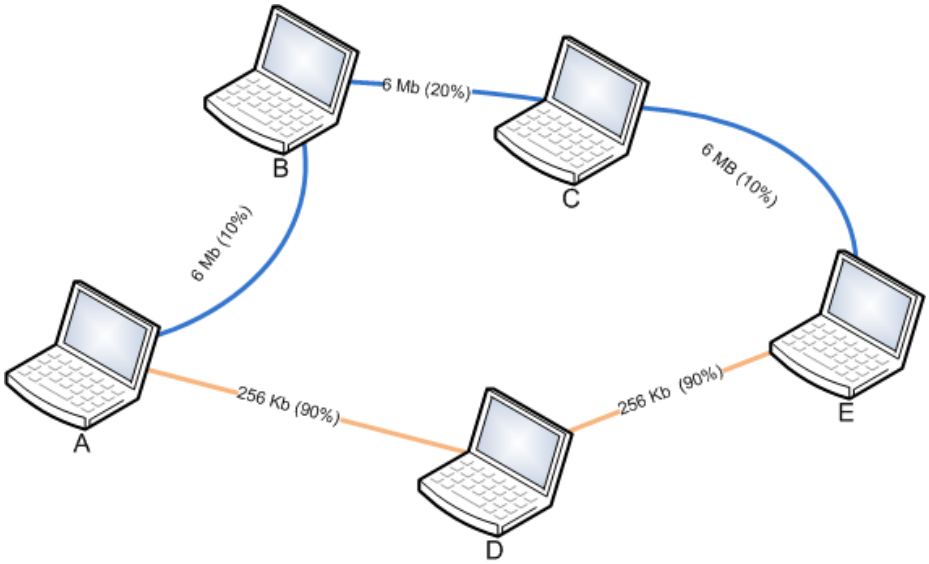


Figura 2.15 - Rede sem fio com indicação de banda

Existem propostas de melhoria ao OLSR. A proposta do OLSR-ML inclui métricas adicionais como o ETX (Saade, Albuquerque *et al.*, 2007), com o monitoramento da taxa de perda dos enlaces e a escolha de rotas baseadas também em enlaces com menor taxa de perda acumulada.

O TBRPF é um protocolo de roteamento que utiliza o estado dos enlaces. Cada nó possui apenas uma visão parcial da topologia da rede, mas suficiente para calcular rotas mais curtas. A topologia é armazenada numa árvore, com todos os caminhos alcançáveis pelo nó. A topologia parcial é construída a partir de uma modificação do algoritmo de *Dijkstra*. Para minimizar o *overhead*, cada nó publica apenas parte desta tabela para seus vizinhos (Prasant e Srikanth, 2005). Este protocolo utiliza uma combinação de atualizações periódicas e diferenciais para manter todos os vizinhos informados sobre a árvore de caminhos. Cada nó tem opção de disponibilizar informações adicionais sobre a topologia (topologia completa), para prover maior robustez a rede. A descoberta dos vizinhos é feita através de mensagens *HELLO*, e estas mensagens informam apenas as mudanças ocorridas nos vizinhos. Como resultado, estas mensagens são menores do que as utilizadas em protocolos como OSPF.

Quando um nó obtém a árvore que é mantida pelos nós vizinhos, ele pode atualizar sua própria árvore de topologia. Este protocolo explora o fato de que as árvores recebidas podem ter sobreposição, assim, o nó pode ainda calcular caminhos mais curtos mesmo recebendo árvores com topologia parcial da rede.

Cada nó divulga somente árvores em que ele é raiz chamada de RT, com o objetivo de reduzir o tamanho das atualizações. Frequentemente, os nós divulgam as alterações nas árvores com topologias parciais, num período mais longo, os nós também divulgam a RT completa. Para o cálculo do RN, o nó X determina um conjunto de RN. O RN é composto pelo próprio nó X e cada vizinho Y, com que o nó X é o caminho mais curto para Y. Para cada nó Y, incluído no RN, o nó X atua como encaminhador de pacotes para o destino Y. Finalmente, X também inclui no RN todos os nós que podem ser alcançados, pelo caminho mais curto, através de um de seus vizinhos que já estão no RN. Quando X completar o cálculo de RN, o conjunto formado por todos os enlaces (u, v) com $u \in$ RN constitui o RT do nó X. Percebe-se que RT apenas especifica um conjunto mínimo da topologia que o nó divulga para seus vizinhos (Bellur e Ogier, 1999).

O protocolo AODV é derivado de outros protocolos de roteamento, como o DSR e DSDV. O AODV faz uso do mecanismo básico de descoberta e manutenção de rotas do DSR e utiliza números de sequencia e pacotes periódicos como o DSDV. A vantagem do AODV sobre o DSR é que não é necessário incluir a rota na origem em cada pacote, assim é reduzido o *overhead*, mas as atualizações periódicas consomem parte da largura de banda economizada. A descoberta de rotas é o processo que inicia quando um nó deseja comunicar-se com outro nó. Cada nó mantém dois contadores SN e *broadcast-ID*. O nó origem inicia a descoberta de caminhos através de mensagens RREQ para seus vizinhos (Perkins, Belding-Royer *et al.*, 2003).

O par <endereço_origem, broadcast-ID> identifica um pacote RREQ. O broadcast-ID é incrementado quando o nó origem envia um novo pacote RREQ. Cada vizinho responde enviando um RREP se ele conhece a rota para o destino, como mostrado na Figura 2.16 (Hussain, Mahmood *et al.*, 2007).

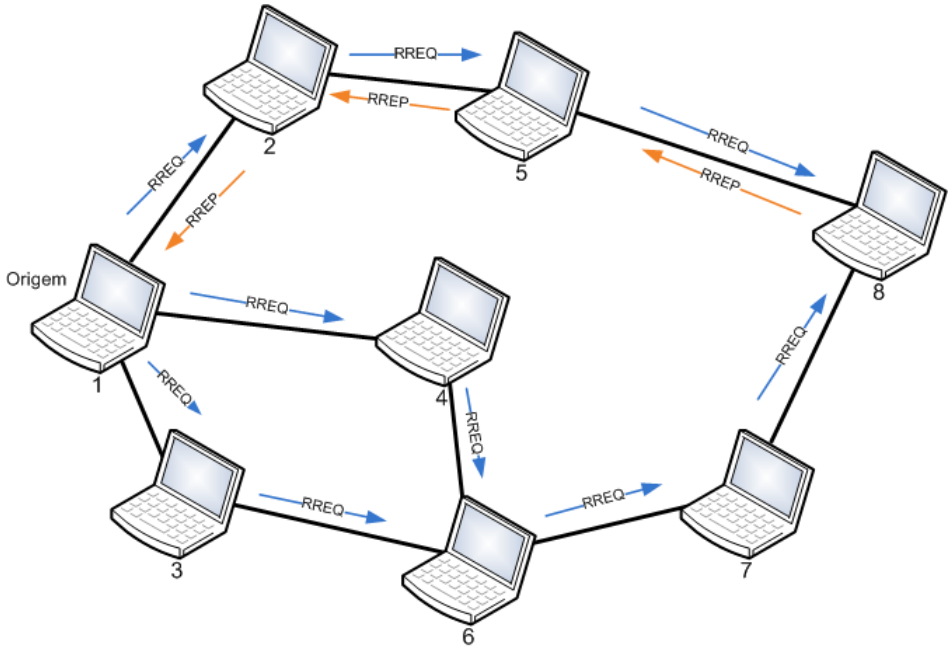


Figura 2.16 - Propagação de Mensagens RREQ e RREP do AODV

O nó pode receber múltiplas cópias do mesmo RREQ oriundas de vários vizinhos. Quando um nó intermediário recebe o pacote duplicado, com broadcast-ID e endereço do nó origem, o pacote redundante é descartado. Durante as transmissões de RREQ, o nó intermediário armazena em sua tabela de roteamento o endereço dos nós vizinhos, criando assim um caminho reverso. Desta forma, se um RREQ chegar novamente a este nó, por outro caminho, ele será descartado, prevalece o primeiro pacote recebido pelo nó. A tabela de roteamento é associada a um período de validade, caso a rota não seja utilizada neste período, ela será removida da tabela. Se houver movimentação do nó origem dentro da rede, ele enviará quadros de RREQ para descoberta de novas rotas. Caso o movimento seja de um nó intermediário, ele notificará seus vizinhos (mensagem RREP com métrica infinita), com objetivo de informar que a rota deve ser excluída. Esta mensagem é propagada pelos vizinhos até que o nó origem seja informado. Deverá então o nó origem iniciar um novo processo de descoberta de rotas com novo RREQ. O AODV utiliza como métrica o caminho mais recente e menor e é capaz de identificar loop na rede.

O protocolo DSR é baseado no roteamento pela origem, onde o nó origem especifica a rota completa até o nó destino, no cabeçalho do pacote, e cada nó intermediário simplesmente repassa o pacote para o nó vizinho que foi indicado. O DSR faz uso de cachê quando para armazenamento de rotas. Portanto, o nó origem,

quando deseja transmitir, primeiro faz a busca no cachê pela rota, caso a rota seja encontrada, o nó origem a utiliza, caso contrário, utilizará um protocolo para descoberta de rota (Mukherjee, Bandyopadhyay *et al.*, 2003). Os nós intermediários não necessitam manter informações atualizadas, pois eles somente realizam o encaminhado especificado nos cabeçalhos dos pacotes. Além disso, como o protocolo é reativo, é desnecessário anúncio periódico de rotas e detecção de nós vizinhos (Johnson, Maltz *et al.*, 2001). Quando um nó precisar descobrir uma rota, ele o fará através de mensagens RREQ. A Figura 2.17 demonstra o processo quando o nó origem A deseja descobrir a rota para o nó destino E. O nó A envia em *broadcast* a mensagem RREQ para E, que é recebida por todos os nós vizinhos de A. Cada mensagem RREQ contém o endereço do nó origem e destino, uma identificação (*Request Id*) e contém ainda a lista de todos os nós intermediários. Os nós intermediários encaminham uma cópia da mensagem RREQ com a inclusão de seu endereço (Johnson, Hu *et al.*, 2007).

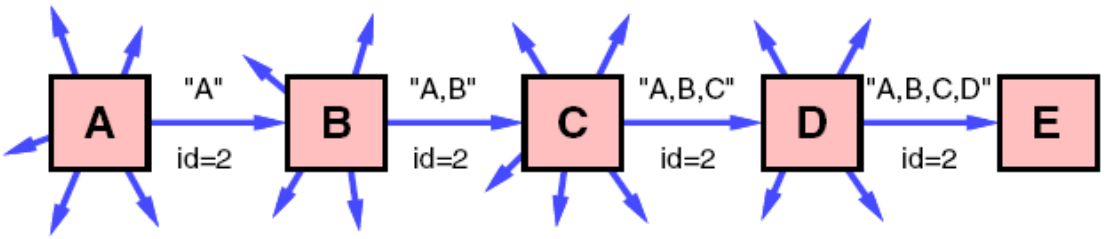


Figura 2.17 - Protocolo DSR, fonte (Johnson, Hu *et al.*, 2007).

Quando o nó destino recebe a mensagem RREQ, ele retorna uma mensagem para o nó origem com uma cópia de todo o percurso percorrido pela mensagem RREQ. Quando o nó origem recebe esta confirmação, ele armazena a rota em cachê.

O CEDAR foi projetado para pequenas e médias redes *ad hoc*, consistindo de dezenas a centenas de nós. A descoberta de rotas é de forma reativa. O CEDAR, de acordo com (Prasant e Srikanth, 2005) é formado por três principais elementos:

- nós principais (*Core Extraction*): é formado por um conjunto de nós, distribuídos e eleitos de forma dinâmica, para constituir o núcleo da rede. Os nós principais mantêm um conjunto de nós e são responsáveis pela sua topologia e pelo cálculo de rotas em seu domínio. Deverá haver um nó

principal a cada três saltos. O caminho entre dois nós principais é chamado de “caminho virtual”. Para a transmissão ser eficiente, cada nó principal precisa conhecer os nós principais vizinhos. Cada nó normal deverá ser capaz de encontrar um nó principal no máximo a um salto. Se um nó principal for desligado ou movimentar-se na rede, os nós normais que estavam anexados a ele precisarão encontrar um novo nó principal;

- propagação do estado dos enlaces: o roteamento com *QoS* no CEDAR é feito com a divulgação e propagação de banda disponível e informação sobre o estado dos enlaces dos nós dominadores. A ideia básica é que a informação sobre estes estados e disponibilidade de banda seja utilizada por nós distantes, enquanto que enlaces dinâmicos e com pouca banda sejam utilizados locais.
- cálculo da rota: o cálculo de rota primeiro estabelece um caminho no núcleo da rede, entre os nós principais da origem até o destino. Este caminho é utilizado pelo CEDAR para cálculo da rota adjacente que satisfaça os requisitos de *QoS*.

Como mostrado na Figura 2.18, o nó origem é o número 1, e o destino 15. Os nós principais são 3, 5, 11, 12 e 13. O nó 5 é o nó principal para os nós 1 e 6. O nó 12 é o principal do nó 15. O procedimento para transmissão é através do nó 5, que pesquisará sobre uma rota até o nó 12 que satisfaça os requisitos de *QoS*, caso não seja encontrada, o pacote será rejeitado (Kilinkaridis, 2007).

O ZRP tem por objetivo aproveitar as melhores características dos protocolos pró-ativos e reativos, com a combinação através de um protocolo híbrido. O esquema utilizado é a formação de uma zona, com os nós vizinhos. Os vizinhos locais formam a região chamada “intrazona” e utiliza protocolo de roteamento pró-ativo, enquanto que o roteamento externo a “interzona” é feita através de um protocolo reativo.

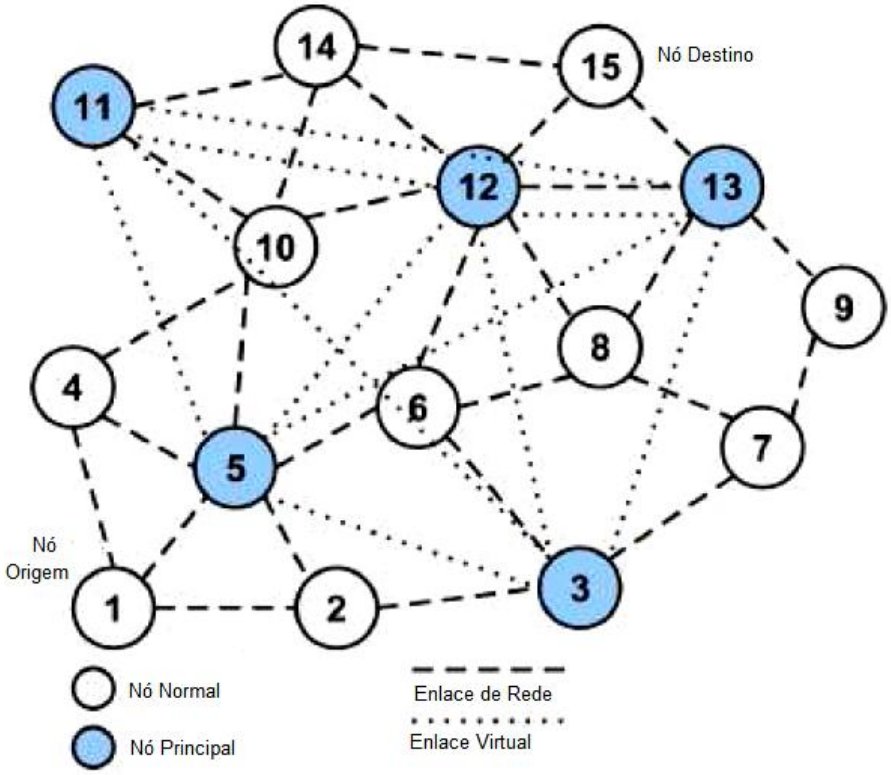


Figura 2.18 - Protocolo CEDAR, fonte (Kilinkaridis, 2007).

Cada zona de um determinado nó é um subconjunto da rede. Cada nó pode participar de uma ou mais zonas com tamanhos diferentes. O tamanho de uma zona é determinado pelo raio R (R é o número de saltos entre um nó até o perímetro da zona) (Kilinkaridis, 2007). Com a zona de raio 2, como mostrado na Figura 2.19, os nós B, C, D, E, F, H, I e J fazem parte da zona do nó A. Os nós C, G e I formam a intrazona, enquanto que B, D, E, G, H e J são nós periféricos. Cada nó mantém informações sobre suas rotas para os nós internos a zona, armazenados na tabela de roteamento. O ZRP não define qual protocolo pró-ativo os nós devem utilizar, assim, é possível existir zonas com diferentes protocolos.

De acordo com (Mukherjee, Bandyopadhyay *et al.*, 2003), para descobrir rotas exteriores a zona é utilizado mecanismo reativo. A descoberta de rotas requer pequeno número de mensagens que são encaminhadas apenas para os nós periféricos, omitindo todos os nós no interior das zonas, este método é chamado de "*bordercasting*".

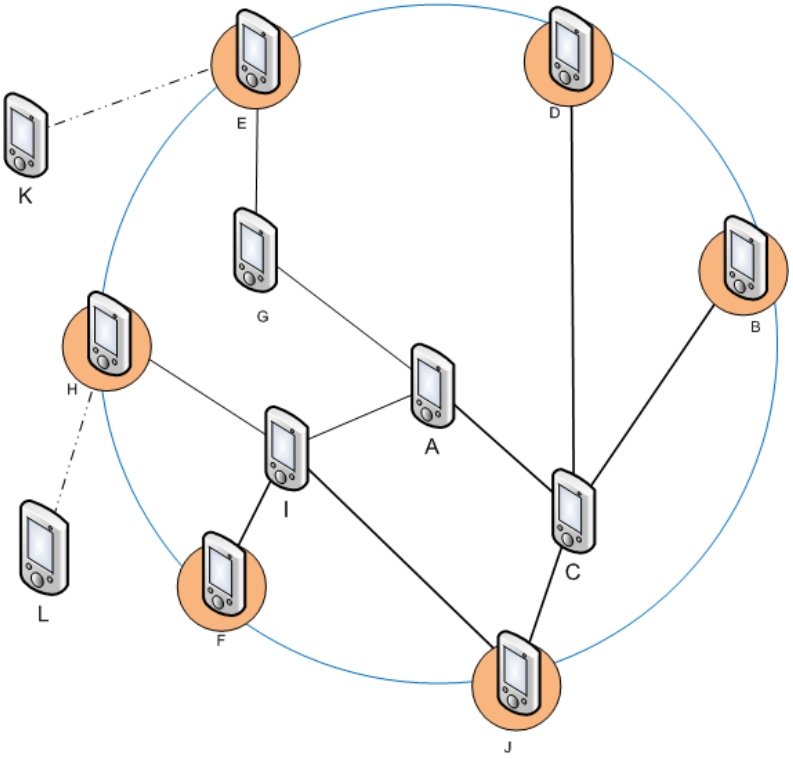


Figura 2.19 – Protocolo ZRP, fonte (Kilinkaridis, 2007).

A vantagem do ZRP é que reduz significativamente o *overhead*, se comparado a protocolos puramente pró-ativos. É necessário apenas que os nós conheçam a topologia da zona que eles pertencem.

Na tabela 2.3 apresenta-se um resumo de algumas características dos protocolos para rede ad hoc sem fio apresentados.

Tabela 2.3 - Comparação entre Alguns Protocolos para Rede Ad Hoc Sem Fio

Protocolo	***Tipo***	***Arquitetura-***	***Métrica***	***Suporte a QoS***	***Número da RFC***
OLSR	Pró-Ativo	Plano	Saltos	Não	3526
TBRPF	Pró-Ativo	Plano	Saltos	Não	3684
AODV	Reativo	Plano	Saltos	Não	3561
DSR	Reativo	Plano	Saltos	Não	4728
CEDAR	Híbrido	Hierárquico	QoS	Sim	Não
ZRP	Híbrido	Hierárquico	Saltos	Não	Não

2.8 Principais Ataques às Redes Ad Hoc Sem Fio

Todas as vulnerabilidades intrínsecas das redes sem fio são herdadas por este tipo de rede. Além disso, uma rede ad hoc possui também as vulnerabilidades específicas em função da tecnologia empregada, associadas principalmente a ausência de infraestrutura e ao encaminhamento colaborativo de mensagens.

O roteamento exige colaboração distribuída dos nós, significa que todos os nós participam do processo de roteamento. Estes nós estão sob o controle de usuários da rede, e não de administradores. Esta particularidade propicia a criação de ataques que visam explorar vulnerabilidades dos algoritmos cooperativos. A ausência de mecanismos centralizados impossibilita o emprego de técnicas usuais de autenticação (Fernandes, Moreira *et al.*, 2006).

Os enlaces de rádio são suscetíveis desde ataques passivos até interferências ativas. Diferente de redes tradicionais, onde o intruso precisa ter acesso físico à rede e precisa passar por vários mecanismos de defesas, como firewall e gateways, as redes ad hoc sem fio utilizam o ar como meio de transmissão, o atacante pode ter acesso diretamente à rede (Prasant e Srikanth, 2005).

De acordo com (Fernandes, Moreira *et al.*, 2006), os principais ataques às redes ad hoc podem ser divididos em dois grupos básicos: passivos e ativos. Os ataques passivos não interferem no funcionamento normal da rede, são caracterizados pela interceptação dos dados sem alteração dos mesmos. Ataques passivos são difíceis de serem identificados. A espionagem de dados é feita através do meio inseguro (transferência sem criptografia) com objetivo de roubar as informações dos usuários ou descoberta de elementos da rede, como por exemplo, a topologia. Para resolver este problema, podem-se utilizar recursos de outras camadas, protocolos que fazem uso de criptografia (HTTPS, SSH ou IPSec).

Os ataques ativos caracterizam-se pela criação, alteração ou descarte de dados em trânsito pelo atacante. Esta classe de ataques pode atuar em diferentes camadas do modelo RM-OSI. Os atacantes podem ainda ser internos ou externos. O primeiro é aquele que consegue fazer parte da rede e se passar por membros, em alguns casos podem ainda ser usuários autênticos da rede, enquanto que o segundo é formado por grupos que influenciam, mas não participam da rede.

Ataques ativos podem incluir a geração proposital de interferência. A interferência contínua entre canais é um ataque com baixo nível de complexidade. O

atacante utiliza-se de um ou vários nós que são configurados para utilizar a mesma faixa de frequência dos nós atacados, com objetivo de provocar erros nas transmissões. Para recuperar-se deste ataque, o administrador deverá observar qual equipamento está com o nível de sinal constante, e assim evitar rotas para aquele nó. Outra opção é fazer uso de analisador de espectro para encontrar o nó malicioso. O espalhamento espectral aumenta a tolerância do sistema a interferências.

Uma variação deste tipo de ataque é a interferência esporádica. O atacante utiliza o mesmo procedimento, mas em períodos aleatórios. Os nós atacados fazem maior uso de retransmissões, que também provoca o aumento do consumo da bateria. Neste caso, o usuário tem sensação de lentidão na rede. Por tratar-se de evento esporádico, a detecção é difícil.

Os nós de uma MANET são geralmente dispositivos com restrições de energia, como notebooks, celulares e sensores. O ataque por exaustão de bateria tem como objetivo consumir os recursos de energia através de geração de retransmissão continuamente, com modificações maliciosas na camada de enlace. Este método pode ser aplicado diretamente ao nó atacado, quando este iniciar uma transmissão, o atacante gera pacotes com objetivo de causar colisão. Nesta colisão, o nó atacado será obrigado e retransmitir o quadro. O tratamento para este ataque é difícil, pois envolveria modificações na camada de enlace (Wood e Stankovic, 2002).

O conluio de nós maliciosos principalmente para gerar problemas nos protocolos de roteamento, seja com loops, rotas falsas, caminhos não ótimos ou até mesmo encaminhamento seletivo é complexo de ser resolvido. De acordo com (Fernandes, Moreira *et al.*, 2006), esse tipo de ataque é de difícil detecção, pois para os nós comuns, o funcionamento está correto, embora, de fato, esteja apresentando anomalias.

Ataque bizantino possui este nome devido a fatos históricos acontecidos relatados sobre generais dos exércitos bizantinos. Os generais, distribuídos em campo de batalha, com suas tropas tinham a missão de organizar ataques aos inimigos. A comunicação entre os generais eram feitas por mensagens. Mas existiam generais traidores e com objetivos de confundir os demais, modificavam as mensagens com alteração de datas e horários dos ataques planejados. Neste tipo de ataque, um conjunto de nós tem procedimentos semelhantes aos generais corruptos, com a mudança de alguns dados dos pacotes transmitidos na rede. Para

garantir a veracidade das informações, pode-se fazer uso de assinatura digital. Outras formas de soluções são baseadas em roteamento seguro, ou outros que tentam minimizar as retransmissões, como proposta apresentada por (Burmester, Le *et al.*, 2007).

Os ataques com objetivo de proporcionar o estouro da tabela de roteamento baseiam-se em protocolos ad hoc pró-ativos, que armazenam as rotas anunciadas pelos nós vizinhos. Um nó malicioso publica rotas para nós não existentes. O atacante divulga um grande conjunto de rotas falsas com objetivo de aumentar a tabela de roteamento dos nós, até que ela estoure (Xiao, Shen *et al.*, 2007). As redes ad hoc, com nós com recursos reduzidos, como por exemplo, sensores, devido ao número elevado de mensagens, podem ser bastante prejudicados. Pode-se limitar o número máximo de valores na tabela de roteamento para solucionar este problema.

O ataque de replicação de pacotes possui como objetivo ocupar o meio de transmissão. O atacante faz uso de encaminhamento de cópias de pacotes de roteamento antigo. Para contornar este problema, pode-se fazer uso do número de sequencia para identificar que foram inseridos, na rede, pacotes que não são válidos. No entanto, este tipo de ataque consome recursos do meio de transmissão e do processamento de nós. Após identificar esta situação de ataque, pode-se ainda tentar isolar a parte da rede comprometida.

O ataque da pressa explora a maneira como os protocolos reativos constroem a tabela de roteamento. Quando um atacante recebe uma mensagem RREQ, este responde antes dos demais nós da rede, de forma que a rota para o nó destino passe por ele. Os protocolos reativos armazenam, na tabela de roteamento, somente a primeira resposta recebida, neste caso, a do atacante. As respostas dos outros nós serão descartadas. O nó atacante mantém-se numa posição privilegiada na rede.

A detecção deste ataque não é simples, pois o protocolo de roteamento considera este procedimento normal. Pode-se fazer uso de rotas múltiplas com objetivo de garantir que num ataque, algumas rotas estariam funcionando. Uma proposta neste sentido foi apresentada por (Oliveira, Wong *et al.*, 2006) para uso em rede de sensores.

O ataque por direcionamento falso faz uso do mecanismo de funcionamento de pacotes *ECHO*. O atacante envia grande quantidade de mensagens, mas com

pacotes modificados, com objetivo de redirecionar as respostas para um determinado nó da rede. O nó atacado, que é identificado como o emissor das requisições *ECHO* receberá grande quantidade de respostas. A utilização de IDS poderá identificar este tipo de ataque, então o administrador poderá bloquear o nó atacante.

Os protocolos que fazem uso de mensagens *HELLO* em geral também o utilizam para anunciarem-se aos vizinhos. Quando os nós vizinhos recebem estes pacotes concluem que ambos estão dentro do alcance do enlace de rádio. No ataque, chamado inundação por *HELLO*, o atacante faz uso de amplificação de potência e encaminha as mensagens com informação sobre rotas boas para outras redes. Os nós atacados incluem as informações sobre as rotas, mas quando tentam utilizá-las não conseguem, pois o atacante está fora de alcance, assim, eles permanecem com rotas para um nó inalcançável. Para evitar este ataque, os protocolos de roteamento devem verificar se os enlaces são bidirecionais.

No ataque por encaminhamento seletivo, o intruso não deseja prejudicar todos os nós, mas ele faz encaminhamento somente de alguns pacotes ou nós. É possível ainda que o nó malicioso realize a escolha de um determinado nó e não faz o encaminhamento de seus pacotes. Este tipo de ataque é difícil de ser detectado, pois enlaces ruins ou com interferência podem causar o mesmo sintoma. A utilização de rotas redundantes poderá minimizar o problema com a identificação do nó malicioso.

Um caso extremo deste tipo de ataque é o descarte de todos os pacotes pelo nó malicioso, também chamado de buraco negro. Se o nó atacante conseguir atrair muito tráfego, a proporção deste ataque pode ser considerável, pois ele provocará consumo elevado de recursos dos nós vizinhos. Um IDS deverá identificar este tipo de ataque na rede e então o administrador bloquear o nó atacante.

Em ataque tipo túnel de minhoca, dois nós da rede fazem conluio e criam entre si um túnel com uso de enlace de baixa latência. O objetivo é convencer os nós da rede que eles podem se comunicar com determinados destinos, através de um único salto, a utilizar múltiplos saltos da rede. Percebe-se que este ataque coloca os atacantes em posições privilegiadas e estes poderão fazer uso deste privilégio quando assim o desejarem. Uma possível solução para este problema é a utilização de protocolos de roteamento seguros, como ARAN ou Ariadne (Xiao, Hong *et al.*, 2006).

Os ataques por sequestro de sessão aproveitam-se de que muitas comunicações são protegidas somente no estabelecimento da sessão. No seqüestro de sessões TCP, o atacante captura os dados transmitidos e recebidos pela vítima para determinar o número de sequencia utilizado. Realiza em seguida algum tipo de ataque de negação de serviços no nó vítima e continua a utilizar a sessão previamente estabelecida. A utilização de criptografia nas mensagens trocadas poderá inibir este ataque.

O ataque Sybil, apresentado por (Douceur, 2002), foi descrito inicialmente para redes *peer to peer*, tem o nome inspirado em um caso ocorrido nos Estados Unidos, onde uma mulher sofria de múltiplas personalidades. O ataque ocorre quando um único hardware assume várias identidades em uma rede. Em redes ad hoc, o atacante pode criar identidades falsas para rotas diferentes, com objetivo centralizar várias rotas. O atacante pode ainda, utilizar-se das várias identidades para ser privilegiado quando ocorrer votações na rede (alguns protocolos de roteamento fazem uso de votação para escolha de nós centrais). Pode ainda, em redes baseadas em confiabilidade, realizar ações malignas através de algumas de suas identidades. Para defender-se deste ataque, (Newsome, Shi *et al.*, 2004) propõe a validação da identidade dos nós, presentes na rede, de acordo com seu endereço físico. Sugere dois métodos para isso: validação direta e indireta. A primeira forma, cada nó verifica se a identidade de outro nó é válida. O segundo tipo, após o nó ser validado, ele poderá testemunhar ou refutar a validade de identidade de outro nó.

As conexões em portas TCP ou exploração de portas UDP, podem ser utilizadas com objetivo de identificar quais serviços estão em execução. Podem ainda determinar ou estado de escuta. Este ataque é chamado varredura de porta. Os principais tipos de varreduras são apresentados a seguir:

- conexão TCP: o atacante conecta-se à porta do TCP da vítima e completa as três etapas da conexão (flags SYN, SYN/ACK e ACK). A identificação deste ataque pelo sistema alvo é relativamente fácil;
- TCP SYN: o atacante não realiza a conexão completa, em vez disso, envia apenas uma solicitação de conexão (SYN) para a vítima. Se a resposta for a confirmação do sistema alvo (SYN/ACK), deduz-se que a porta está em estado de escuta. A conexão completa nunca é estabelecida, e os buffers alocados neste procedimento ficam ativos até o “timeout” definido pelo TCP e

é mais difícil sua detecção, se comparada com a varredura completa do protocolo;

- TCP FIN: o atacante envia um segmento com o *flag* FIN ativado, a vítima deve responder com um segmento com o *flag* RST para cada porta fechada;
- Árvore de Natal (Xmas): o atacante envia um segmento com os *flags* FIN, URG e PUSH ativados, o sistema alvo deverá responder com um seguimento com o *flag* RST ativado para cada porta fechada;
- TCP NULL: neste ataque, é encaminhado um segmento com todos os *flags* desligados, o sistema alvo deverá responder com um segmento com o *flag* RST para cada porta fechada;
- TCP RPC: neste tipo de ataque, são identificadas todas as portas que estão no estado aberto, e são inundadas com o comando NULL do sistema de Chamadas de Procedimentos Remoto com objetivo de identificar portas RPC. Com esta informação, pode-se obter a listagem dos programas e versões que estão em execução ligados a estas portas;
- varredura UDP: devido à simplicidade do protocolo UDP, o atacante envia um pacote para cada porta que deseja verificar. Existem três resultados possíveis: Resposta de Porta UDP Inacessível (indica que a porta está fechada), Resposta UDP (indica que a porta está aberta) ou Sem Resposta (a porta pode estar aberta ou filtrada por firewall).

São apresentados na Tabela 2.4 os principais ataques às redes ad hoc, também é indicada a camada do modelo RM-OSI que os ataques pode ocorrer, além de um resumo sobre formas de combates.

Tabela 2.4 - Principais Ataques às Redes Ad Hoc

Nome	***Tipo***	***Camada***	***Solução***
Espionagem	P	Rede, Transporte ou Aplicação.	Utilizar protocolos seguros como HTTPS, SSH ou IPSec
Interferência contínua ou esporádica	A	Física	Utilizar rotas alternativas ou espalhamento espectral
Exaustão de bateria	A	Enlace	Modificação da subcamada MAC
Ataque bizantino	A	Rede	Assinatura digital

Estouro da tabela de roteamento	A	Rede	Limitar o tamanho da tabela de roteamento
Replicação de pacotes	A	Rede	Fazer uso do número de sequencia
Ataque da pressa	A	Rede	Roteamento seguro
Direcionamento falso	A	Rede	Detecção com uso de IDS e bloqueio do nó pelo administrador
Inundação de Hello	A	Rede	Utilizar roteamento com verificação de enlaces bidirecionais
Encaminhamento seletivo	A	Rede	Utilizar rotas redundantes
Buraco negro	A	Rede	Detecção com uso de IDS e bloqueio do nó pelo administrador
Túnel de minhoca	A	Rede	Roteamento seguro
Sincronização	A	Transporte	Limitar o número solicitação de conexões
Sequestro de sessão	A	Transporte	Utilização de criptografia
Sybil	A	Física, enlace e rede	Validar identidade através de endereço físico ou utilizar certificados digitais
Varredura de Portas	A	Transporte	Detecção com uso de IDS e bloqueio do nó atacante pelo administrador

(P = Passivo, A = Ativo)

O trabalho desenvolvido por (Panjwani, Tan *et al.*, 2005) determinou uma correlação entre varredura de portas e ataques. Os resultados experimentais mostraram que 50% de ataques não foram precedidos por varreduras. Contudo, a combinação com outros métodos (análise de vulnerabilidades, por exemplo), é um fator relevante que poderá indicar um possível ataque, com a varredura de porta, um estágio precursor de ataque. A identificação do ataque pode ser realizada através de

sistema de detecção de intrusos, e então o administrador poderá bloquear o nó atacante.

2.9 Comentários Finais

Com a popularização da Internet, a família de protocolos TCP/IP tornou-se padrão em rede de computadores. Isso gerou problemas relacionados à segurança, pois em seu projeto original, essa não era premissa principal destes protocolos, a preocupação do projeto era relacionada à disponibilidade da rede, mesmo sob a falha de alguns nós. Estes protocolos, aplicados diretamente em redes sem fio, herdam os problemas relativos à falta de segurança e também os problemas intrínsecos da tecnologia do IEEE 802.11a/b/g, causados devido à utilização do meio comum de transmissão de radiofrequência.

3 TRANSFORMADAS WAVELETS E REDES NEURAIS ARTIFICIAIS

3.1 Introdução

A wavelet é uma função capaz de descrever outras funções no domínio da frequência, tornando possível a análise em diferenças escalas de frequência e tempo. Além disso, permite analisar o comportamento dos dados de entrada identificando possíveis variações dos padrões de entrada, dessa forma, tornando-se boas candidatas para utilização em IDS. As redes neurais artificiais são tentativas de reprodução de funcionamento do cérebro humano para resolução de alguns tipos de problemas. Tem sido bastante utilizada para realizar tarefas relativas à classificação e reconhecimento de padrões. Nesse capítulo é apresentado o funcionamento básico das wavelets e redes neurais e suas principais propriedades.

3.2 Wavelet

De acordo com a análise de Fourier, um sinal, seja determinístico ou estocástico, pode ser decomposto em um conjunto de sinais regulares, com várias amplitudes e frequências. Com esta ferramenta, o sinal passa a ser analisado no domínio de Fourier (domínio da frequência). A decomposição em série evolui para a reescrita do sinal original via transformada de Fourier e as maiorias dos estudos envolvendo sinais fazem uso desta técnica (Oliveira, 2007).

A Transformada de Fourier de um sinal f(t), com $-\infty<t<+\infty$, e que seja a função integrável e finita, é descrito na Equação 3.1.

$$F(w) = \int_{-\infty}^{+\infty} f(t) e^{-jwt}\, dt \qquad \textbf{(3.1)}$$

De forma semelhante, através do espectro *F(w)* de um sinal, a inversa é possível, para obter o sinal no domínio do tempo, através da síntese de Fourier, como mostrado na Equação 3.2.

$$f(t) = \frac{1}{2\pi}\int_{-\infty}^{+\infty} F(w)e^{-jwt}\,dw \quad \textbf{(3.2)}$$

A transformada de tradicional de Fourier avalia a contribuição de cada componente no sinal como um todo. Além disso, também é analisado o sinal em todo o tempo de existência (de −∞ até +∞). Esta ferramenta não permite que seja criada uma janela no tempo para análise somente neste período. Com a visão global do sinal, não existe a possibilidade de verificar a comportamento de um componente num determinado instante. A aplicação de análise de sinais, com a transformada de Fourier, para detecção de comportamentos anômalos numa rede não teria efeito então, pois é possível identificar o componente da frequência, mas não em que tempo ela ocorreu.

Algumas propostas para análise de sinais não estacionários, com objetivo de ter informações em ambos os domínios (tempo e frequência) foram desenvolvidas. Uma das mais conhecidas foi a Transformada de Fourier de Tempo Curto (*Short Time Fourier Transform*), ou Transformada de Gabor.

A ideia básica da transformada de Gabor é introduzir um parâmetro de frequência local para análise da transformada de Fourier, desta forma, é analisada apenas uma janela limitada do tempo, onde a função permanece estacionária. O cálculo do espectro é mostrado na Equação 3.3.

$$SFFT(w,\tau) = \int_{-\infty}^{+\infty} f(t)W^{*}(t-\tau)e^{-jwt}\,dt \quad \textbf{(3.3)}$$

Com a Equação 3.3, o sinal é representado de forma bidimensional, em função de (w, τ) e, em geral, com a escolha da janela Gaussiana. Mas esta transformada ainda tem duas deficiências: a janela utilizada é fixa, não é possível modificá-la após o início do processamento e, as funções trigonométricas possuem energia infinita.

A técnica de wavelet foi então desenvolvida com alternativa a transformada de Gabor, com funcionamento semelhante. O sinal é multiplicado por uma função, a wavelet mãe (ou função protótipo) e a transformada são calculadas separadamente por segmentos diferentes do sinal no domínio do tempo. As wavelets são funções

matemáticas que separam dados em suas diferentes componentes de frequência, e extraem cada componente com uma resolução adequada à sua escala. Elas têm vantagens em relação à análise de Fourier, pois esta última analisa o sinal como um todo, acarretando uma representação mais pobre para sinais que contêm descontinuidades e variações bruscas (Oliveira, 2007).

A definição matemática da transformada contínua de wavelet (*Continuous Wavelet Transform*) é apresentada pela Equação 3.4. As variáveis *a* e *b* são reais, onde *a* é o parâmetro de escala (contração o dilatação), *b* é um parâmetro de deslocamento.

$$CWT(a,b) = \int_{-\infty}^{+\infty} f(t)\psi_{a,b}^{*}(t)dt \tag{3.4}$$

A função $\psi_{a,b}(t)$, que é a wavelet, matematicamente é definida como apresentada na Equação 3.5.

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}}\psi\left(\frac{t-b}{a}\right), a \neq 0, b \in \Re \tag{3.5}$$

Uma função para ser wavelet deve ser oscilatória, assim seu valor médio no domínio temporal é nulo, então a função precisa satisfazer a propriedade apresentada na Equação 3.6.

$$\int_{-\infty}^{+\infty} \psi(t)dt = 0 \tag{3.6}$$

A inversa da transformada de wavelet é definida na Equação 3.7.

$$f(t) = \frac{1}{C}\iint_{-\infty}^{+\infty} CWT(a,b)\,\psi_{a,b}(t)\frac{db\,da}{a^2} \tag{3.7}$$

Onde, *C* é a condição de admissibilidade, definido na Equação 3.8. O parâmetro *C* deve ser finito e positivo.

$$C = \int_{-\infty}^{+\infty} \frac{|\psi(\omega)|^2}{|\omega|} d\omega \quad \textbf{(3.8)}$$

Existe um grande número de funções que podem ser consideradas wavelets-mãe, que são encontradas na literatura. Podem-se citar algumas: Haar, Daubechies, Symlets, Coiflets, Biortogonal, Meyer, Gaussiana, Chapéu Mexicano, Morlet, Gaussiano Complexo, Shannon, entre outras.

A Transformada Discreta de Wavelet (*Discrete Transform Wavelet*) não são transladas e nem escalonada continuamente, mas em intervalos discretos.

Em (Oliveira, 2007) é apresenta uma pequena modificação na wavelet contínua para propiciar o processo de discretização em pequenos intervalos, como mostrado na Equação 3.9.

$$\psi_{a,b}(t) = \frac{1}{\sqrt{a}} \psi\left(\frac{t-b}{a}\right) \Rightarrow \psi_{m,n}(t) = \frac{1}{\sqrt{|a_0^n|}} \psi\left(\frac{t - nb_0 a_0^m}{a_0^m}\right) \quad \textbf{(3.9)}$$

Onde *m* e *n* são inteiros, a_0>1 é um parâmetro de dilatação fixo b_0 é o fator de translação fixo e *b* depende agra do fator de dilatação.

A discretização ocorre assim no domínio dos parâmetros (escala e translação), não na variável independente do sinal a ser analisado (tempo ou espaço), mostrado na Equação 3.10, com a série de wavelet de tempo contínuo (*Continue Time Wavelet Series*).

$$WT(a,b) = CTWS(m,n) = \frac{1}{\sqrt{|a_0^n|}} \int_{-\infty}^{+\infty} f(t)\, \psi\left(\frac{t - nb_0 a_0^m}{a_0^m}\right) dt \quad \textbf{(3.10)}$$

A variável independente do sinal também pode ser discretizada, como apresentado na Equação 3.11, com as séries wavelets de tempo discreto (*Discrete Time Wavelet Series*).

$$WT(a,b) = DTWS(m,n) = \frac{1}{\sqrt{a_0^n}} \sum_{k=-\infty}^{+\infty} f(k)\,\psi\left(\frac{k - nb_0 a_0^m}{a_0^m}\right) dt \qquad \textbf{(3.11)}$$

Seja um vetor de dados $X=(X_0, X_1, ...,X_T)^T$, para $T=2^M$, com M>0 inteiro, de acordo com (Oliveira, 2007), a transformada discreta de wavelet de X, com uma wavelet mãe $\boldsymbol{\psi}$, é definida na Equação 3.12.

$$d_{j,k}^{(\psi)} = \sum_{l=0}^{T-1} X_l \psi_{j,k}\left(\frac{l}{T}\right), j = 0,1,\dots,M-1 \; e \; k = 0,1,2,\dots,2^j - 1 \qquad \textbf{(3.12)}$$

As transformadas de wavelets possuem uma propriedade que é capaz de representar o comportamento dos dados ou da função de entrada, através dos valores do cálculo de seus coeficientes. Quando ocorrem pequenas variações na função de entrada são gerados coeficientes de baixa amplitude. Quando aplicada no monitoramento de uma rede, sempre que a wavelet apresentar altos coeficientes, pode haver suspeita que a rede encontra-se sob ataque ou tenha ocorrido um evento inesperado (queda do enlace, movimentação de um nó) que causou um distúrbio no sinal. Exatamente por conta dessa sensibilidade da wavelet em detectar variações no fluxo analisado, é que se torna muito importante a sua configuração, a fim de se minimizar o número de falsos positivos.

3.3 Redes Neurais Artificiais

As Redes Neurais Artificiais foram desenvolvidas com objetivo de representar computacionalmente o cérebro humano, onde os computadores têm a capacidade de aprendizado, de realizar generalizações e descobertas. Uma rede neural é composta de um conjunto de neurônios interconectados; a forma de interconexão propicia várias arquiteturas da rede.

A pesquisa sobre redes neurais artificiais teve início nos anos 40, através dos estudos de (Mcculloch e Pitts, 1943) com analogia entre células vivas e o processo eletrônico, através da simulação do comportamento do neurônio biológico.

Os neurônios biológicos, que formam a base do sistema nervoso humano, possuem estrutura comum as outras células, porém é dotado de filamentos

(dendritos e axônios) responsáveis pela conexão com neurônios próximos, através da transmissão e recepção dos impulsos nervosos. Os axônios de um neurônio são conectados aos dendritos dos outros neurônios, através de sinapses.

Nas sinapses não acontece o contato direto entre os neurônios, mas através de uma substância neurotransmissora, que em função de sua quantidade, permite o impulso nervoso atravesse a separação.

Os princípios e algoritmos empregados em redes neurais são utilizados em diversas áreas, incluindo reconhecimento de padrões e processamento de sinais.

Em redes neurais, o neurônio também é chamado de unidade ou nó. O modelo possui um conjunto de pesos, uma unidade de soma ou viés (correspondente à sinapse do neurônio biológico) e uma função de ativação ou de transferência (Cheriet, Kharma *et al.*, 2007). O modelo de um neurônio artificial é mostrado na Figura 3.1.

O sinal de entrada do neurônio é correspondente a um vetor $X=[x_1, x_2, ...,x_d]$ de dimensão *d*. Existe ainda, para cada entrada do vetor *X*, um peso associado w_i, que simula a concentração dos neurotransmissores.

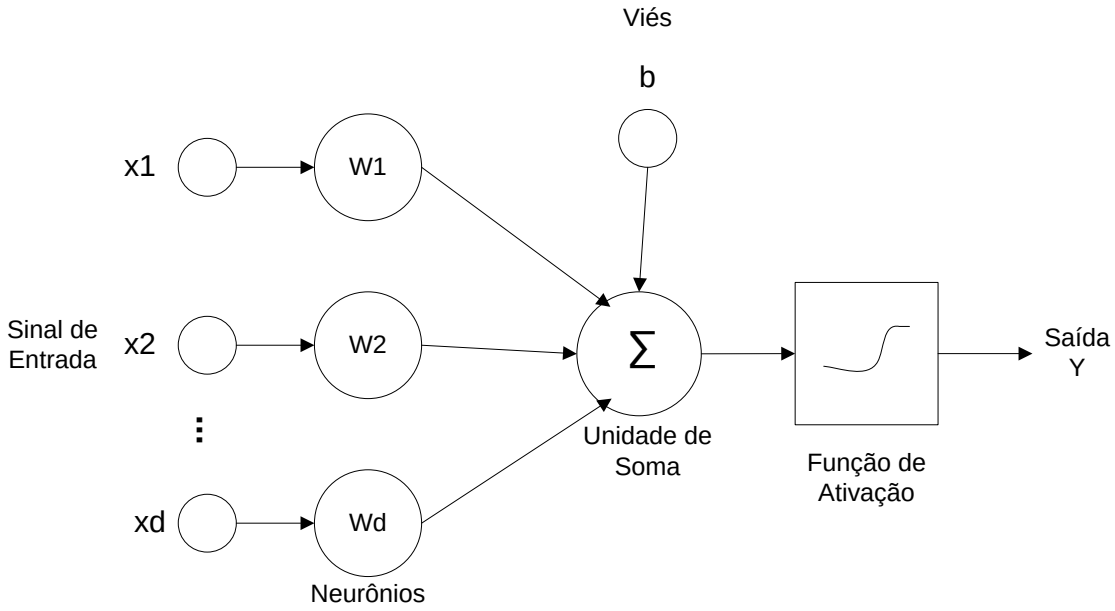


Figura 3.1 - Modelo de um Neurônio

A combinação linear dos valores de entrada, com os respectivos pesos, além de um elemento externo *b* (chamado de viés), produz um valor que ainda deve ser submetido a uma função de ativação para produzir o resultado final. Em termos matemáticos, o valor que a função de transferência recebe é apresentado pela Equação 3.13.

$$y = \sum_{i=1}^{d} w_i x_i + b \qquad \textbf{(3.13)}$$

Existem diversas funções de transferências. São exemplos: função linear, função degrau, função tangente hiperbólica e função sigmóide.

A arquitetura das redes neurais relaciona-se com a organização dos neurônios. Existem três categorias básicas de arquitetura: *feed-forward, feed-back* e auto-organizáveis (Karayiannis e Venetsanopoulos, 2004).

As redes do tipo *feed-forward* são formadas por camadas de neurônios, inclusive poderá haver camadas ocultas. O processamento na rede ocorre somente num sentido, onde os neurônios de uma camada somente se conectam com neurônios das camadas adjacentes. São exemplos deste tipo de arquitetura as redes Perceptron (formada por uma única camada) e de Múltiplas Camadas. A arquitetura *feed-back* contempla as redes cujos neurônios podem se conectar a si mesmo (em uma de suas entradas) ou a neurônios da mesma camada, diferente das redes *feed-forward*. São exemplos desta arquitetura (Widrow e Lehr, 1990), (Hopfield, 1982) e (Elman, 1993).

As redes auto-organizáveis, propostas por (Kohonen e Somervuo, 1998), realizam mapeamento de um espaço de dimensão elevada em estruturas com dimensão topológica inferior com a preservação das relações de vizinhança dos dados de entrada. Uma característica importante das redes neurais é a capacidade de aprendizado. Neste processo, a rede adquire a habilidade de responder a estímulos, através do ajuste de parâmetros internos.

O aprendizado das redes neurais pode ser classificado de duas formas:

- aprendizado supervisionado: são apresentados para a rede, conjuntos formados por valores de entrada e a respectivas saídas. Para cada item na entrada, a rede gera uma saída e compara com o valor

informado. O ajuste de pesos sinápticos é feito com objetivo de obter-se a saída desejada;

- aprendizado não supervisionado: não são apresentadas as saídas desejadas. Em vez disso, para cada conjunto de entrada fornecida, a rede realiza o cálculo da saída. Após isso, a saída é agrupada em classe. Se não houver classe, uma nova será criada. Geralmente é utilizado em redes auto-organizáveis.

Existem várias técnicas para o aprendizado da rede: aprendizagem por correção de erro, aprendizado baseado em memória ou aprendizado competitivo. A representação do conhecimento da rede relaciona-se com os pesos definidos para as conexões e sua formação da base de conhecimento.

As redes neurais possuem alta capacidade de reconhecimento de padrões, que tem por objetivo a classificação em categorias. Essa característica, aliada à competência de generalização, permite o uso das redes neurais em IDS. Porém, somente reconhecerá ataques para os quais foram previamente treinadas. Além disso, o uso de redes neurais tem um processamento muito intenso na fase de treinamento que exige muito do poder computacional do nó. Em contrapartida, na fase operacional ocorre, relativamente, uma baixa demanda dos recursos do nó.

3.4 Comentários Finais

A análise por wavelet permite divisões sucessivas em aproximação e detalhe. Esse método possui a capacidade de ajuste adaptativo e pode detectar anomalias de baixa, média e alta intensidade. Mesmo pequenas anomalias ao longo do tempo poderão ser identificadas através do uso de wavelets.

Uma característica importante das redes neurais é a capacidade de aprendizado. Neste processo, a rede adquire a habilidade de responder a estímulos, através do ajuste de parâmetros internos. Existem várias técnicas para o aprendizado da rede: aprendizado por correção de erro, aprendizado baseado em memória e aprendizado competitivo. A representação do conhecimento da rede relaciona-se com os pesos definidos para as conexões e sua formação da base de conhecimento.

As características apresentadas das wavelets e redes neurais tornam essas técnicas boas candidatas para o emprego em sistemas de detecção de intrusão, conforme apresentado no próximo capítulo.

4 SISTEMAS DE DETECÇÃO DE INTRUSÃO

4.1 Introdução

Com a popularização dos acessos à Internet, muitas instituições utilizam sua infraestrutura para prover a comunicação entre suas unidades. As necessidades de negócios têm motivado as empresas e órgãos governamentais a desenvolver sofisticadas e complexas redes de informações. Essas redes contêm uma variedade de tecnologias, incluindo armazenamento de dados, técnicas de criptografia e autenticação, voz e vídeo sobre IP, acesso remoto e sem fio, entre outros serviços. Contudo, as redes corporativas estão tornando-se mais acessíveis, muitas organizações permitem que seus usuários utilizem os serviços da rede através de conexões oriundas da Internet (Patcha e Park, 2007).

Os benefícios disponibilizados pela Internet, principalmente a web, possibilitou que muitas pessoas, inclusive com poucas habilidades técnicas, tivessem acesso a informações de várias áreas do conhecimento. É provável que a interface gráfica dos aplicativos tenha contribuído nesse processo. Nas informações disponíveis ao público geral, também é possível encontrar receitas para realizar ataques a sistemas de computação, desde sistemas operacionais, aplicativos até as redes de computadores. Em muitos casos, são disponibilizados programas prontos para explorar vulnerabilidades ainda não corrigidas pelos desenvolvedores.

Com motivações financeiras, os crimes que ocorrem hoje na web atualmente são bastante diferentes dos ataques tradicionais (Provos, Rajab *et al.*, 2009). Houve uma evolução significativa na inteligência de ataques e com a dependência crescente da utilização da Internet, torna-se necessário o emprego de técnicas para garantir disponibilidade de serviços na rede. Além disso, são necessários recursos que garantam a integridade dos dados transmitidos e também a identificação correta dos usuários remotos.

Nesse capítulo são apresentados alguns conceitos relacionados à segurança da informação, ataques às redes e o funcionamento dos sistemas de detecção de intrusão.

4.2 Sistema de Detecção de Intrusos

No Brasil, o CERT.br é o grupo de resposta à incidentes de segurança para a Internet, mantido pelo NIC.br, do Comitê Gestor da Internet no Brasil. O CERT.br é responsável por receber, analisar e responder a incidentes de segurança envolvendo redes conectadas à Internet no Brasil. Como apresentado na Figura 3.1, o número de incidentes reportados tem crescido muito nos últimos anos. É possível que esses valores sejam bem maiores, pois parte dos incidentes não são reportados.

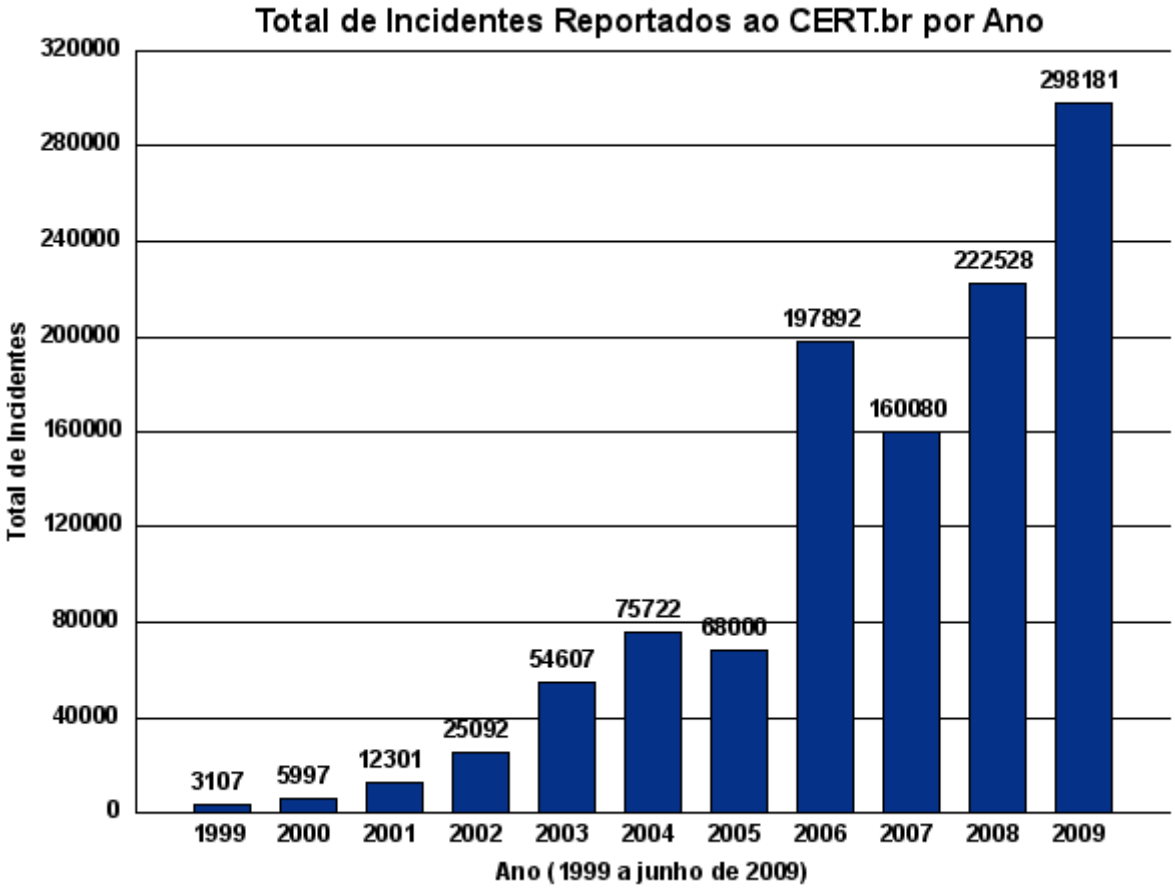


Figura 4.1 - Incidentes de Segurança Reportados no Brasil, fonte: (Centro De Estudos Resposta E Tratamento De Incidentes De Segurança No Brasil, 2009)

A segurança da informação tornou-se fundamental para garantir o correto funcionamento das redes e sistemas computacionais. A segurança da informação está relacionada com a tríade: integridade, confidencialidade e disponibilidade. A integridade está pautada com a exatidão da informação, a certeza de que a mesma não sofreu alterações. A confidencialidade é o processo que busca garantir acesso

somente de usuários autorizados. A disponibilidade relaciona-se com a certeza de que os usuários permitidos terão acesso à informação quando necessitarem. As ações com objetivo de comprometer a integridade, confidencialidade e a disponibilidade de um sistema, ou rede, podem ser classificadas como umaintrusão. Todos os procedimentos, com objetivos de identificar e isolar intrusos é chamado detecção de intrusão.

Um sistema de detecção de intrusão é um software utilizado para detectar acesso sem autorização a um computador ou rede. Um IDS deve ser capaz de descobrir a existência de todo o tráfego malicioso na rede de computadores. Isso inclui ataques à rede, exploração de vulnerabilidades de serviços, ataques à computadores, tentativa de aumento de privilégios, *login* sem autorização além das tentativas de acesso a arquivos. Um IDS é uma entidade de monitoramento dinâmico que complementa a habilidade de monitoramento estático de firewalls (Patcha e Park, 2007).

A anatomia de um ataque tradicional envolve várias fases. A primeira está relacionada com a aquisição de informações sobre a vítima. A segunda fase é composta pela pesquisa de vulnerabilidades, o que pode ser executado com o auxílio de softwares que identificam aplicações e suas fragilidades. A terceira fase envolve a exploração de vulnerabilidades existentes, e finalmente, a manutenção do acesso obtido na fase anterior (Labib, 2004). A partir da segunda fase ocorrem modificações dos pacotes que trafegam pela rede, o que deverá ser detectada pelo IDS, já que mudanças significativas no perfil da rede são esperadas.

O funcionamento geral do IDS é dividido conforme as seguintes tarefas:

- obtenção dos dados de auditoria: quando aplicados às redes existentes, é comum a utilização de aplicativos como *tcpdump* para leitura do tráfego na rede;
- seleção de características: os dados de auditoria formam um conjunto muito amplo e sua utilização direta acarreta um elevado custo computacional. Esse conjunto deve ser otimizado para gerar um subconjunto menor. O subconjunto deve possuir somente as características que mais contribuem para o funcionamento do IDS e manter um nível aceitável da taxa de detecção;

- análise: essa tarefa é responsável pela avaliação que identificará que a rede está sob algum ataque ou permanece numa situação normal de funcionamento.

O IDS é formado por um conjunto de técnicas e métodos que são utilizados na pesquisa de atividades maliciosas no nível da rede bem como no nível de host (Vokorokos, Kleinová *et al.*, 2006):

- monitoramento e análise de atividades dos usuários e sistema;
- controle de configuração de sistema e suas vulnerabilidades;
- análise estatística de padrões baseados na comparação com ataques conhecidos;
- análise de atividades anormais.

Existem diversas propostas para construção de IDS: aprendizagem de máquina (Tsai, Hsu *et al.*, 2009), consumo de energia dos nós em rede ad hoc sem fio (Ma e Fang, 2009), mineração de dados (Singh, Masseglia *et al.*, 2009). Porém, o IDS precisa ser efetivo e eficiente. O IDS efetivo consegue distinguir a classificação correta de ações maléficas ou normais. O IDS eficiente é executado continuamente, mas com pouco consumo de memória e processamento dos nós, e não deve interferir de forma significativa no desempenho da rede. De acordo com (Xiao, Hong *et al.*, 2006), a detecção de intrusos assume que usuários e atividades de programas são observáveis, isso significa que quaisquer ações que o usuário ou aplicação inicia geram atividades que podem ser gravadas em registros (*logs*) e o IDS tem acesso a estes dados. Os registros gerados são chamados de “dados de auditoria”. O IDS realiza análise nos dados de auditoria com objetivo de identificar comportamentos anormais de elementos da rede. Se ocorrer comportamento anômalo, o IDS poderá inferir que o sistema está sob ataque.

A detecção de intrusão pode ser agrupada em duas categorias principais: detecção por anomalia e detecção por abuso. No sistema de detecção por anomalia é criado um perfil de comportamento normal dos nós. Alguma atividade diferente das que foram definidas pelo perfil podem ser consideradas como possível ataque. Mudanças topológicas ou comportamentais (perfil de utilização de banda da rede, execução de aplicações, entre outros) também podem ser reconhecidas como ataques. A detecção baseada em especificação confina uma aplicação ou protocolo num conjunto de operações que são consideradas corretas. A execução da

aplicação ou protocolo é monitorada. Esta modelo permite detectar ataques desconhecidos (Axelsson, 2000).

A comparação das categorias permite concluir que a desvantagem da primeira refere-se ao alto número de alarmes falso-positivos, enquanto que da segunda categoria está relacionada com a necessidade de conhecimento prévio dos ataques. Com relação às vantagens, a primeira categoria pode detectar mesmo os ataques não conhecidos ou novos, enquanto que a segunda pode ser executada com menor esforço computacional. Existem propostos de IDS híbridos que fazem uso das duas técnicas, com objetivo de aproveitar as vantagens de ambas e minimizar suas desvantagens.

A análise dos dados de auditoria pode ser realizada através de dois tipos de mecanismos de detecção. A análise *postmortem* pode explorar muitas horas de tráfego de dados como um único conjunto de dados, com processamento mais rigoroso exigindo e maior emprego de recursos computacionais. Tal análise pode ser útil para fins de engenharia de tráfego, análise de uso de recursos, criação de perfil de utilização, etc. Por outro lado, a análise em tempo real concentra em analisar uma pequena janela de tráfego dados, com vista a fornecer um alerta rápido e iminente de anomalias do trafego. Análise em tempo real utiliza técnicas menos sofisticadas devido à demanda de recursos para detecção de ataques (Kim e Reddy, 2008).

Um IDS pode realizar a análise dos dados de auditoria em computadores individuais. A arquitetura do IDS é então baseada em agentes instalados nesses computadores que se comunicam com um sistema central, nesse caso, o IDS é denominado *Host-based IDS*. Outra forma ocorre quando os dados de auditoria são capturados através do tráfego da rede e então são chamados *Network-based IDS*.

Uma arquitetura genérica de IDS contém os módulos apresentados na Figura 4.2 e descritos a seguir (Patcha e Park, 2007):

- coleta de dados de auditoria: utilizado na fase de coleta. Os dados coletados nessa fase são analisados pelo algoritmo de detecção de intrusão para descobrir traços de atividades suspeitas. Os dados coletados podem ter origem em logs de computadores ou rede, registro de comandos ou logs de aplicações;
- armazenamento: os dados de auditoria são armazenados em algum local, temporariamente ou definitivamente, para então serem

processados. Em alguns casos, o volume de armazenamento pode ser muito grande. Este é um elemento crucial em qualquer IDS, e isso levou alguns pesquisadores da área iniciar pesquisas com objetivo de reduzir os dados de auditoria;

- processamento (detecção): o bloco de processamento é o coração do IDS. Nesse módulo são executados algoritmos para encontrar provas (com certo grau de certeza) de comportamentos suspeitos nos dados de auditoria;
- dados de configuração: este bloco afeta as operações do IDS. Ele contém os parâmetros sobre a localização dos dados de auditoria, como serão realizadas as repostas aos incidentes. Este é, portanto, o principal meio de controle do responsável pela segurança sobre o IDS;
- dados de referência: armazena informações sobre assinatura e/ou perfil normal de comportamentos conhecidos. No último caso, as atualizações do perfil considerado normal são permitidas e ocorrem em intervalos regulares;
- processamento de dados: o elemento de processamento frequentemente deve armazenar resultados intermediários, por exemplo, informações sobre assinaturas. O espaço necessário para armazenar estes dados pode crescer muito.
- alarme: parte do sistema responsável que informa sobre eventos suspeitos detectados pelo IDS;

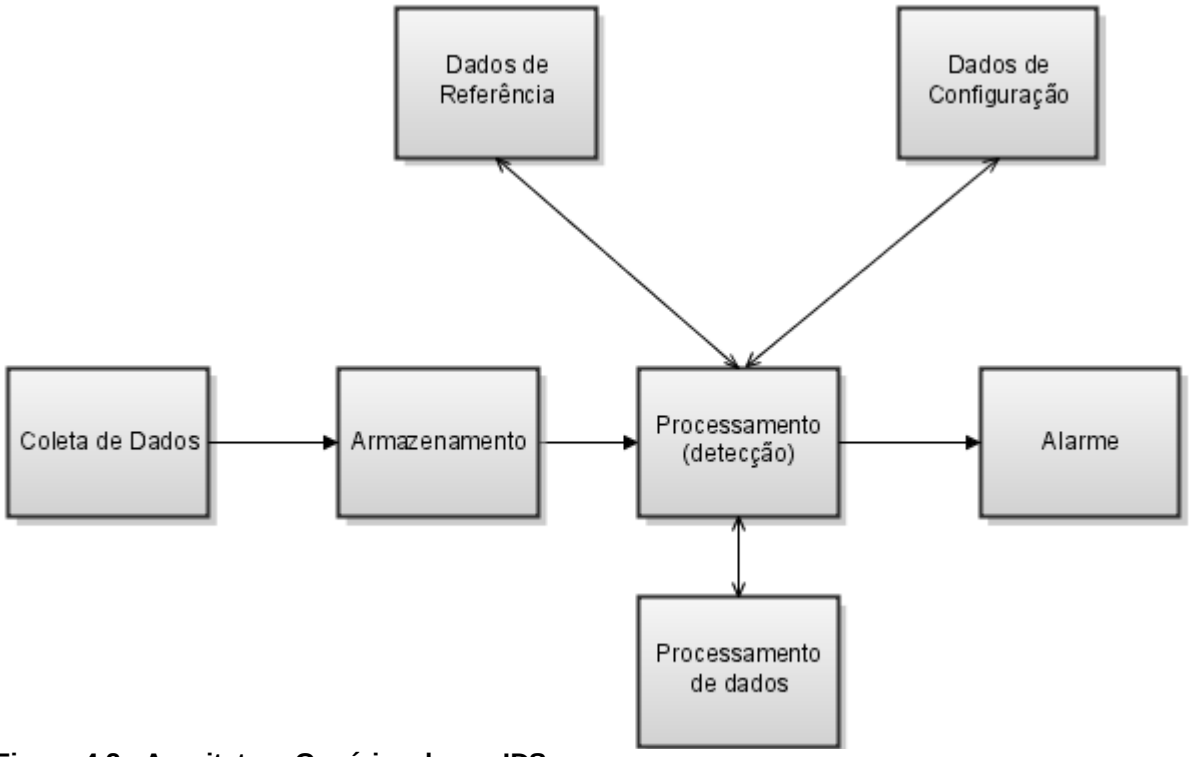


Figura 4.2 - Arquitetura Genérica de um IDS

A avaliação dos IDS ocorre de acordo com métricas escolhidas que representam o grau de precisão de detecção, como apresentado na Tabela 4.1. As métricas bastante utilizadas são (Lazarevic, Ertoz *et al.*, 2003):

- falso negativo: eventos intrusivos, ou ataques, que são classificados pelo IDS como atividades normais;
- falso positivo: ocorrência de eventos normais que são classificados pelo IDS como atividades intrusivas;
- verdadeiro negativo: atividade normal classificada corretamente pelo IDS;
- verdadeiro positivo: atividade intrusiva classificada corretamente pelo IDS.

Algumas avaliações de IDS podem ser realizadas através do uso de bases de treinamento. Nesses casos, a base possui todos os dados referentes às conexões de rede ou utilização de um determinado sistema. Além disso, também é incluída a identificação do evento (normal ou ataque). Assim, as métricas são calculadas através da comparação das classificações realizadas pelo IDS com a base original empregada.

Tabela 4.1 - Métricas de Avaliação de IDS

Tipo do Evento	*Classificação pelo IDS como Normal*	*Classificação pelo IDS como Ataque*
Normal	Verdadeiro negativo	Falso positivo
Ataque	Falso negativo	Verdadeiro positivo

Apenas a análise individual das métricas não é suficiente para avaliação de IDS, a análise dos resultados independentes poderá ocasionar erro. A precisão global relaciona as identificações corretas realizadas pelo IDS comparadas com as taxas de classificações erradas. O cálculo da precisão global é realizado conforme a Equação 4.1.

$$Precisão\ Global\ (OA) = \frac{TP + TN}{TP + TN + FP + FN} \tag{4.1}$$

Onde:

TP é a taxa de verdadeiro positivo

TN é a taxa de verdadeiro negativo

FP é a taxa de falso positivo

FN é a taxa de falso negativo

A taxa da precisão global informa a acuraria das detecções realizadas pelo IDS e exprimem uma relação entre as classificações corretas e incorretas.

4.3 Algumas Abordagens Propostas para Construção de Sistemas de Detecção de Intrusão

Encontra-se na literatura muitos trabalhos que apresentam propostas de IDS. As abordagens utilizadas são variadas e sugerem aplicações de diferentes técnicas nas implementações de sistemas de detecção de intrusão. Além disso, algumas propostas são específicas para a arquitetura de rede (rede de sensores, redes ad hoc sem fio, redes com fio, etc), pois consideram características intrínsecas do modelo de comunicação da rede. São apresentados a seguir alguns trabalhos com propostas de IDS.

A proposta de (Huang, Thareja *et al.*, 2006), (Kim, Reddy *et al.*, 2004) e (Magnaghi, Hamada *et al.*, 2004) baseiam-se no uso de wavelet para um IDS em roteador de borda. Nessa abordagem, é possível identificar apenas anomalias entre os hosts interno e externo, pois o tráfego entre os nós internos não passam pelo roteador, e por isso não podem ser identificados.

É apresentado por (Hamdi e Boudriga, 2007) uma proposta de IDS baseado na detecção de comportamento anômalo em redes com fio. Um módulo classificador cria um perfil da rede que é atualizado constantemente. A detecção é realizada comparando o estado atual com o perfil gerado previamente. Mudanças sutis na rede podem não ser identificadas por esta técnica. Este tipo de IDS consome mais recursos computacionais do que os baseados em assinaturas.

Abordagens baseadas em redes neurais artificiais também são encontradas em grande número na literatura. De certa forma, as abordagens são parecidas, a exemplo de (Ahmad, Ansari *et al.*, 2008), (Song, Zhang *et al.*, 2008), (Yu, Chen *et al.*, 2007) e (Mafra, Da Silva Fraga *et al.*, 2008). Neste último, os autores usam duas camadas (Mapas de Kohonen e Support Vector Machine), a primeira faz a classificação enquanto que a segunda executa a detecção propriamente dita. Essas abordagens foram empregadas em redes com fio, com a obtenção dos dados de auditoria diretamente de roteadores de borda.

A proposta de (Karygiannis, Antonakakis *et al.*, 2006) faz a detecção de nós maliciosos em MANET através de um conjunto de métricas e representação por grafos. O sistema percorre todos os nós da rede e realiza testes de encaminhamento de pacotes para a rede local ligada ao nó. Se não houver hosts conectados à rede local dos nós, o sistema poderá concluir que está acontecendo algum tipo de ataque, certamente isso é uma desvantagem desta proposta.

A sugestão apresentada em (Xiao, Hong *et al.*, 2006) é fundamentada no uso de firewalls em nós da rede. Tais nós são responsáveis pelo controle de ingresso na rede, mas esse procedimento é baseado em endereço MAC. Atualmente é muito simples a clonagem de endereços MAC, desta forma, a proposta é ineficiente.

A abordagem apresentada em (Nong, Xiangyang *et al.*, 2001) faz uso de propriedades de probabilidades para detecção de intrusão em sistemas de informação. Os autores consideram sistemas de informação um conjunto de hosts e os enlaces de comunicação entre eles. As técnicas utilizadas incluem árvores de decisão, teste de *Hotelling*, *Chi-quadrado* e cadeias de *Markov*. Esta proposta é

baseada na análise estatística da frequência, duração e a ordem de eventos que ocorrem durante um intervalo de tempo observado nos dados de auditoria. A proposta é para IDS baseado em host e os testes foram realizados em uma máquina Sun SPARC.

Em (Cabrera, Ravichandran *et al.*, 2000) é encontrada uma proposta de modelagem estatística de tráfego para detectar ataques em redes de computadores. São avaliados ataques de *DoS* e *Probe*. Os ataques são identificados devido ao aumento do número total de conexões durante um período de observação e a avaliação é realizada através do teste de *Kolmogorov-Smimov* aplicado sobre a taxa de bytes transmitidos na conexão.

Em (Schonlau, Dumouchel *et al.*, 2001) é apresentada uma proposta para detecção de intrusão para hosts baseado em métodos estatísticos. O objetivo é realizar a detecção de ataques mascarados. Esse tipo de ataque é caracterizado pela tentativa de uso de contas alheias para danificar a rede ou host. Os dados de auditoria são obtidos através dos *logs* do sistema que armazenam todos os comandos executados pelos usuários. Nessa proposta, várias técnicas são utilizadas *Uniqueness, Bayes one-step Markov, Hybrid multistep Markov, Compression*. Porém, não houve integração entre os métodos, cada um foi utilizado separadamente.

4.4 Comentários Finais

Foram apresentados nesse capítulo conceitos relacionados à segurança da informação e sua importância na atualidade. Com o crescimento das interligações das redes e uso da Internet, houve aumento considerável de usuários dos serviços providos pelas redes. A segurança da informação tornou-se imprescindível para disponibilização de serviços aos usuários.

Os sistemas de detecção de intrusão tornaram-se ferramentas importantes para melhorar o nível de segurança das redes e das aplicações. Diversas técnicas são empregadas na tentativa de melhorar a taxa de detecção global, a exemplo de inteligência artificial, processamento digital de sinais, métodos estatísticos, séries temporais, entre outros.

O número de propostas para implementação de IDS são elevados, esse é um indício que se trata de uma área que ainda necessita de muitos estudos,

principalmente para acompanhar o alto desenvolvimento das tentativas de ataques sem comprometer o funcionamento da rede ou dos sistemas.

5 SISTEMA DE DETECÇÃO DE INTRUSÃO COM ABORDAGEM BASEADA EM PROCESSAMENTO DIGITAL DE SINAIS E REDES NEURAIS PARA REDES DE COMPUTADORES

5.1 Introdução

Com o aumento da interconexão entre redes, a segurança da informação tornou-se um desafio. As redes estão sujeitas a vários tipos de ataques que podem ter origem interna ou externa, alguns com objetivos de paralisar serviços, outros com a intenção de roubar informações e em outros casos, apenas por diversão dos atacantes. Além disso, até pouco tempo, as redes eram restritas a computadores, agora aceitam vários tipos de equipamentos: sensores, telefones inteligentes e celulares, entre outros. Portanto, as propostas de melhoria de segurança devem considerar a evolução tecnológica que está ocorrendo.

A tríade confidencialidade, integridade e disponibilidade de recursos representam fatores vitais para a segurança da informação, onde uma ação maléfica ou não intencional pode comprometer o sistema, caracterizando uma intrusão. O IDS deve conseguir identificar essa ação, mas sem comprometer o funcionamento normal da rede. Um IDS é uma ferramenta de segurança que, como outras medidas, a exemplo de antivírus e firewalls, destinam-se a reforçar a segurança da informação em sistemas de comunicação (Teodoro, Verdejo *et al.*, 2009).

Nesse capítulo é apresentada uma proposta de sistema de detecção de intrusão com abordagem híbrida para redes de computadores. Essa abordagem utiliza duas camadas: a primeira baseada em wavelets, para detecção de comportamentos anômalos e a segunda utiliza redes neurais para classificação dos ataques ou confirmação de uma situação normal.

5.2 Abordagem Proposta com Utilização das Transformadas de Wavelet e Redes Neurais Artificiais

Nesse trabalho é proposto um IDS híbrido de duas camadas: a primeira baseada em transformadas de wavelets, para detecção de comportamentos anômalos, e a segunda em redes neurais artificiais, para classificação dos ataques.

As wavelets podem identificar até mesmo mudanças sutis em resultados de uma função (ou conjunto de dados). Essa capacidade pode ser empregada para detectar variações de comportamento em uma rede de computadores e essa mudança pode ser um indício de um ataque. A proposta aqui apresentada inclui as transformadas de wavelets para descrever o comportamento característico da rede. Essa informação é empregada para avaliar se a rede está funcionando corretamente ou está sob algum tipo de ataque. As redes neurais, depois do treinamento inicial, não necessitam de grande esforço computacional para classificação de padrões. Esses métodos, utilizados em conjunto, permitem então reconhecer e classificar alterações oriundas de ataques na rede.

O uso das transformadas de wavelets baseia-se em uma função de protótipo, chamada wavelet mãe. Com a wavelet mãe é realizada uma operação de convolução com outra função ou conjunto de dados. Essa operação matemática produz os coeficientes wavelets. As oscilações dos valores dos coeficientes representam as variações dos padrões dos dados de entrada que foram processados pela transformada de wavelet. O indício de uma situação anômala é obtido através de um limiar relacionado aos valores dos coeficientes. Nesse caso, é utilizada a segunda camada para classificar o possível ataque.

Um conjunto de métricas pode ser utilizado para compor o perfil de funcionamento da rede, que será utilizado juntamente com a wavelet mãe para produção dos coeficientes. Exemplos de métricas importantes incluem:

- banda disponível;
- banda utilizada;
- número de fluxos;
- número de pacotes enviados e/ou recebidos;
- quantidade de bytes enviados e/ou recebidos;
- taxa de transmissão.

Deve existir um limiar que indicará se a rede está sob condição normal ou anômala. Nesse trabalho, o limiar é dado pelo desvio padrão da distância euclidiana entre os coeficientes da transformada de wavelet. A distância euclidiana entre os coeficientes é definida pela Equação 5.1. Nas avaliações realizadas, o uso da distância euclidiana foi satisfatório.

$$D_e = \sqrt{(C_a - C_b)^2 + (t_a - t_b)^2} \quad \textbf{(5.1)}$$

Onde:

C_a – indica o valor do coeficiente da transformada de wavelet no dado auditado a.

C_b – indica o valor do coeficiente da transformada de wavelet no dado auditado b.

t_a – indica o tempo em que o valor do coeficiente da transformada de wavelet foi calculado no dado auditado a.

t_b – indica o tempo em que o valor do coeficiente da transformada de wavelet foi calculado no dado auditado b.

A distância euclidiana foi escolhida por mostrar quando ocorre uma alteração nos valores dos coeficientes, quanto maior for esse valor entre os coeficientes, maiores serão as chances de ocorrer um comportamento anômalo no tráfego da rede. Porém, como é calculada a distância euclidiana para todos os valores dos coeficientes, é necessário obter uma medida da dispersão estatística que indique o quão longe os valores calculados se encontram do valor médio. Esta medida define o limiar de um comportamento normal.

O algoritmo da primeira camada, de forma simplificada é apresentado na Figura 5.1. Os dados de auditoria, que serão utilizados no cálculo da métrica, são obtidos através da captura do tráfego da rede. Note que o algoritmo fica continuamente executando para auditar os dados e calcular os coeficientes relacionados. A seguir, a distância euclidiana entre os coeficientes é calculada, assim como seu desvio padrão, que é o limiar proposto no algoritmo. Para valores maiores do que o limiar, um comportamento anômalo é inferido, que ativa o algoritmo da segunda camada, baseado em redes neurais, para classificar o possível ataque.

```
Algoritmo IDS_Wavelet_e_Redes_Neurais
Declare S – Conjunto de dados auditados no tráfego da rede
        C – Conjunto de coeficientes da transformada Wavelet
        DE – Conjunto de valores correspondente a distância
euclidiana entre os coeficientes da transformada Wavelet
        Limiar – valor máximo para considerar uma função f(t)
como comportamento normal
Início
     Repita Enquanto Houver Dados de Auditoria
     S = Dados de Auditoria
     C = Calcular Coeficientes Wavelet(S)
     DE = Calcular Distância Euclidiana(C)
      Limiar = Calcular Desvio-Padrão(DE)
     Se (C > Limiar) Então
          Utilizar Redes Neurais para confirmar ataque
     Fim (Se)
Fim
```

Figura 5.1 - Algoritmo da Primeira Camada

A segunda camada é formada por uma rede neural artificial, com cinco neurônios na camada de saída. Essa rede foi treinada para reconhecer quatro classes distintas de ataques, conforme apresentada na Tabela 5.1, além do tráfego normal.

Tabela 5.1 - Classes de Reconhecimento do Tráfego

Classe	***Significado***	***Descrição***
U2R	Aumentar Privilégios	Acesso não autorizado com intuito de aumentar privilégios.
R2L	Remoto para Local	Tentativa de acesso remoto não autorizado, por exemplo, pelo uso de quebra de senhas.
Probe	Sondagem	Tentativa de identificar serviços ativos através de varreduras de portas.
DoS	Negação de	Caracteriza-se pelo envio de grande número de

	Serviço	solicitações a um mesmo host, em curto período de tempo.
Normal	Tráfego normal	Identifica um comportamento normal na rede.

A rede neural é formada por um MLP que foi treinado com o algoritmo *back propagation* (Hecht-Nielsen, 1988). A camada de entrada foi constituída com 41 neurônios, a camada oculta com 20 neurônios. O valor de ativação do neurônio na camada de saída é de 0,9, com o uso da função tangente hiperbólica.

Para avaliar a abordagem proposta foi utilizado o software Matlab (The Matworks, 2008) e aplicado a um conjunto de dados de auditoria. O Matlab é um ambiente voltado para cálculos numéricos com ferramentas capazes de reproduzir importantes classes de processos dinâmicos, permitindo a execução de cálculos matemáticos de forma simples. A implementação da rede neural foi realizada com a função *newff*, que automaticamente realiza a normalização dos dados de entrada. O software ainda possui vários conjuntos de bibliotecas, chamadas *toolbox,* que programam diversas funções utilizadas nessa proposta.

5.3 Simulações e Resultados

5.3.1 Avaliação de Algumas Famílias de Wavelets

As transformadas wavelets possuem várias famílias que diferem entre si através da wavelet mãe ou função protótipo (Graps, 1995). Para auxiliar na escolha da família da wavelet, foi realizado experimento com o uso do simulador NCTUns (Wang, Chou *et al.*, 2003). Esse simulador utiliza a pilha de protocolos TCP/IP implementado pelo sistema operacional e também permite a integração e uso de aplicações reais no ambiente simulado, além da utilização dos comandos tradicionais de sistemas Unix como *ping, ifconfig, netstat e tcpdump*.

Foi simulada uma rede ad hoc sem fio, conforme topologia apresentada na Figura 5.2 e parâmetros de configuração de acordo com a Tabela 5.2. Os dados de auditoria foram obtidos diretamente no simulador, através de uma função de exportação de arquivos textos. A rede simulada foi configurada para representar duas situações: um comportamento normal e a incidência de um ataque.

O funcionamento normal da rede é representado pela comunicação dos clientes (computadores um e dois) e servidor (computador três). Nesse caso, foi

gerado tráfego UDP com taxa de transmissão de 2248Kbps (ou). A situação do ataque foi simulada por dois nós atacantes (computadores quatro e cinco). Foi simulado um ataque *DoS* ao servidor através da geração de tráfego dos ataques para o servidor com taxa de transmissão de 32768Kbps, no período entre 100 à 160s.

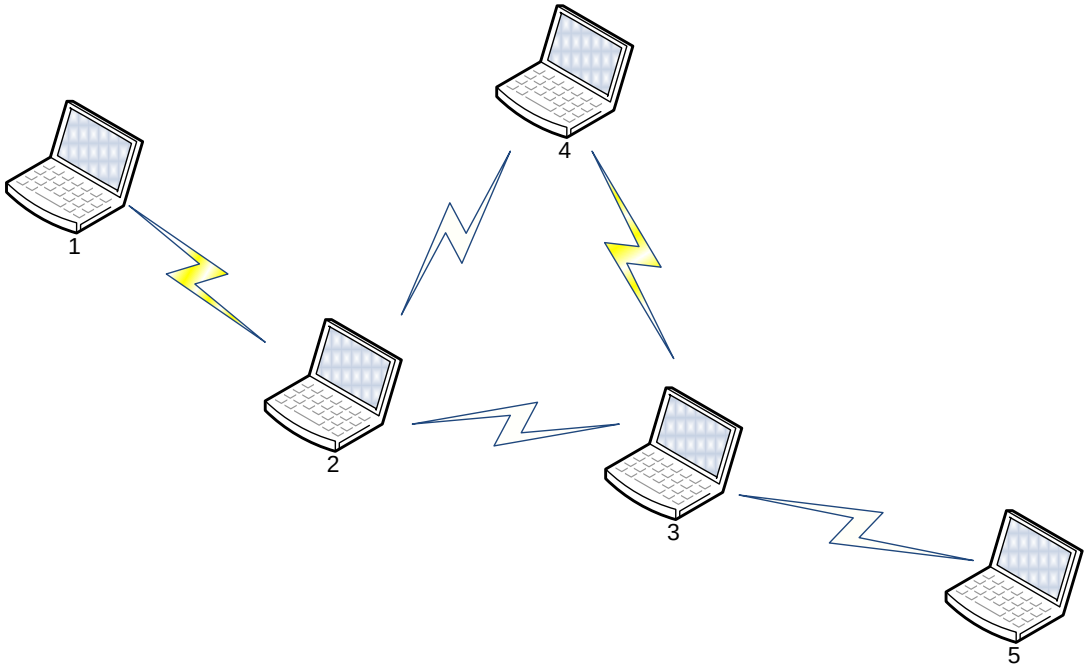


Figura 5.2 - Topologia Simulada

Tabela 5.2 - Parâmetros da Simulação

Parâmetro	***Valor***
Tempo de simulação	400s
Quantidade de nós	5
Distância média entre os nós	200m
Interface de rede	IEEE 802.11b
Alcance da transmissão	250m

É apresentado na Figura 5.3 o gráfico da Taxa de Recebimento dos Nós. Essa métrica foi obtida diretamente no simulador e foram coletadas em intervalos de um segundo. Ainda nessa figura, é percebida a taxa acentuada de pacotes

recebidos pelo nó três durante o período de 100 a 160s, devido ao ataque simulado nesse intervalo.

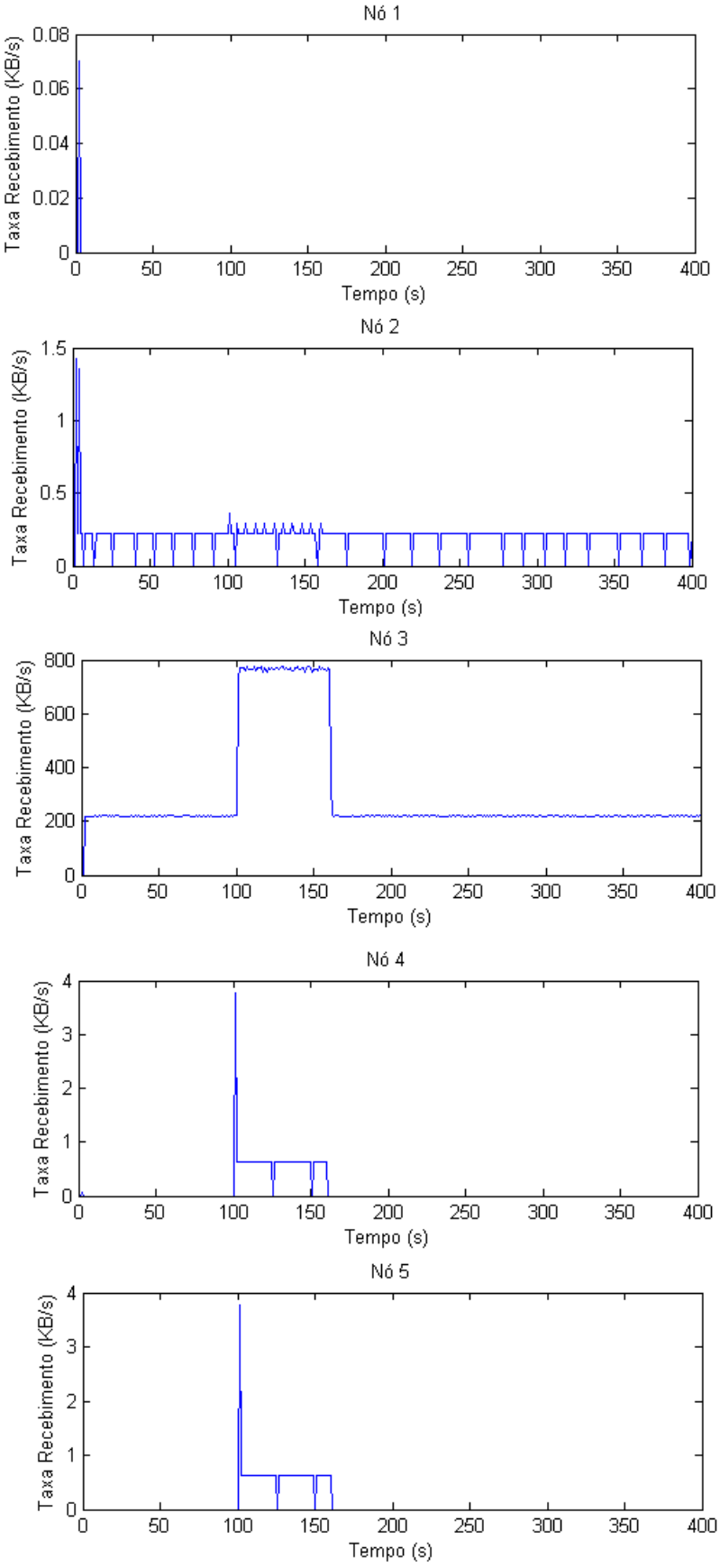


Figura 5.3 - Taxa de Recebimento dos Nós

Foram utilizadas várias famílias de wavelets para avaliar o comportamento da rede. Utilizou-se a taxa de pacotes descartados como métrica que foram obtidos em intervalos de tempo a cada segundo. São apresentados na Figura 5.4 os coeficientes das transformadas de Daubechies, Symlets, Coiflets e DMeyer. Observa-se que todas as famílias empregadas detectaram as perturbações no sinal avaliado, como é observado pelas oscilações de valores dos coeficientes no período compreendido de 100 a 160s. Além disso, com o uso da transformada de Haar os mesmos resultados foram obtidos. Portanto, nesse cenário avaliado, os resultados sugerem que a função empregada não é relevante.

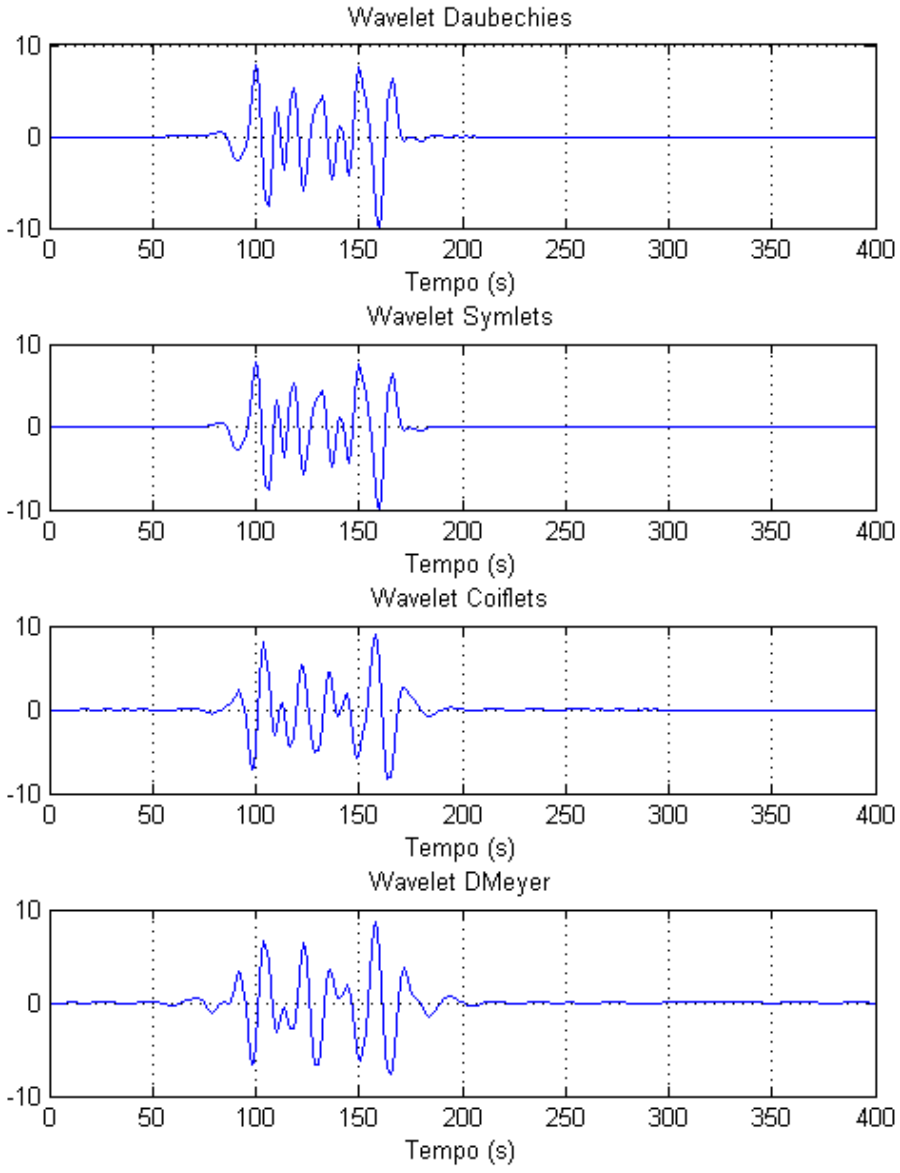


Figura 5.4 - Algumas Famílias de Wavelet

5.3.2 Emprego da Proposta em uma Rede em Laboratório

Para avaliar a abordagem apresentada, foi construída uma rede ad hoc sem fio em laboratório, constituída por três equipamentos, interligada conforme a topologia apresentada na Figura 5.5 e configurados conforme parâmetros apresentados na Tabela 5.3.

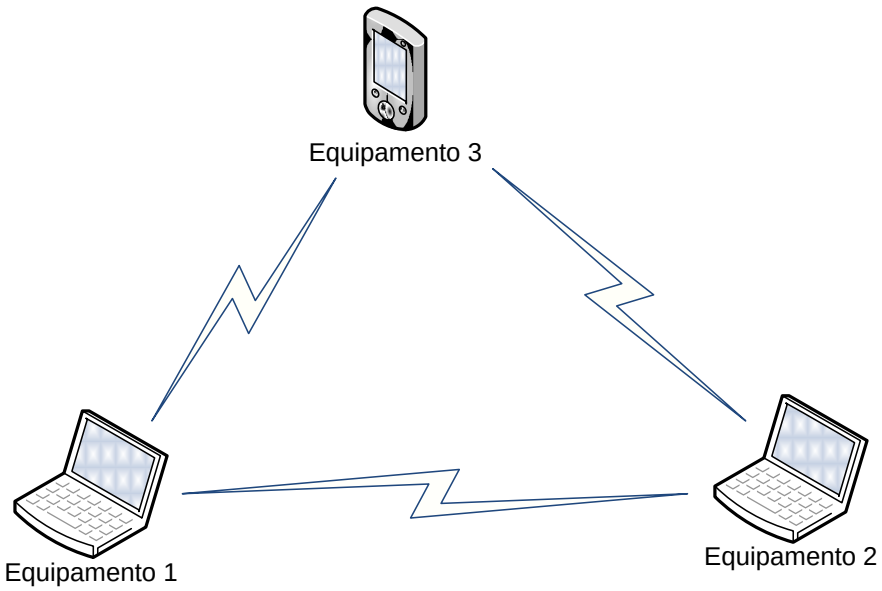


Figura 5.5 - Topologia da Rede no Laboratório

Tabela 5.3 - Rede em Laboratório

Equipamento	***1***	***2***	***2***
Tipo	Notebook	Notebook	Smartphone HTC Touch
Sistema Operacional	Linux Fedora 8	Windows XP SP2	Windows Mobile 6.0 Pro
Interface de Rede	IEEE 802 a/b/g	IEEE 802 a/b/g	IEEE 802.11b/g
Função	Atacante	Vítima	Host na rede
Softwares Utilizados	Nmap e Servidor Web	Navegador de Internet e Sniffer de Rede	Navegador de Internet

Como uso normal da rede, foi habilitado um servidor web no equipamento número um, enquanto os outros nós o utilizaram a partir de um navegador (browser).

Durante esse período, foram realizados ataques TCP/RPC, conforme os períodos indicados na Tabela 5.4. Esse tipo de ofensiva foi escolhido pela sua popularidade nos dias atuais e por ser comum seu emprego para enumeração de serviços disponíveis nos computadores.

Tabela 5.4 - Períodos dos Ataques

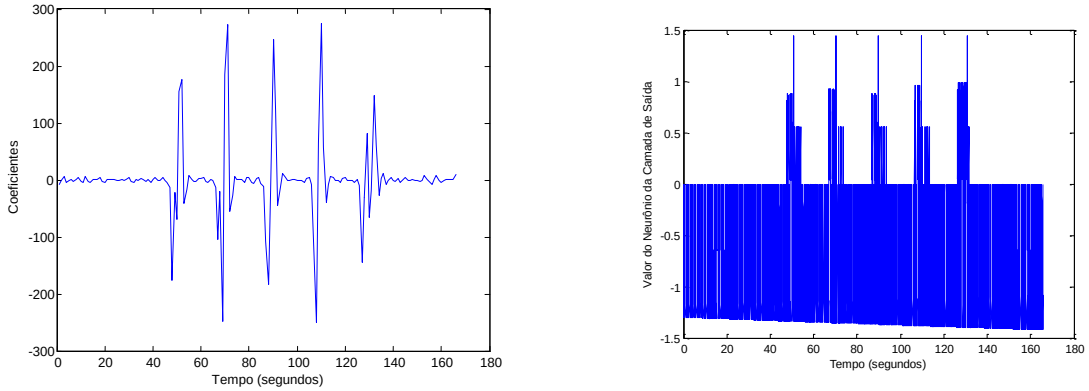
Ataques	***Período (em segundos)***
Primeiro	Entre 47 a 50
Segundo	Entre 67 a 69
Terceiro	Entre 86 a 89
Quarto	Entre 106 a 108
Quinto	Entre 126 a 130

Os dados transmitidos na rede foram capturados com o uso de um *sniffer*. Esses dados foram organizados na seguinte ordem:

- número (sequencia do pacote);
- tempo (medido em segundos com seis casas decimais, relacionado ao início da simulação);
- protocolo;
- endereço IP de origem;
- endereço IP de destino;
- porta de destino;
- tamanho dos pacotes;
- flags do protocolo TCP (no caso dos protocolos ARP, ICMP e UDP este campo é nulo).

Nessa avaliação, utilizou-se o número de pacotes recebidos na interface como métrica para aplicação da wavelet de Daubechies, além disso, houve um processamento para agrupar os dados de auditoria em uma unidade de tempo (um segundo). Após o processamento, foram obtidos os coeficientes da wavelet como mostrados na Figura 5.6(a), onde é possível verificar cinco momentos de perturbações no sinal que correspondem aos cinco intervalos dos ataques. Dessa forma, a wavelet foi capaz de detectar, satisfatoriamente, as anomalias. Para facilitar

a comparação entre os resultados das duas camadas, os valores do neurônio da camada de saída foram agrupados e apresentados no mesmo período de tempo de observação da rede, como mostrado na Figura 5.6(b). Observa-se que a rede neural classificou de forma satisfatória os cinco ataques realizados.



(a) - Camada Wavelet (Coeficientes x Tempo)

(b) - Camada Rede Neural (Valores do neurônio da camada de saída x Tempo)

Figura 5.6 - Resultados experimentados pelo IDS híbrido proposto no cenário de ataque TCP/RPC.

Durante o funcionamento dos testes, todos os pacotes que trafegaram pela rede foram capturados, que totalizaram 17.317. Desse conjunto, 1000 pacotes foram selecionados para treinamento, e 350 para testes de validação da segunda camada do IDS. Durante esta fase, foram utilizados 2500 ciclos de treinamento (época). A curva de treinamento é apresentada na Figura 5.7. Todos os pacotes foram então apresentados à segunda camada para classificar o ataque.

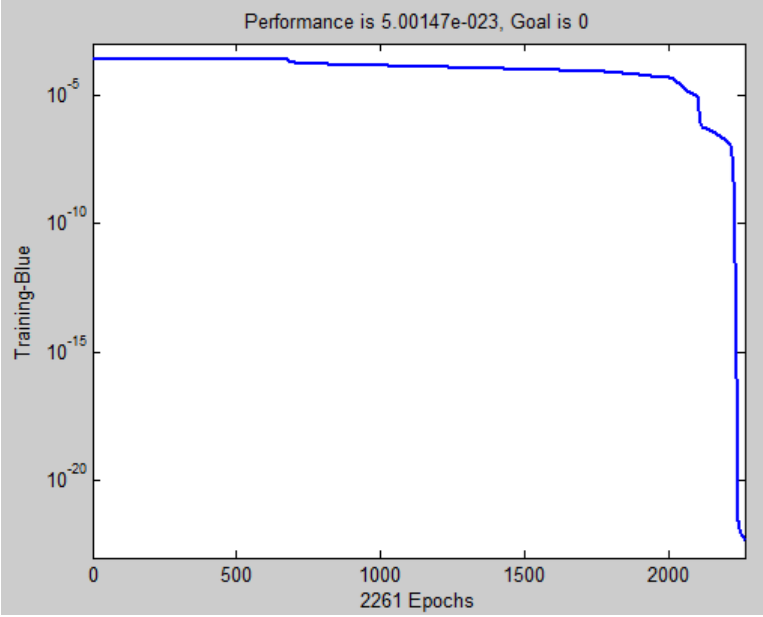


Figura 5.7 - Curva de Treinamento da Rede Neural

5.4 Estudo de Caso – Utilização do KDD 99 com Dados de Auditoria

5.4.1 Descrição da Base de Dados do KDD 99

Para avaliar a proposta aqui apresentada, foi utilizado um conjunto de dados de auditoria disponibilizado pelo laboratório Lincoln do MIT chamada de KDD 99. Essa base foi construída em 1999 através da captura do tráfego, no formato do *tcpdump*, da rede do laboratório e dados inseridos artificialmente que simulam certos tipos de ataques a três alvos baseados em sistemas operacionais distintos. A topologia da rede que originou essa base de dados é apresentada na Figura 5.8.

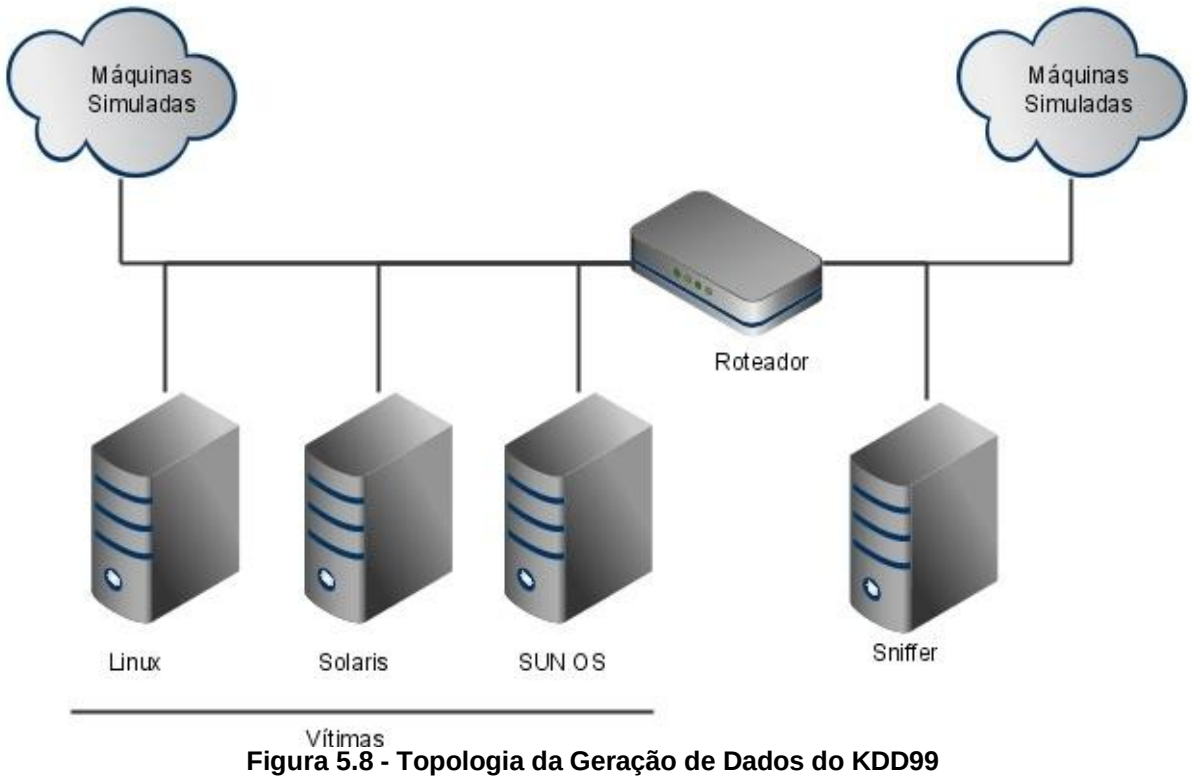


Figura 5.8 - Topologia da Geração de Dados do KDD99

Os dados capturados foram processados para serem utilizados como parâmetro para análise de desempenho em uma competição de Sistemas de Detecção de Intrusão em 1999. Para isso, foram sumarizados em conexões e cada conexão é representada por uma linha. Cada linha contém 41 características (duração, protocolo, aplicação, flags, entre outros). As conexões possuem uma etiqueta que informa sua classificação como normal ou o nome do ataque. Na Figura 5.9 são apresentados alguns exemplos de conexões do KDD 99.

```
5,tcp,smtp,SF,959,337,0,0,0,0,0,0,0,1,0,0,0,0,0,0,0,0,0,0,1,1,0.00,0.00,0.00,0.00,
1.00,0.00,0.00,144,192,0.70,0.02,0.01,0.01,0.00,0.00,normal.

0,tcp,http,SF,54540,8314,0,0,0,2,0,1,1,0,0,0,0,0,0,0,0,0,2,2,0.00,0.00,0.00,0.00,1.
00,0.00,0.00,118,118,1.00,0.00,0.01,0.00,0.00,0.00,0.02,0.02,back.

0,tcp,http_443,S0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,0,114,2,1.00,1.00,0.00,0.00,0.0
2,0.06,0.00,255,2,0.01,0,07,0.00,0.00,1.00,1.00,1.00,0.00,neptune.
```

Figura 5.9 – Exemplos de Conexões na Base do KDD

Uma conexão é uma sequencia de pacotes TCP com início e término num determinado intervalo de tempo, com fluxo de dados entre um nó origem e destino, sobre um protocolo definido. Esses dados foram processados e sumarizados em quatro conjuntos (Kayacik, Zincir-Heywood *et al.*, 2005), (Fries, 2008):

- características básicas: constituídas por informações dos cabeçalhos dos pacotes sem a análise dos dados transportados. Apresentam informações básicas das conexões e são detalhadas na Tabela 5.5;
- características sugeridas: são utilizados conhecimentos de especialistas para extrair informações do conteúdo dos pacotes transportados. Isso inclui, por exemplo, o número de tentativas de login com erros. São apresentados na Tabela 5.6;
- características de tráfego com janela de 2s: apresentam as características de tráfego calculadas no intervalo de tempo (2s). Um exemplo disso é o número de conexões destinadas a um mesmo nó da rede. Na Tabela 5.7 são apresentadas essas características;
- características de tráfego das últimas cem conexões: essas métricas apresentam o perfil histórico das últimas cem conexões realizadas. Um exemplo dessas métricas é a quantidade de conexões para uma mesma máquina destino, são apresentadas na Tabela 5.8.

Tabela 5.5- Características Básicas no KDD 99

Nome	***Descrição***	***Tipo***
duration	Duração da conexão em segundos	contínuo
protocol_type	Protocolo da camada de transporte (TCP, UDP)	discreto
service	Protocolo da camada de aplicação (HTTP, Telnet, etc)	discreto
src_bytes	Quantidade de bytes transmitidos da origem para o destinatário	contínuo
dst_bytes	Quantidade de bytes transmitidos do destinatário para a origem	contínuo
flag	Indica o estado da conexão (normal, erro)	discreto
land	Possui valor 1 se o número de porta origem e de destino são iguais e 0 caso diferentes	discreto
wrong_fragment	Número de fragmentos errados	contínuo

urgent	Indica o número de pacotes marcados como urgente	contínuo

Tabela 5.6- Características Sugeridas no KDD 99

Nome	***Descrição***	***Tipo***
hot	Indica o número de pacotes importantes	contínuo
num_failed_logins	Número de falhas de login	contínuo
logged_in	O valor 1 indica que o login foi efetuado com sucesso e o valor 0 caso contrário	discreto
num_compromised	Indica o número de conexões “comprometedoras”	contínuo
root_shell	O valor 1 indica que o shell do root foi obtido e 0 caso contrário	discreto
su_attempted	O valor 1 indica que houve tentativa de se obter o shell do root e o valor 0 caso contrário	discreto
num_root	Número de acessos como root	contínuo
num_file_creations	Indica o número de operações com criação de arquivos	contínuo
num_shells	Indica o número de shell abertos	contínuo
num_access_files	Indica o número de operações no controle de acesso de arquivos	contínuo
num_outbound_cmds	Número de comandos executados numa sessão de FTP	contínuo
is_hot_login	O valor 1 indica se o login pertence a uma lista importante e 0 caso contrário	discreto
is_guest_login	O valor 1 indica se o login foi executado como convidado e 0 caso contrário	discreto

Tabela 5.7- Características de Tráfego com Janela de 2s no KDD 99

Nome	***Descrição***	***Tipo***
count	Número de conexões par ao mesmo host nos últimos 2s	contínuo
serror_rate	Percentual de conexões com erros do tipo SYN para	contínuo

	o mesmo host	
rerror_rate	Percentual de conexões com erros do tipo REJ para o mesmo host	contínuo
same_srv_rate	Percentual de conexões ao mesmo serviço para o mesmo host	contínuo
diff_srv_rate	Percentual de conexões a serviços diferentes para o mesmo host	contínuo
srv_count	Número de conexões para o mesmo serviço de um mesmo host	contínuo
srv_serror_rate	Percentual de conexões com erros do tipo SYN para os mesmos serviços	contínuo
srv_rerror_rate	Percentual de conexões com erros do tipo REJ para os mesmos serviços	contínuo
srev_diff_host_rate	Percentual de conexões a diferentes hosts para os mesmos serviços	contínuo

Tabela 5.8- Características de Tráfego Calculadas Utilizando o Histórico das Últimas 100 Conexões no KDD 99

Nome	***Descrição***	***Tipo***
dst_host_count	Número de conexões para o mesmo host destino	contínuo
dst_host_srv_count	Número de conexões para o mesmo host destino e o mesmo serviço	contínuo
dst_host_same_srv_rate	Percentual de conexões para o mesmo host destino e o mesmo serviço	contínuo
dst_host_diff_srv_rate	Percentual de conexões de serviços diferentes para o mesmo host atual	contínuo
dst_host_same_src_port_rate	Percentual de conexões para o host atual e mesma porta	contínuo
dst_host_srv_diff_host_rate	Percentual de conexões para o mesmo serviço com diferentes hosts de origem	contínuo
dst_host_serror_rate	Percentual de conexões ao host atual com	contínuo

	um erro S0	
dst_host_srv_serror_rate	Percentual de conexões para o host atual e serviço especificado com um erro S0	contínuo
dst_host_rerror_rate	Percentual de conexões para o host atual que tiveram um erro RST	contínuo
dst_host_srv_rerror_rate	Percentual de conexões para o host atual e serviço especificado que tiveram um erro RST	contínuo

A base do KDD 99 é relativamente antiga, muitos trabalhos baseiam-se nessa base, a exemplo de (Dash e Liu, 2000), (Lippmann, Haines *et al.*, 2000), (Mukkamala, Janoski *et al.*, 2002) e (Depren, Topallar *et al.*, 2005). No entanto, essa base de dados tornou-se referência para análise de propostas de IDS, sendo utilizada em trabalhos recentes, (Tang, Cao *et al.*, 2008), (Michailidis, Katsikas *et al.*, 2008), (Mao, Lee *et al.*, 2009) e (Orfila, Estevez-Tapiador *et al.*, 2009), devido à sua elevada representatividade para essas análises.

A utilização da base do KDD não é direta, pois cada conexão é representada por um conjunto alfanumérico constituído pelos nomes dos protocolos, duração da conexão, classificação da conexão e outros dados. Um pré-processamento foi necessário para agrupar os dados em classes e convertê-los para numéricos. As categorias utilizadas para as conexões existentes na base são apresentadas na Tabela 5.9.

Tabela 5.9 - Classes das Conexões no KDD 99

Núm.	***Classe***	***Significado***	***Descrição***	***Nome dos Ataques***
1	U2R	Aumentar Privilégios	Acesso não autorizado com intuito de aumentar privilégios.	sqlattack, perl, xterm, rootkit, httptunnel, ps, buffer_overflow, loadmodule.
2	R2L	Remoto para Local	Tentativa de acesso remoto não autorizado, por exemplo, pelo uso de	ftp_write, guess_passwd, imap, phf, xsnoop, multihop, xlock, named, worm, warezmaster,

			quebra de senhas.	snmpguess, imap, guess_passwd.
3	Probe	Sondagem	Tentativa de identificar serviços ativos através de varreduras de portas.	ipsweep, nmap, portsweep, satan, saint, mscan.
4	DoS	Negação de Serviço	Caracteriza-se pelo envio de grande número de solicitações a um mesmo host, em curto período de tempo.	back, land, neptune, pod, smurf, teardrop, back, apache2, land, mailbomb, processtable, udpstorm.
5	Normal	Tráfego Normal	Indica que a conexão é baseada na utilização normal da rede	Não aplicado

5.4.2 Análise da Base do KDD 99 Através da Distribuição de Probabilidade de Poisson

Uma definição clássica da probabilidade estimada ou probabilidade empírica de um evento está relacionada com a frequência relativa de sua ocorrência. Se o número de observações for grande, a probabilidade propriamente dita será o limite da frequência relativa (Spiegel, Schiller *et al.*, 2000).

Em um processo aleatório, representado por qualquer fenômeno, os resultados finais podem ser definidos como eventos. A frequência relativa de um evento yi é definida na Equação 5.2.

(**5.2**)

$$\boldsymbol{F(y_i) = \frac{N(y_i)}{N}}$$

Onde:

$F(y_i)$ é a frequência do evento yi;

N é o número de eventos;

y_i é o iésimo evento.

Se o processo é repetido indefinidamente, com N→∞, a Equação (5.2) poderá ser reescrita como apresentado na Equação 5.3, transformando-se na probabilidade de ocorrência do evento y_i.

$$P(y_i) = \lim_{N\to\infty}\left(\frac{N(y_i)}{N}\right) = \lim_{N\to\infty} F(y_i) \quad \textbf{(5.3)}$$

A distribuição de Poisson é apresentada na Equação 5.4. Seu funcionamento prevê que a variável aleatória (Y), que representa os eventos, seja independente e expresse a probabilidade de ocorrência de certo número de eventos, num período de tempo, com a taxa média (μ) conhecida.

$$P_\mu(Y) = \frac{\mu^Y}{Y!}e^{-\mu} \quad \textbf{(5.4)}$$

Onde:

μ é a taxa média do evento;

e é a base do logaritmo natural.

O processo de contagem, utilizado pela distribuição de Poisson é relativamente simples e não envolve operações matemáticas complexas, o que torna leve em termos de custo computacional.

A análise da base do KDD 99, através da distribuição de Poisson foi realizada para cada conexão. Nesse caso, foi construído um vetor de probabilidade (VP) com cinco valores correspondentes a Função Densidade de Probabilidade da Distribuição de Poisson, calculada conforme a Equação 5.4. Cada posição do vetor indica a probabilidade da conexão que está sendo avaliada pertencer a cada uma das classes do KDD, a posição 1 contém a probabilidade da conexão ser da classe 1 (U2R), a posição 2 contém a probabilidade da conexão ser da classe 2 (R2L) até a última posição, que contém a probabilidade da conexão ser da classe 5 (Conexão Normal). A posição que contém o maior valor de probabilidade é escolhida para

classificar a conexão em avaliação. Essa escolha é comparada com a classificação original na base do KDD. Quando são iguais, significa que o IDS classificou corretamente e é atualizado o contador de acerto (TP - Verdadeiro Positivo ou TN – Verdadeiro Negativo), caso contrário, houve uma falha e o contador de erro é atualizado (FP – Falso Positivo ou FN – Falso Negativo). Esse algoritmo é apresentado na Figura 5.10

```
   Algoritmo AvaliarKDDPoisson
   Declare
    KDD – Base de dados original do KDD
    KDDN – Base de dados pré-processada do KDD
    fi[1..5] – Vetor de probabilidade das classes
    Classe – Identificação da classe
    EstatísticaCorreta – Contadores de acerto
(verdadeiro positivo ou verdadeiro negativo)
    EstatísticaErrada – Contadores de erro (falso
negativo ou falso positivo)
   Início
    KDD1 = Pré-Processamento(KDD)
    KDD2 = Selecionar_Características(KDD1)
    fi[1..5] = Média_Frequencia_Eventos_por_Tipo(KDD2)
    VP                                              =
Função_Densidade_Probabilidade_Poisson(KDD2,fi)
    IDS = Selecion_Maior_Elemento(VP)
    Se Classe no IDS for igual à Classificação em KDD1
     Então
          Atualiza(EstatísticaCorreta)
     Senão
          Atualiza(EstatísticaErrada)
    Fim (se)
   Fim
```

Figura 5.10 - Algoritmo para Análise da Base do KDD através de Poisson

Para o cálculo da Função Densidade de Probabilidade da Distribuição de Poisson, é necessário conhecer a taxa média dos eventos analisados, parâmetro μ da Equação 5.4. Para escolha desse parâmetro, foram selecionados dois subgrupos do KDD, com 10.000 e 100.000 conexões respectivamente. Em cada subgrupo, foram realizados testes variando o percentual de conexões utilizado para calcular a frequência dos eventos (10%, 20%, 30%, 40%, 50%, 70% e 100%). O tempo necessário para detecção de um ataque está relacionado com o tamanho do

conjunto de amostras utilizadas para contagem de eventos, quanto menor esse conjunto, mais rápido deverá ser a detecção de intrusão. Como mostrado na Figura 5.11, utilizando 30% das conexões para contagem de eventos, obteve-se a melhor taxa de detecção global de 97,57%. A utilização de um conjunto menor com apenas 20% das conexões também apresentou valor elevado de detecção.

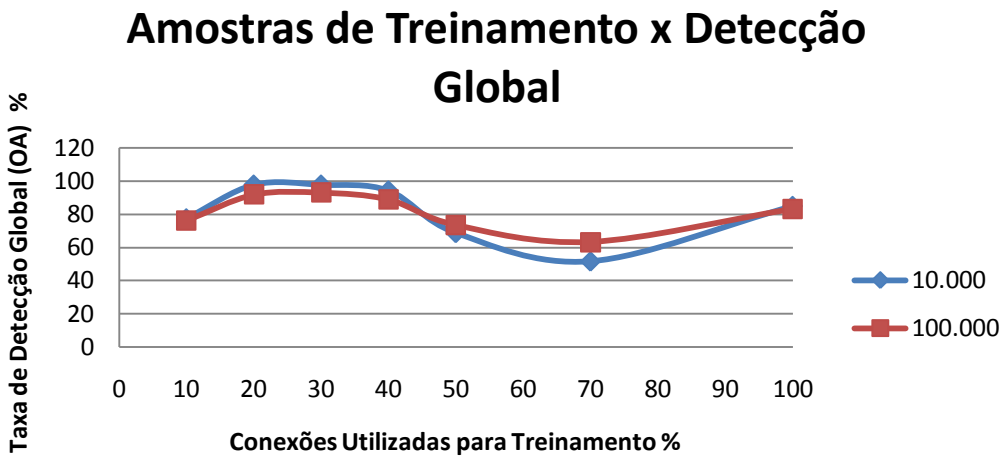


Figura 5.11 - Percentual de Amostras de Treinamento

Avaliações com outras distribuições de probabilidade também foram realizadas e os resultados estão apresentados na Tabela 5.10. Com o uso das distribuições Geométricas e Multinomial não foi possível identificar nenhum tipo de ataque, além disso, ambas as distribuições apresentaram elevada taxa de falso negativo. Nesses testes foram utilizadas 494.021 amostras e 30% delas para cálculo da frequência dos ataques.

Tabela 5.10 - Comparação com outras distribuições

Função	***TP %***	***TN %***	***FP %***	***FN %***	***AO %***	***Tempo s***
Poisson	75,07	14,23	5,45	0,66	93,57	543
Geométrica	0	19,69	0	80,3	19,69	182
Multinomial	0	19,69	0	80,3	19,69	567

Os ataques da classe *Probe* realizam sondagem no sistema alvo para identificar aplicações que estão em execução. Os ataques mais simples podem ser realizados através da tentativa de conexões em certas portas. Quando a conexão é

realizada com sucesso, pode-se concluir que determinada aplicação está disponível. Esse tipo de ataque gera vários pacotes que são transmitidos pela rede. A classe de ataque *DoS* é caracterizada pela geração de pacotes, que são encaminhados ao alvo, em quantidade muito elevada. Em ambos as classes de ataques, a rede recebe grande número de pacotes muito parecidos, em certo intervalo de tempo. Esse cenário propicia o uso da Distribuição de Poisson para modelagem do tráfego e a identificação dos ataques na rede.

A base do KDD 99 foi segmentada em dois conjuntos. O conjunto principal, ou base completa, possui 4.898.430 registros de conexões. O segundo conjunto é formado por aproximadamente 10% dos registros da base completa e geralmente é utilizado para realizar o treinamento dos algoritmos de detecção de intrusão. Nos experimentos realizados, percebeu-se que a utilização da base completa e da base reduzida (10% das conexões) apresentou resultados semelhantes, como mostrados na Tabela 5.11. Para melhorar o processamento, as avaliações seguintes foram executadas na base reduzida.

Tabela 5.11 Comparação entre Conjunto Completo e Parcial do KDD

Conjunto	Verdadeiro Positivo (TP)	Verdadeiro Negativo (TN)	Falso Positivo (FP)	Falso Negativo (FN)
KDD Completo	79,87%	14,18%	5,67%	0,06%
KDD 10%	75,07%	14,23%	5,45%	0,66%

5.4.3 Seleção de Características da Base do KDD 99 para o IDS

Para diminuir o custo computacional do IDS, o conjunto dos dados de auditoria deve ser reduzido. Existem muitas formas que podem ser empregadas para otimização desses dados, através da seleção de características que contribuem para a classificação das conexões. Dessa maneira, somente características relevantes são selecionadas para análise, com a formação de um subconjunto ótimo capaz de representar todos os dados. Os testes foram aplicados em 494.021 amostras de conexões do KDD, sendo 30% delas utilizadas para cálculo da frequência dos ataques. A precisão global foi obtida conforme a Equação 4.1.

Avaliações com subconjuntos diferentes, escolhidos a partir de diversos trabalhos proporcionaram resultados distintos, como apresentado na Tabela 5.12. Além disso, para comparação com esses trabalhos, constituiu-se um subconjunto com características escolhidas ao acaso, através de um sorteio. A escolha do subconjunto que não contribui para a classificação das conexões pode ser bastante prejudicial. Evidentemente, a utilização do conjunto completo apresenta melhor resultado. Porém, com as 13 características sugeridas em (Suebsing e Hiransakolwong, 2009), mesmo com a precisão global ligeiramente inferior, bons resultados foram obtidos, sendo esse o subconjunto ótimo utilizado. Ainda pela Tabela 5.12, é possível concluir que as características selecionadas estão relacionadas com o método empregado na detecção de intrusão. Um conjunto de características pode ser muito bom para determinado método e ruim para outro. Por exemplo, com o uso da proposta de (Ghali, 2009) houve redução de aproximadamente 85% do tempo de processamento e de 90% da taxa de erro. Entretanto quando tal conjunto de características foi utilizado na proposta deste artigo a precisão global de detecção foi reduzida para 35,43%. De acordo com essa informação, a escolha das características da base do KDD para formar um conjunto ótimo depende do método empregado para detecção de intrusão. As avaliações foram realizadas através da verificação baseada na Distribuição de Poisson.

Tabela 5.12 Precisão Global Conforme Seleção de Algumas Características do KDD

Proposta	*Método de Seleção*	*Características do KDD*	*Precisão Global (OA)*
	Todas as Características	Todas as características do KDD	95,95
(Suebsing e Hiransakolwong, 2009)	Euclidean Distance and Cossine Similarity	duration, protocol_type, logged_in, serror_rate, srv_error_rate, rerror_rate, srv_rerror_rate, diff_srv_rate, srv_diff_host_rate, dst_host_diff_srv_rate, dst_host_srv_diff_host_rate,	93,57

		dst_host_error_rate, dst_host_srv_rerror_rate	
(Li e Guo, 2008)	Transductive Confidence Machines for K-Nearest Neighbors	src_bytes, count, service, dst_host_same_src_port_rate, dst_host_srv_count, dst_host_same_srv_rate, logged_in	90,34
(Yu, Wu *et al.*, 2008)	Protocol Type and Logistic Regression	protocol_type, logged_in, count, srv_count, same_srv_rate, srv_diff_host_rate, dst_host_count, dst_host_same_src_port_rate, dst_host_srv_serror_rate	87,34
	6 Características Aleatórias	duration, su_attempted,is, is_guest_login, dst_host_count, dst_host_srv_serror_rate, dst_host_rerror_rate	80,37
(Khor, Ting *et al.*, 2009)	Correlation-based Fearure Selection Subset Evaluator (CFSE) and Consistency Subset Evaluator (CSE)	service, dst_bytes, logged_in, count, dst_host_count, root_shell, dst_host_rerror_rate	58,41
(Ghali, 2009)	Rogh Set Neural Network Algorithm (RSNNA)	src_byte, dst_byte, count, srv_count, dst_host_srv_count, dst_host_same_src_port_rate	35,43

É possível verificar a contribuição de cada característica no processo de classificação do tipo de conexão através do uso de histograma. Pode-se inferir que determinadas características contribuem mais para a classificação de certas classes. Na Figura 5.12 são apresentados os histogramas da característica *duration* (tempo

de duração da conexão). Pela análise visual, observa-se que existe uma dispersão do número de conexões em ataques tipo U2R, entretanto isso não acontece para as outras classes, as conexões estão agrupadas em torno de um único intervalo. Então, essa característica é uma candidata para formar um conjunto que permitirá classificar ataques tipo U2R, porém, não é recomendável utilizá-la para classificação das outras classes, pois provavelmente não propiciará contribuição significativa nesse processo. De forma similar, a mesma análise pode ser realizada com a característica *source bytes* (quantidade de bytes enviados da origem para o nó destino), como apresentado na Figura 5.13.

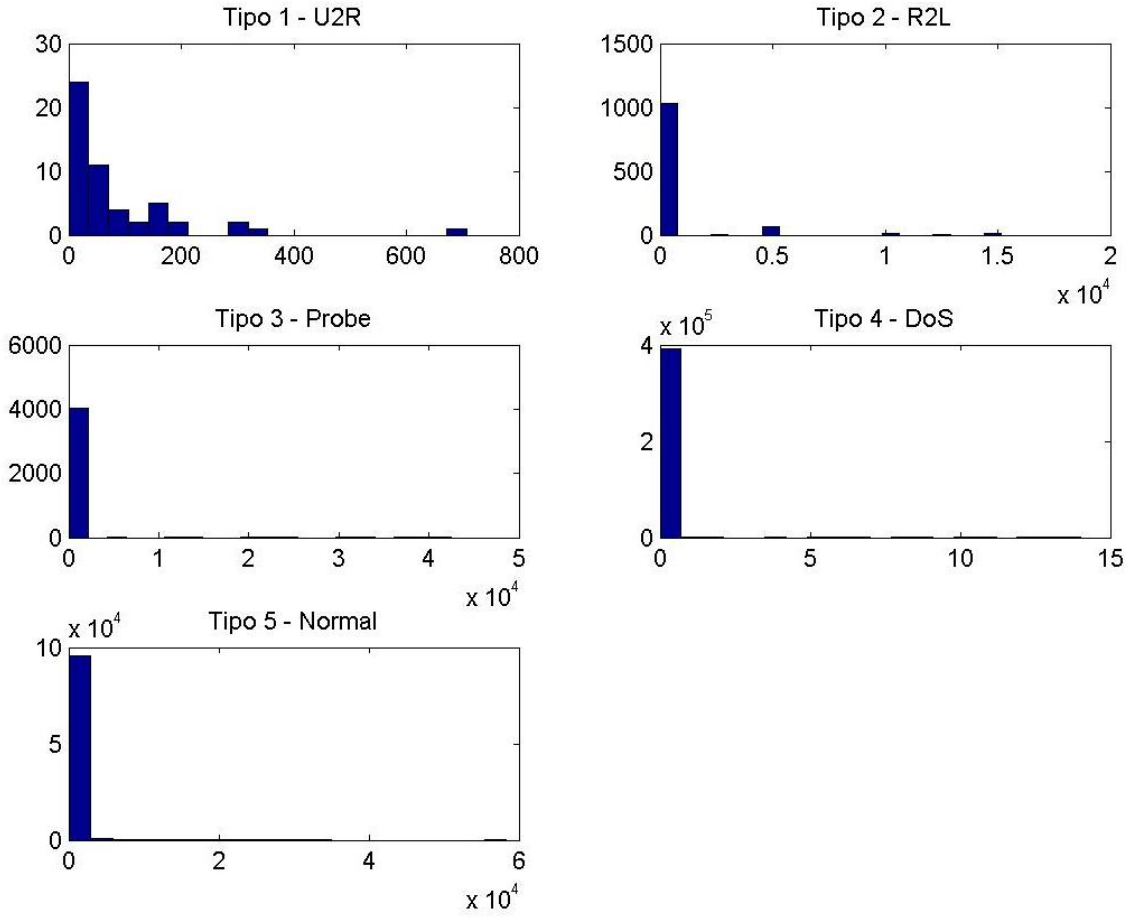


Figura 5.12 - Histograma da Característica "*duration*"

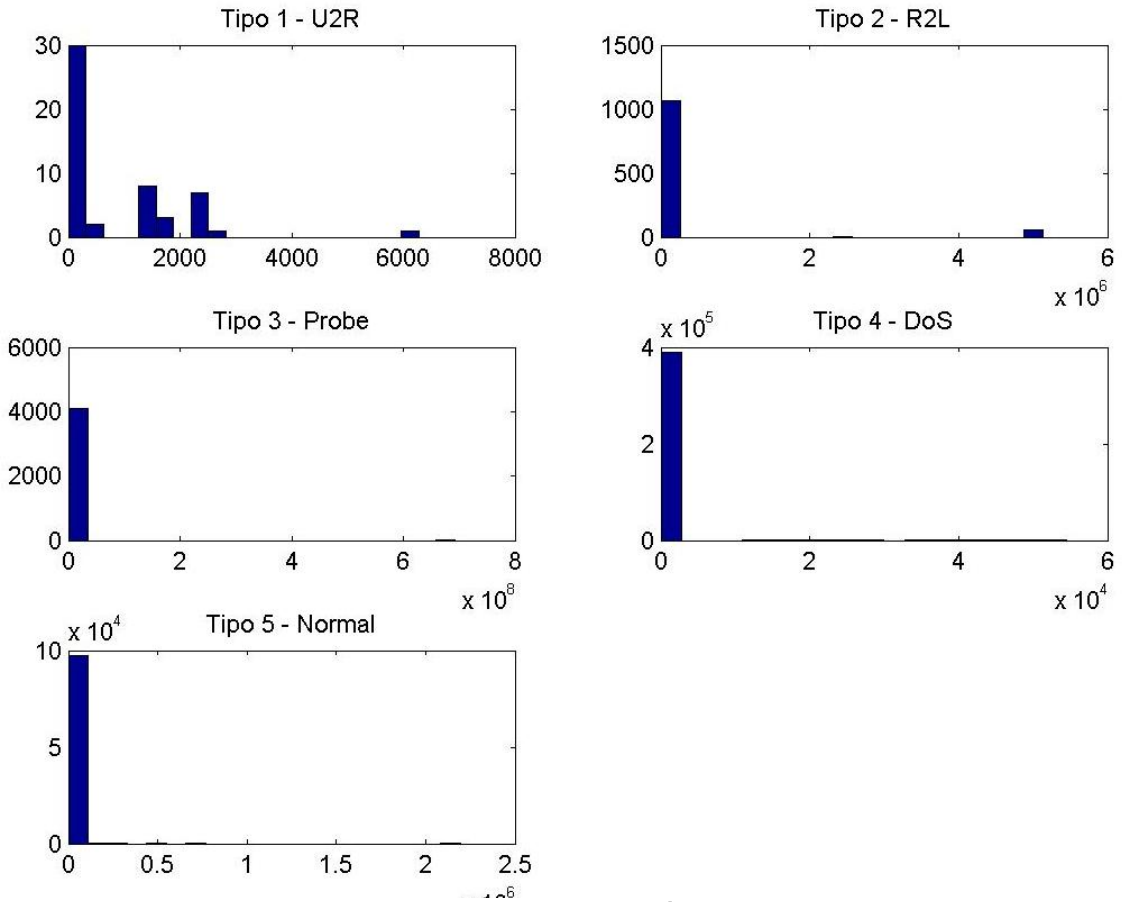


Figura 5.13 - Histograma da Característica "*source bytes*"

Existem características na Base do KDD 99 cujas conexões de todas as classes estão agrupadas em um pequeno intervalo. São apresentados na Figura 5.14 os histogramas da característica *is guest login* (indica se o login foi executado como usuário convidado). Nesse caso, essa característica não realiza contribuição significativa para o processo de classificação, então, é possível melhorar o desempenho com a exclusão desses dados.

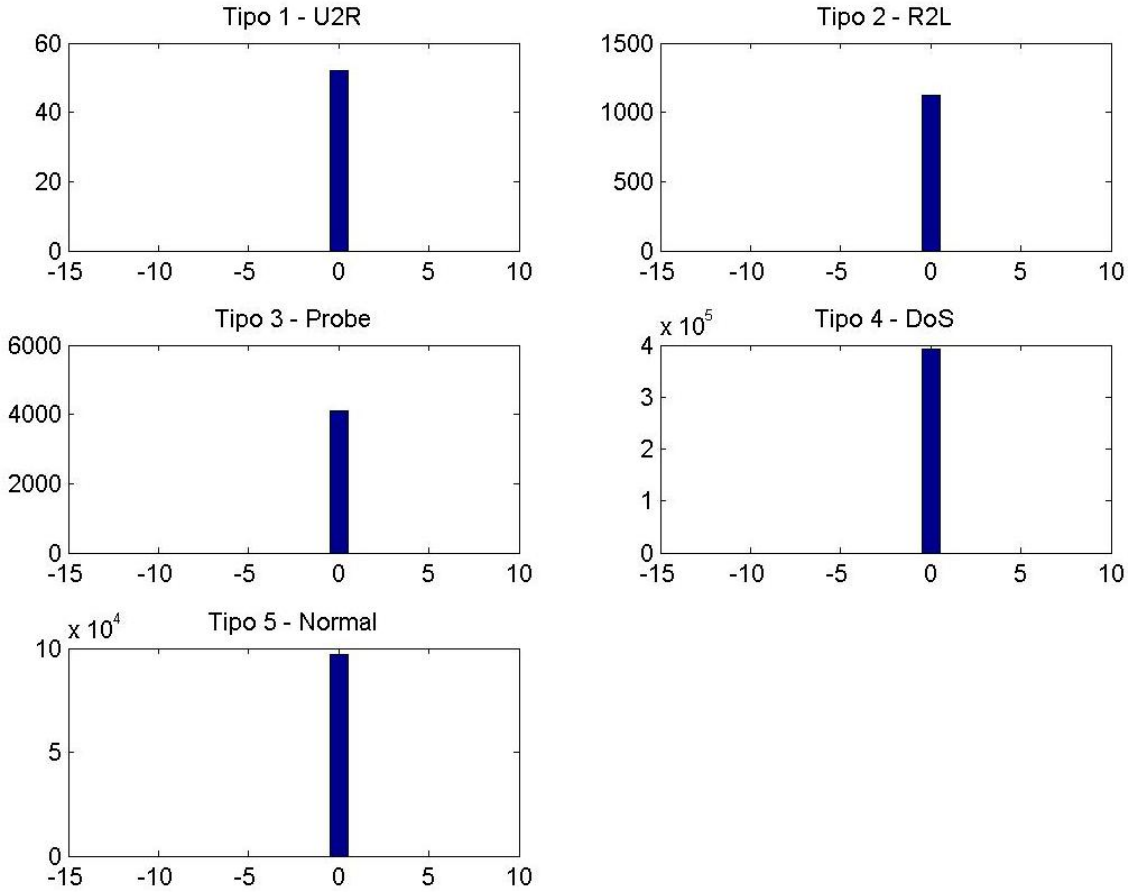


Figura 5.14 - Histograma da Característica "*is guest login*"

5.5 Resultados Obtidos

Foram realizados vários experimentos para avaliar a configuração da segunda camada do IDS, constituída pela rede neural. Todas as avaliações foram executadas em um computador equipado com Processador Intel Core 2 DUO T5500, com 4GB de RAM e Sistema Operacional Windows 7 64 bits. O cálculo da Precisão Global foi realizado conforme a Equação 5.5. A cada avaliação, a rede neural foi destruída e criada novamente para eliminar a aprendizagem obtida nos testes anteriores e a interferência de resultados entre os testes.

A primeira avaliação foi realizada com mil conexões para treinamento e aplicadas sobre 5.000 amostras. Esse conjunto foi formado por todas as 41 características da base do KDD 99. Os dados obtidos são apresentados na Tabela 5.13 e na Figura 5.15. O tempo gasto para o treinamento, em alguns casos, foi

elevado, porém em todos os testes, a detecção foi muito rápida, a classificação ocorreu com tempo inferior a um segundo.

Tabela 5.13 - Precisão Global (1.000 amostras para treinamento 5.000 para testes)

Época	*AO %*	*Tempo Treinamento (s)*	*Tempo de Teste (s)*
50	99,800	70,00	0,14
100	99,980	134,00	0,14
200	99,880	271,00	0,15
300	99,960	415,00	0,14
500	71,311	712,00	0,17
1.000	99,720	1.343,00	0,15
1.200	99,780	18,15	0,19
1.500	99,980	2.032,00	0,14
2.000	99,820	2.663,00	0,14

Apenas no treinamento com 500 épocas a rede neural não convergiu para o resultado satisfatório. As redes neurais são sistemas muitos complexos e dinâmicos, apesar de seu alto poder de simulação, muitas vezes não converge para uma única solução ou, quando converge, consome muito tempo com as iterações (Capuano, 2009). Nesses casos, torna-se necessário realizar testes para escolha de parâmetros adequados (época, número de camadas, número de neurônio das camadas e outros) que apresentem os melhores resultados.

A quantidade de dados disponíveis na Base do KDD é muito superior ao conjunto utilizado para esse teste, portanto, os resultados não são significativos. São necessárias outras análises para avaliar os resultados e sugerir a configuração adequada para a rede neural artificial.

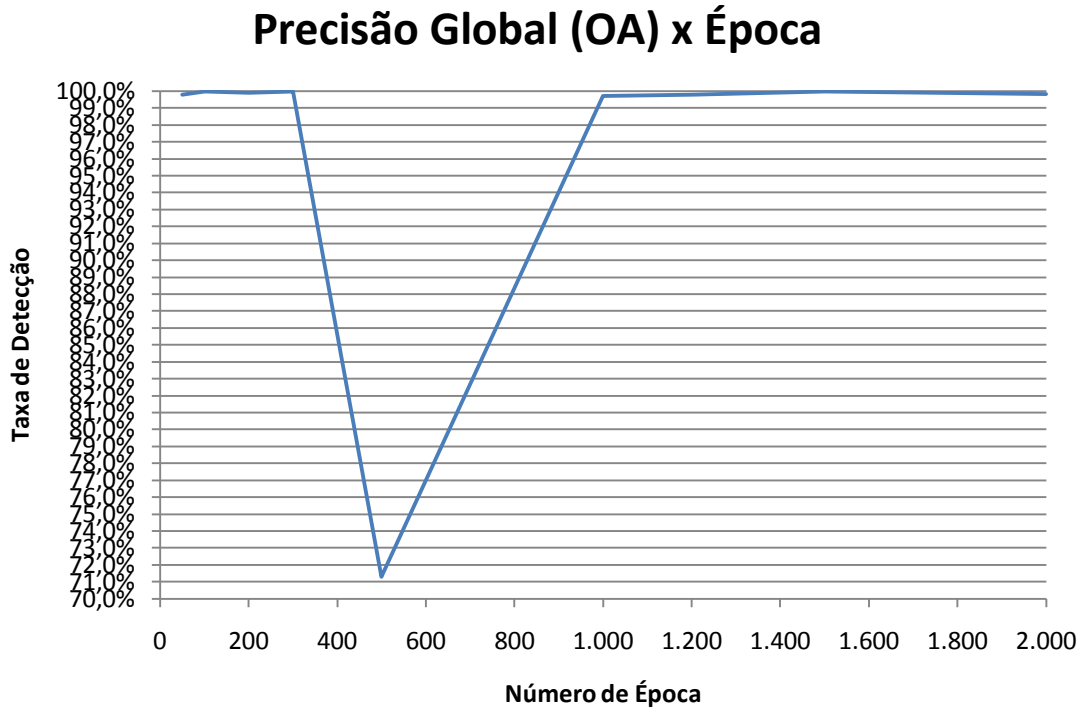


Figura 5.15 - Precisão Global (1.000 amostras para treinamento e 5.000 para testes)

A avaliação com 10.000 conexões utilizadas para o treinamento da rede neural e aplicadas sobre 50.000 amostras de testes produziram os resultados apresentados na Tabela 5.14 e Figura 5.16. Com a utilização dos valores 200 e 500 para a época a rede neural, durante o treinamento, não convergiu para resultado esperado.

Tabela 5.14- Precisão Global (10.000 amostras para treinamento, 50.000 conexões para testes)

Época	*OA*	*Tempo Treinamento (s)*	*Tempo de Teste (s)*
50	93,1%	728,00	1,30
100	84,5%	1.449,00	1,35
200	39,0%	272,00	1,35
300	87,5%	4.335,00	1,38
500	80,2%	889,00	2,02
1.500	87,8%	21.558,00	1,31
2.000	63,4%	1.065,00	1,60

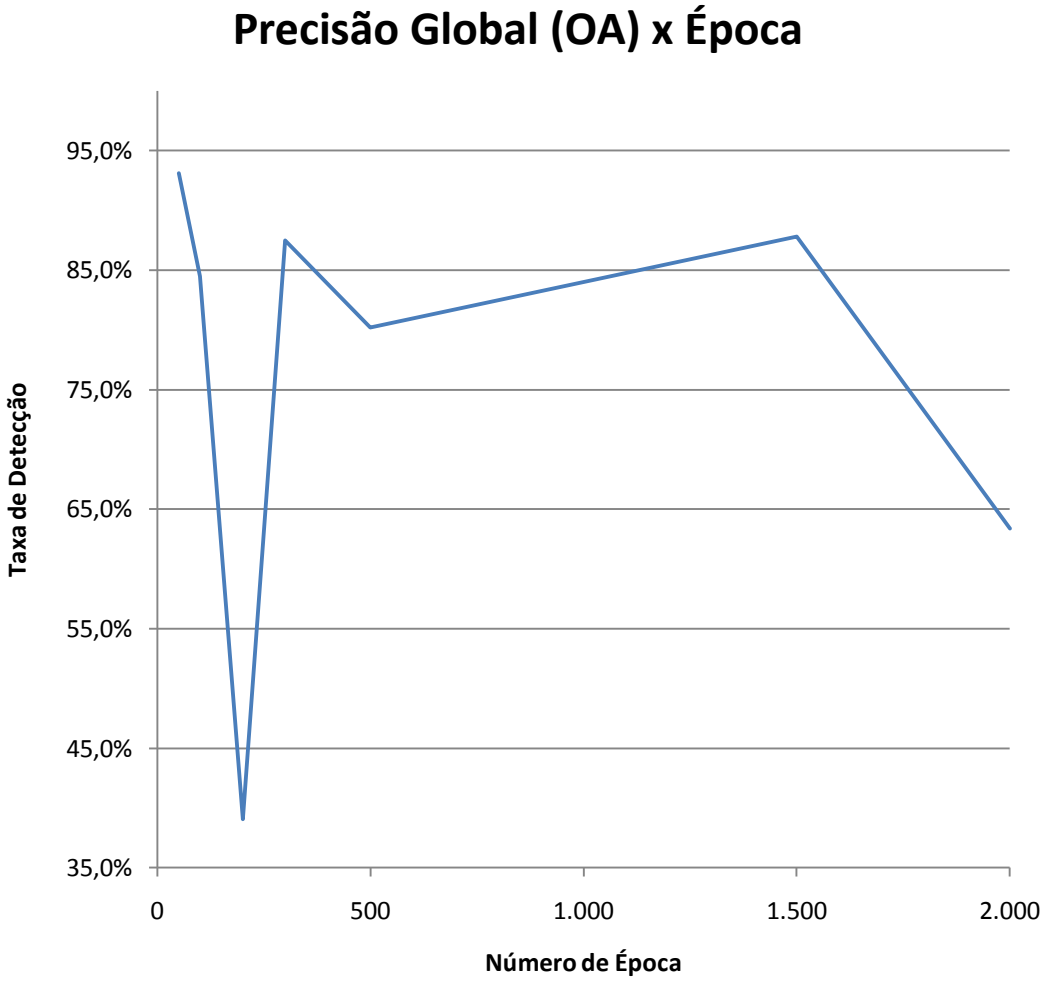


Figura 5.16 - Precisão Global (10.000 amostras para treinamento, 50.000 conexões para testes)

Foi realizada a avaliação da abordagem proposta aplicado sob um conjunto parcial dos atributos existentes na base do KDD. Os atributos foram escolhidos conforme sugestão de alguns trabalhos. Os atributos do "Histograma" foram selecionados a partir da análise visual dos histogramas das classes e esse conjunto é formado por: *duration, protocol_type, service, flag, dst_bytes, hot, num_compromised, root_shell, num_file_creations, count, srv_cont, dst_host_count, dst_host_srv_count. As descrições dos atributos estão apresentadas na Tabela 5.12 e o resultado da análise na Tabela 5.15. O MSE, ou erro quadrático médio é calculado conforme a Equação 5.6.*

Tabela 5.15 – Precisão Global Conforme Seleção de Alguns Atributos

TP	*TN*	*FP*	*FN*	*AO %*	*Atributos*	*Número de Atributos*	*MSE*
31,10	62,07	0,95	5,77	93,27	(Khor, Ting *et al.*, 2009)	7	3,00E-08
30,92	62,58	0,44	5,86	93,69	Histograma	13	3,00E-10
18,15	62,91	0,10	18,81	81,08	(Li e Guo, 2008)	7	1,00E-12
0,02	62,96	0,05	36,97	62,98	Aleatório	6	2,00E-08
0,00	33,64	29,37	0,88	52,65	(Ghali, 2009)	7	8,00E-08
25,14	62,74	0,28	11,75	87,96	(Fries, 2008)	8	9,37E-06
26,17	62,89	0,12	10,77	89,10	(Yu, Wu *et al.*, 2008)	9	9,14E-06
0,00	62,98	0,03	36,97	62,99	(Suebsing e Hiransakolwong, 2009)	15	6,00E-07

Nesse caso é percebido que a quantidade de atributo selecionada não está relacionada com a Precisão Global da Detecção. A proposta que possui maior número de atributos da base do KDD foi a que apresentou pior resultado no cenário avaliado.

$$MSE(f,x) = \frac{1}{n}\sum_{i=1}^{n}(f_i - x_i) \tag{5.5}$$

Onde:

n é o número de amostras;

f_i é o valor da i-ésima observação;

x_i é o valor calculado para a i-ésima observação.

Nesse experimento, as conexões, com todos os atributos, foram separadas em grupos e organizadas conforme sua classificação. Para cada grupo, foram calculados os coeficientes da wavelet e esses foram apresentados para treinamento

da rede neural. Ao término do treinamento, foram realizados testes de classificação e reconhecimento das conexões, conforme a estrutura apresentada na Figura 5.17.

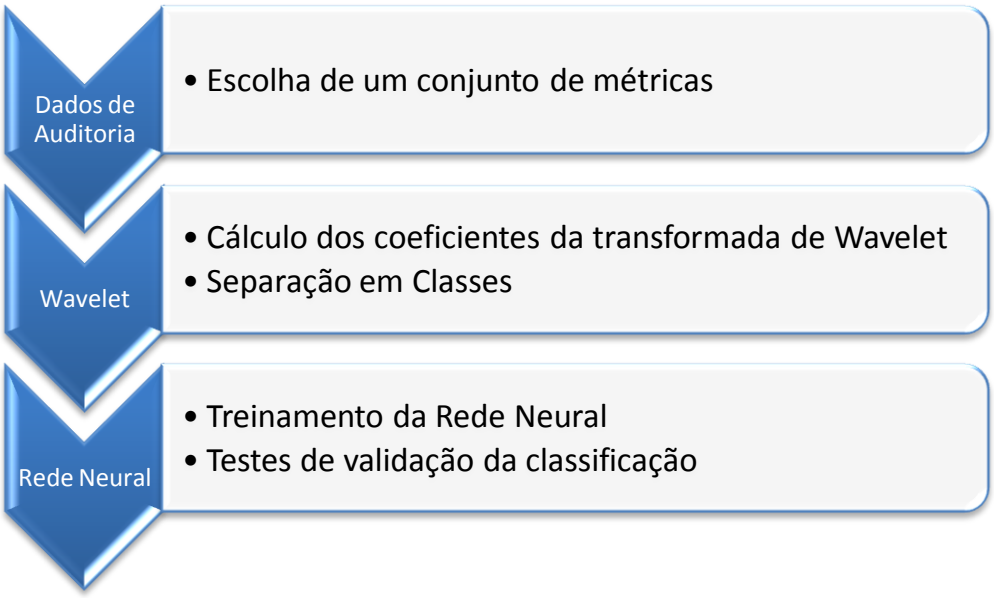


Figura 5.17 - Estrutura de Teste

Com essa estrutura aplicada sobre todas as conexões da base de dados, os seguintes resultados foram obtidos: o treinamento utilizando 10.000 conexões, a Precisão Global foi de 43%, porém, com o aumento do número de amostras, através do treinamento da rede neural com 120.000 amostras e aplicados sobe 400.000 conexões, a Precisão Global aumentou para 80%.

Experimentos realizados utilizando cinco redes neurais distintas, onde cada rede foi treinada para classificar um tipo de tráfego (U2R, L2R, *Probe*, *DoS* e Normal) foram realizados. Os valores da Precisão Global foram semelhantes aos resultados quando utilizado apenas uma rede neural com cinco neurônios na camada de saída.

5.6 Comparação dos Resultados com Outras Abordagens Propostas

Existem diversas propostas para construção de sistemas de detecção de intrusão, com abordagens diferentes. A fim de realizar uma comparação entre resultados obtidos, algumas propostas foram selecionadas e apresentadas na

Tabela 5.16, juntamente com a taxa de acerto. Nessas propostas, não fica claro se os autores utilizaram a taxa de acerto como verdadeiros positivos ou precisão global, porém, todos fizeram uso da base de dados do KDD 99. As propostas baseadas em redes neurais artificiais foram avaliadas utilizando 30.000 amostras para treinamento e 10.000 épocas, conforme explicado nos artigos. A taxa de acerto relacionada à proposta dessa tese foi substituída pelo melhor resultado obtido através da taxa da precisão global. Ainda pela Tabela 5.16, é possível perceber que a proposta aqui apresentada proporcionou a melhor taxa de acerto, demonstrando que trata-se de uma abordagem viável para construção de IDS.

Tabela 5.16 - Comparação entre Diversos Trabalhos

Proposta	***Taxa de Acerto***
Proposta desse trabalho	99,98%
Polvo-IIDS (Mafra, Da Silva Fraga *et al.*, 2008)	96,55%
IDS Wavelet (Lu e Ghorbani, 2009)	94,67%
Anomalous Payload-based IDS (Bolzoni, Zambon *et al.*, 2006)	93,70%
Multi-level Hybrid Classifier (Xiang e Lim, 2005)	89,19%
MADAM ID (Lee e Stolfo, 2000)	77,97%
HPCANN (Liu, Yi *et al.*, 2007)	77,49%

5.7 Comparação dos Resultados com a Aprendizagem por Quantização Vetorial

A aprendizagem por quantização vetorial é percussora dos mapas auto-organizáveis ou mapas de *Kohonen*. Essa arquitetura pode ser considerada um tipo especial de redes neurais (Monteiro, 2008). A quantização vetorial é utilizada em diversas áreas de conhecimento, a exemplo de combate a spam (Chuan, Xianliang *et al.*, 2005), classificação de imagens médicas (Chalabi, Berrached *et al.*, 2008), reconhecimento de gestos (Tolba, Elsoud *et al.*, 2009) e IDS para hosts (Marin, Ragsdale *et al.*, 2001), entre outras.

A LVQ é constituída por uma classe de algoritmos de mesmo nome: LVQ1, LVQ2 e LVQ3 que descreve uma aprendizagem supervisionada. Como o LVQ depende de uma classificação estatística ou de um método de reconhecimento, seu

propósito é definir regiões de classes no espaço de dados de entrada, semelhante aos algoritmos de *clustering*. Para isso, um subconjunto de vetores de referência é utilizado (Monteiro, 2008).

A LVQ é baseada no método do vizinho mais próximo. É calculada a menor distância entre vetores desconhecidos, que serão classificados, dos vetores de referência. Nesse processo de treinamento, apenas o vetor de referência mais próximo é atualizado. Essa atualização é realizada tanto nas classificações corretas como também nas incorretas. Como resultado, os vetores de referência são aproximados por funções de densidade de probabilidade dos padrões das classes. Os vizinhos mais próximos definem superfícies de decisão entre as classes, semelhante ao classificador de *Bayes*. O algoritmo LVQ1, apresentado por (Yang, Chen *et al.*, 2007) é mostrado na Figura 5.18.

O algoritmo LVQ2 é muito parecido com o LVQ1, a decisão de classificação são idênticas em ambos. Contudo, na aprendizagem, dois vetores de referência (ou *codebooks*) m_i e m_j, que são vizinhos mais próximos de x são simultaneamente atualizados. Um deles deve pertencer à classe correta e o outro à classe incorreta, respectivamente. Contudo, x precisa estar numa zona de valores, chamada *janela*, que é definida em volta do plano m_i e m_j. Considerando que d_i e d_j são as distâncias euclidianas de x a m_i e m_j, respectivamente, então x é definido para cair na *janela* de largura w se (Sanchez, Pla *et al.*, 1999), como apresentado na Equação 5.6.

Algoritmo LVQ

Declare

x – vetor de entrada (do conjunto de treinamento)

w_i – o i-ésimo vetor de referência (ou *codebook*): $w_i \in R^N$

T – número total de iterações de aprendizagens

$w_c(t)$ – valor seqüencial de w_c no domínio discreto de tempo (t =0,1,2,...)

Início

Encontre c, vencedor do processo de competição através da equação:

$$\|x - w_c\| = \min_i(\|x - w_i\|)$$

Ajuste w_c

$$w_c(t+1) = \begin{cases} w_c(t) + s(t)\alpha(t)[x - w_c], i = c \\ w_i(t), i \neq c \end{cases}$$

Em que:

$$s(t) = \begin{cases} 1, se\ a\ classificação\ está\ correta \\ -1, se\ a\ classificação\ está\ errada \end{cases}$$

$$\alpha(t) = \alpha_0(1 - t/T)$$

Onde $0<\alpha_0<1$, e T é o número total

Fim

Figura 5.18 - Algoritmo LVQ

$$\min\left(\frac{d_i}{d_j}, \frac{d_j}{d_i}\right) > s, onde\ s = \frac{1-w}{1+w} \quad \textbf{(5.6)}$$

Uma janela com largura relativa de *w* de 0,2 a 0,3 é recomendada (Monteiro, 2008). Uma versão do LVQ2 chamada LVQ2.1 possibilitou uma melhoria no algoritmo original, pois permite que m_i e m_j sejam os vizinhos mais próximos de *x*, considerando que no LVQ2 original m_i tem de ser o mais próximo. Os ajustes são apresentados na Equação 5.7.

$$m_i(t+1) = m_i(t) + \propto (t)[x(t) - m_i(t)] \quad \textbf{(5.7)}$$
$$m_j(t+1) = m_j(t) - \propto (t)\left[x(t) - m_j(t)\right]$$

O algoritmo LVQ3 difere do LVQ1 na forma de como os vetores de referência são atualizados. Assumindo que *x(t)* caia dentro de uma janela entre dois clusters adjacentes com vetores de referência y_i e y_j. Supondo que x e y_j pertencem a mesma classe, e x e y_i pertencem a classes diferentes, ambos os vetores serão atualizados conforme a Equação 5.8 (Hu, Palreddy *et al.*, 1997).

$$y_i(t+1) = y_i(t) - \propto (t)[x(t) - y_i(t)] \quad \textbf{(5.8)}$$
$$y_j(t+1) = y_j(t) + \propto (t)\left[x(t) - y_j(t)\right]$$

Por outro lado, se ambos y_i e y_j pertencem a mesma classe que *x(t)*, e *x(t)* deve estar entre os valores definidos da janela, então os valores dos novos vetores serão calculados conforme a Equação 5.9 (Hu, Palreddy *et al.*, 1997). Os valores ótimos de ε depende do tamanho da janela, sendo menor para janelas mais estreitas.

$$y_k(t+1) = y_k(t) - \varepsilon \propto (t)[x(t) - y_k(t)], \quad \textbf{(5.9)}$$
$$k \in \{i, j\} \; e \; 0{,}1 < \varepsilon < 0{,}5.$$

Com objetivo de realizar comparações entre a abordagem proposta e os algoritmos baseados no LVA, foi utilizada a base do KDD 10 no ambiente Weka. O Weka é um conjunto de softwares para aprendizado de máquina que implementa diversos algoritmos e técnicas de mineração de dados na linguagem Java (Holmes, Donkin *et al.*, 1994). A separação de parte dos dados para treinamento e o restante para testes foi realizada de duas formas: validação cruzada (*cross-validation folds* 10) e separação percentual (*percentage split* 66). A validação cruzada separa a base de dados em dez subconjuntos. Nove são utilizados para o treinamento e um subconjunto para avaliação. A separação em percentual separa 66% da base para treinamento e o restante para os testes. Em ambos os casos, a separação dos conjuntos é realizada de forma aleatória. Os resultados obtidos são apresentados na Tabela 5.17.

Tabela 5.17 - Resultados com LVQ

	U2R %	*R2L %*	*Probe %*	*DoS %*	*Normal %*
LVQ1 VC	0	0	0	86,4	98,8
LVQ1 SP	0	0	0	82,1	99,1
LVQ2 VC	0	0	0	89,0	97,1
LVQ2 SP	0	0	7,4	92.1	99,2
LVQ3 VC	0	0	0	87,1	98,3
LVQ3 SP	0	0	5,8	88,3	98,7

VC: validação cruzada; SP: separação em percentual.

O melhor resultado obtido foi utilizando LVQ2, com 99,2% de classificação correta de conexões do tipo Normal. Nessa avaliação, 89,63% das instâncias foram classificadas corretamente e 10,36% de forma incorreta. Esses valores são bem inferiores dos que foram obtidos pela abordagem proposta nessa tese, pois o melhor resultado obtido foi de 99,98% de precisão global. Com a utilização do LVQ não foi possível classificar conexões das classes U2R e R2L. Essas classes possuem número reduzido de conexões quando comparadas com as classes DoS e Normal, como mostrada na Figura 5.19, é provável que essa grande diferença entre as quantidades de conexões das classes influenciaram a classificação.

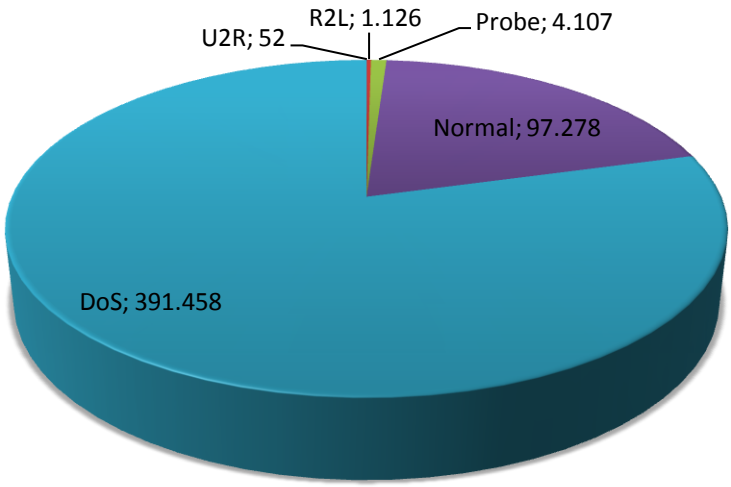


Figura 5.19 - Quantidade de Conexões por Tipo no KDD10

5.8 Comentários Finais

Nesse capítulo foi apresentada uma proposta de IDS híbrido, de duas camadas, baseado em transformadas wavelets e redes neurais artificiais, para detecção e classificação de anomalias, causados por ataque à rede de computadores.

A análise por wavelet permite divisões sucessivas em aproximação e detalhe. Esse método possui a capacidade de ajuste adaptativo e pode detectar anomalias de baixa, média e alta intensidade. Mesmo pequenas anomalias ao longo do tempo poderão ser identificadas através de seu uso.

É necessário o treinamento da rede neural antes de sua utilização. Após essa fase, o reconhecimento de padrões, que indicam o ataque é simples e rápido. Quando um novo tipo de ataque é descoberto, poderá ser necessário o treinamento novamente da rede.

A abordagem híbrida permite compartilhar as melhores características de cada método. Além disso, o uso em conjunto permite a redução de falso-positivos, se comparado com a utilização isolada da primeira camada.

As avaliações foram executadas num ambiente simulado e numa rede em laboratório, ambas ad hoc sem fio. Além disso, também foi empregado em uma base de testes, resultado da captura de dados de uma rede ethernet com fio que é muito utilizada para análise de IDS. Os resultados obtidos aqui permitem concluir que a abordagem proposta é muito promissora, e um bom nível de detecção foi conseguido nas avaliações realizadas.

6 CONCLUSÕES E TRABALHOS FUTUROS

Nesse trabalho foi proposto o uso de duas técnicas diferentes: transformadas wavelets e redes neurais artificiais para constituir um IDS para redes de computadores. Nessa abordagem, a primeira técnica foi utilizada para avaliar o estado da rede, com a indicação do funcionamento normal ou se a mesma está sob uma condição anômala, com o indício de ataque. A rede neural foi empregada para classificar o ataque conforme as categorias adotadas (U2R, L2R, *Probe, DoS* e Normal). A proposta aqui sugerida foi utilizada em cenários distintos, com dados obtidos a partir de simulação, testes em uma rede em laboratório e na base de treinamento e avaliação do KDD 99. Em todos os casos avaliados, foram obtidos bons níveis de detecção.

As transformadas de wavelet possuem propriedades que podem ser empregadas para indicar situações anormais em uma rede de computadores. A análise dos coeficientes gerados permite inferir que a rede está sob a condição de um ataque, mas não é possível indicar a classe do mesmo. O emprego em conjunto com um mecanismo de classificação complementa o funcionamento em um IDS.

A avaliação da rede neural ocorreu em dois cenários distintos. No primeiro, foram utilizados os mesmos parâmetros de entrada da wavelet, os dados obtidos diretamente das conexões, sem modificações ou tratamento. No segundo cenário, a rede neural foi treinada com os coeficientes gerados pela wavelet, as conexões foram separadas por classes, e após o cálculo dos coeficientes, um subconjunto foi utilizado para o treinamento da rede neural e outro para os testes. A precisão global foi calculada em ambos os casos. O resultado obtido no primeiro cenário foi melhor do que segundo.

A escolha da arquitetura da rede neural aconteceu com base em dois testes realizados. Primeiro foram criadas e treinadas cinco redes neurais do tipo MLP com apenas um neurônio na camada de saída. Cada rede foi concebida para reconhecer uma classe de tráfego. Nessa arquitetura, os dados de auditoria eram apresentados para as cinco redes, de forma independente. No segundo caso, apenas uma rede neural do tipo MLP foi criada, porém, a camada de saída foi constituída por cinco neurônios. A rede foi treinada para ativar um determinado neurônio conforme a classe do ataque. O primeiro caso possui implementação mais simples, porém, cada rede é testada individualmente, enquanto que o segundo caso, a implementação é

mais complexa, mas a avaliação pela rede neural é executada uma única vez. Em ambos os cenários, os resultados obtidos foram praticamente idênticos. É possível concluir que ambas as arquiteturas podem ser empregadas, porém, a segunda foi escolhida por apresentar melhor desempenho.

Em determinadas situações, é muito difícil o emprego de redes neurais artificiais. Em um cenário específico, através do uso de 10.000 conexões do KDD 99 para treinamento e parâmetros como duzentos e quinhentos para a época, não houve resultados satisfatórios, pois a rede neural não convergiu no processo de aprendizado. A investigação para descobrir a razão da não convergência do treinamento da rede neural é muito difícil, afinal, muitos parâmetros são envolvidos. Nesse caso, foi necessária a utilização de conjuntos de treinamentos com tamanhos diferentes, isso possibilitou o funcionamento correto da rede neural.

É esperado que a maior parte do tempo uma rede de computadores funcione em condição normal, que os ataques aconteçam em durante momentos em pequenos períodos. É interessante então haver um mecanismo que indique condições anômalas na rede, mas que consuma baixo poder computacional. Havendo um indício de ataque, outro método pode ser empregado para sua classificação, nesse caso poderá consumir mais recursos. Dessa forma, a abordagem proposta com o uso de wavelets e redes neurais na construção do IDS é a principal contribuição desse trabalho.

Os trabalhos futuros podem ser realizados de diversas maneiras:

- aplicação em rede sem fio: avaliação da abordagem proposta em uma rede sem fio real. Com a mudança da camada de enlace, são esperados ataques específicos dessa arquitetura, poderá ser necessário o uso de atributos obtidos na camada de enlace para detecção e classificação dos ataques;
- detecção *on line*: adaptação dessa proposta para avaliações instantâneas na rede. Para realizar esse tipo de atividade, o código deverá ser escrito e compilado numa linguagem que permite alto desempenho, como por exemplo, C++, visto que o Matlab é adequado à criação de protótipos;
- substituição da rede neural: as redes neurais necessitam de certo poder computacional na fase de treinamento. A adoção de outras

técnicas para classificação dos ataques poderá acelerar o processamento do IDS; e

- integração com sistema de prevenção de ataques: na abordagem proposta, o IDS indica se a rede está sob condição normal ou ataque. Se houver uma situação de ataque, o administrador deverá realizar ações para anular ou minimizar os danos. Futuramente poderá ser realizada a integração com um sistema de prevenção de ataques, nesse caso, ações automáticas poderão ser executadas sem a intervenção manual de um administrador, para garantir o funcionamento correto da rede.

REFERÊNCIAS BIBLIOGRÁFICAS

Ahmad, I., M. A. Ansari*, et al.* Performance Comparison Between Backpropagation Algorithms Applied to Intrusion Detection in Computer Network Systems: World Scientific and Engineering Academy and Society (WSEAS) Stevens Point, Wisconsin, USA, 2008. 47-52 p.

Akl, R. e A. Arepally. Dynamic Channel Assignment in IEEE 802.11 Networks. IEEE Portable Information Devices, 2007: IEEE 2007.

Akyildiz, I. e X. Wang. A Survey on Wireless Mesh Networking. Communications Magazine, IEEE. 43**:** New York p. 2005.

ANSI/IEEE Std 802.11, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (R2003). Piscataway: IEEE Computer Society. 1999

ANSI/IEEE Std 802.11a, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications (R2003). Piscataway: IEEE Computer Society. 1999

ANSI/IEEE Std 802.11b, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: Higher-Speed Physical Layer Extension in the2.4 GHz Band (R2003). New York: IEEE Computer Society. 1999

ANSI/IEEE Std 802.11g, Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications Amendment 4: Further Higher Data Rate Extension in the 2.4 GHz Band (R2003). New York: IEEE Computer Society. 1999

Axelsson, S. Intrusion Detection Systems: A Survey and Taxonomy. Chalmers University, p.1-27. 2000

Bellur, B. e R. G. Ogier. A Reliable, Efficient Topology Broadcast Protocol for Dynamic Networks. IEEE INFOCOM '99 Proceedings 1999.

Bolzoni, D., E. Zambon*, et al.* POSEIDON: a 2-tier Anomaly-based Network Intrusion Detection System. Proceedings of the 4th IEEE International Workshop on Information Assurance (IWIA), 2006. 144-156 p.

Burmester, M., T. V. Le*, et al.* Adaptive gossip protocols: managing security and redundancy in dense ad hoc networks. Ad Hoc Networks. 5**:** 313-323 p. 2007.

Cabrera, J. B. D., B. Ravichandran*, et al.* Statistical Traffic Modeling for Network Intrusion Detection. 8th International Symposium on Modeling, Analysis and Simulation of Computer and Telecommunication Systems 2000. 466-473 p.

Capuano, E. A. O Poder Cognitivo das Redes Neurais Artificiais Modelo Art1 na Recuperação da Informação. Ciência da Informação, v.38, n.1, p.9-30. 2009.

Centro De Estudos Resposta E Tratamento De Incidentes De Segurança No Brasil. Estatística dos Incidentes Reportados ao CERT.br. 2009 2009.

Chalabi, Z., N. Berrached*, et al.* Classification of the Medical Images by the Kohonen Network SOM and LVQ. Journal of Applied Sciences, v.8, n.7, p.1149-1158. 2008.

Chen, H. e M. Guizani. Next Generation Wireless Systems and Networks. London: John wiley & Sons Inc. 2006

Cheriet, M., N. Kharma*, et al.* Character Recognition Systems - A Guide for Students and Practioners. Hoboken: John Wiley & Sons, Inc. 2007

Chuan, Z., L. Xianliang*, et al.* A LVQ-Based Neural Network Anti-Spam Email Approach. ACM SIGOPS Operating Systems Review, v.39, n.1, p.34-39. 2005.

Clausen, T. e P. Jacquet. Optimized Link State Routing Protocol (OLSR): RFC 3626. 2003

Comer, D. E. Internetworking with TCP/IP - Volume I - Principles, Protocols and Architecture. New Jersey: Prentice Hall. 1995

Dash, M. e H. Liu. Feature Selection for Clustering. Lecture Notes in Computer Science, p.110-121. 2000.

Depren, O., M. Topallar*, et al.* An Intelligent Intrusion Detection System (IDS) for Anomaly and Misuse Detection in Computer Networks. Expert Systems With Applications, v.29, n.4, p.713-722. 2005.

Douceur, J. R. The Sybil Attack. In: (Ed.). Lecture Notes in Computer Science. Berlin: Springer, 2002. The Sybil Attack

Elman, J. L. Learning and Development in Neural Networks: The Importance of Starting Small. Cognition, v.48, n.1, p.71-99. 1993.

Fernandes, N. C., M. D. D. Moreira*, et al.* Ataques e Mecanismos de Segurança em Redes Ad Hoc. In: (Ed.). Minicursos do Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais - SBSeg'2006. Santos, 2006. Ataques e Mecanismos de Segurança em Redes Ad Hoc, p.49-102

Fries, T. P. A Fuzzy-Genetic Approach to Network Intrusion detection. 2008 GECCO Conference Companion on Genetic and Evolutionay Computation. Atlanta, GA, USA: ACM 2008. 2141-2146 p.

Ghali, N. I. Feature Selection for Effective Anomaly-Based Intrusion Detection. IJCSNS International Journal of Computer Science and Network Security, v.9, n.3, p.285-289. 2009.

Graps, A. An Introduction to Wavelets. IEEE Computational Science & Engineering, v.2, n.2, p.50-61. 1995.

Hamdi, M. e N. Boudriga. Detecting Denial-of-Service Attacks Using the Wavelet Transform. Computer Communications v.30, n.16, p.10. 2007.

Hecht-Nielsen, R. Theory of the Backpropagation Neural Network. Neural Networks, v.1, p.593-605. 1988.

Held, G. Wireless Mesh Networks. Boca Raton: Auerbach Publications. 2005

Holmes, G., A. Donkin*, et al.* WEKA: a Machine Learning Workbench. Second Australian and New Zealand Conference on Intelligent Information Systems. Bisbane, 1994. 357-361 p.

Hopfield, J. J. Neural Networks and Physical Systems with Emergent Collective Computational Abilities. Proceedings of the National Academy of Sciences, v.79, n.8, p.2554. 1982.

Hu, Y. H., S. Palreddy*, et al.* A Patient-Adaptable ECG Beat Classifier Using a Mixture of Experts Approach. IEEE Transactions on Biomedical Engineering, v.44, n.9, p.891-900. 1997.

Huang, C. T., S. Thareja*, et al.* Wavelet-based Real Time Detection of Network Traffic Anomalies. Workshop on Enterprise Network Security and the 2nd International Conference on Security and Privacy in Communication Networks. Baltimore, MD, USA, 2006. 1-7 p.

Hussain, S. A., K. Mahmood*, et al.* Factor Affecting Performance of AODV. Information Technology Jornal**:** 237-241 p. 2007.

Johnson, D., Y. Hu*, et al.* The Dynamic Source Routing Protocol (DSR) for Mobile Ad Hoc Networks for IPv4: RFC 4728. 2007

Johnson, D. B., D. A. Maltz*, et al.* DSR: The Dynamic Source Routing Protocol for Multi-Hop Wireless Ad Hoc Networks: Addison-Wesley. 2001

Karayiannis, N. B. e A. N. Venetsanopoulos. Artificial Neural Networks: Learning Algorithms, Performance Evaluation and Applications. New York: Springer-Verlag. 2004

Karygiannis, A., E. Antonakakis*, et al.* Detecting Critical Nodes for MANET Intrusion Detection Systems. 2nd International Workshop on Security, Privacy and Trust in Pervasive and Ubiquitous Computing, v.29, p.9-15. 2006.

Kauffmann, B., F. Baccelli*, et al.* Self Organization of Interfering 802.11 Wireless Access Network. INSTITUT NATIONAL DE RECHERCHE EN INFORMATIQUE ET EN AUTOMATIQUE. Paris. 2005

Kayacik, H. G., A. N. Zincir-Heywood*, et al.* Selecting Features for Intrusion Detection: A Feature Relevance Analysis on KDD 99 Intrusion Detection Datasets. Third Annual Conference on Privacy, Security and Trust (PST-2005). Canada. Octuber 12-14, 2005. p.

Khor, K.-C., C.-Y. Ting*, et al.* A Feature Selection Approach for Network Intrusion Detection. 2009 International Conference on Information Management and Engineering. Kuala Lumpur, Malaysia. 3 to 5 April 2009, 2009. 133-137 p.

Kilinkaridis, T. Routing Protocols for Wireless Ad Hoc Networks Hybrid routing protocols. 2007

Kim, S. S. e A. L. N. Reddy. Statistical Techniques for Detecting Traffic Anomalies Through Packet Header Data. IEEE/ACM Transactions on Networking (TON), v.16, n.3, p.562-575. 2008.

Kim, S. S., A. L. N. Reddy*, et al.* Detecting Traffic Anomalies through Aggregate Analysis of Packet Header Data. LECTURE NOTES IN COMPUTER SCIENCE, p.1047-1059. 2004.

Kohonen, T. e P. Somervuo. Self-Organizing Maps of Symbol Strings. Neurocomputing, v.21, n.1-3, p.19-30. 1998.

Kurose, J. F. e K. W. Ross. Redes de Computadores e a Internet. São Paulo: Pearson Education Companion. 2006

Labib, K. Computer Security and Intrusion Detection. ACM Crossroads, v.11, n.1, p.2-2. 2004.

Lazarevic, A., L. Ertoz*, et al.* A Comparative Study of Anomaly Detection Schemes in Network Intrusion Detection. Third SIAM International Conference on Data Mining. San Francisco, 2003. 25-36 p.

Lee, W. e S. J. Stolfo. A Framework for Constructing Features and Models for Intrusion Detection Systems. ACM Transactions on Information and System Security (TISSEC), v.3, n.4, p.227-261. 2000.

Li, Y. e L. Guo. TCM-KNN Scheme for Network Anomaly Detection Using Feature-based Optimizations. ACM Symposium on Applied Computing. Fortaleza, Ceará, Brazil: ACM 2008. 2103-2109 p.

Lippmann, R., J. W. Haines*, et al.* The 1999 DARPA off-line intrusion detection evaluation. Computer Networks, v.34, n.4, p.579-595. 2000.

Liu, G., Z. Yi*, et al.* A hierarchical intrusion detection model based on the PCA neural networks. Neurocomputing, v.70, n.7-9, p.1561-1568. 2007.

Lu, W. e A. A. Ghorbani. Network Anomaly Detection Based on Wavelet Analysis. EURASIP Journal on Advances in Signal Processing, v.2009, p.1-16. 2009.

Luo, H. e N. K. Shankaranarayanan. A distributed dynamic channel allocation technique for throughput improvement in a dense WLAN environment. 2004 IEEE International Conference on Acoustics, Speech and Signal Processings. Middletown:

IEEE. 5**:** Acoustics, Speech, and Signal Processing, 2004. Proceedings. (ICASSP apos;04). IEEE International Conference p. 2004.

Ma, C.-X. e Z.-M. Fang. A Novel Intrusion Detection Architecture Based on Adaptive Selection Event Triggering for Mobile Ad-hoc Networks. Second International Symposium on Intelligent Information Technology and Security Informatics, 2009. 198-201 p.

Mafra, P. M., J. Da Silva Fraga*, et al.* POLVO-IIDS: Um Sistema de Detecçao de Intrusao Inteligente Baseado em Anomalias. SBSeg2008. Gramado, 2008. 201-214 p.

Magnaghi, A., T. Hamada*, et al.* A Wavelet-based Framework for Proactive Detection of Network Misconfigurations: ACM New York, NY, USA, 2004. 253-258 p.

Mao, C. H., H. M. Lee*, et al.* Semi-supervised Co-training and Active Learning Based Approach for Multi-view Intrusion Detection: ACM New York, NY, USA, 2009. 2042-2048 p.

Marin, J., D. Ragsdale*, et al.* A Hybrid Approach to the Profile Creation and Intrusion Detection. DARPA Information Survivability Conference and Exposition II. Los Alamitos, 2001. 69–76 p.

Mcculloch, W. S. e W. Pitts. A Logical Calculus of the Ideas Immanent in Nervous Activity. Bulletin of Mathematical Biology, v.5, n.4, p.115-133. 1943.

Michailidis, E., S. K. Katsikas*, et al.* Intrusion Detection Using Evolutionary Neural Networks. Informatics, 2008. PCI '08. Panhellenic Conference on, 2008. 8-12 p.

Mohammad, I. The Handbook of Ad Hoc Wireless Networks. Boca Raton: CRC Press. 2003

Monteiro, L. P. Algoritmo Plug-in LVQ para Servidor SQL. (Dissertação). Engenharia Informática e de Computadores (MEIC), Universidade Técnica de Lisboa, Lisboa, 2008. 93 p.

Mukherjee, A., S. Bandyopadhyay*, et al.* Location Management and Routing in Mobile Wireless Networks. Boston: Artech House. 2003

Mukkamala, S., G. Janoski*, et al.* Intrusion Detection Using Neural Networks and Support Vector Machines. Hybrid Informational Systems Advances in Soft Computing, 2002. 121-138 p.

Newsome, J., E. Shi*, et al.* The Sybil Attack in Sensor Networks: Analysis & Defenses. In the Third International Symposium on Information Processing in Sensor Networks (IPSN)**:** 259- 268 p. 2004.

Nie, N. e C. Comaniciu. Adaptive Channel Allocation Spectrum Etiquette for Cognitive Radio Networks. Mobile Networks and Applications - Springer Netherlands 2006.

Nong, Y., L. Xiangyang*, et al.* Probabilistic Techniques for Intrusion Detection Based on Computer Audit Data. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans, v.31, n.4, p.266-274. 2001.

Ogier, R., F. Templin*, et al.* Topology Dissemination Based on Reverse-Path Forwarding (TBRPF): RFC 3684. 2004

Oliveira, H. M. D. Análise de Sinais para Engenheiros - Uma abordagem via Wavelets. Rio de Janeiro: Brasport. 2007

Oliveira, S. D., H. C. Wong*, et al.* Rotas Alternativas para Detecção e Aumento da Resiliência à Intrusão Distribuída em RSSF. SBRC'06. 2006.

Orfila, A., J. M. Estevez-Tapiador*, et al.* Evolving High-Speed, Easy-to-Understand Network Intrusion Detection Rules with Genetic Programming. Lecture Notes in Computer Science, v.5484/2000, p.93-98. 2009.

Panjwani, S., S. Tan*, et al.* An Experimental Evaluation to Determine if Port Scans are Precursors to an Attack. International Conference on Dependable Systems and Networks (DSN-2005), Yokohama, Japan, June: IEEE Computer Society, 2005. 602-611 p.

Papadimitriou, G. I., A. S. Pomportsis*, et al.* Wireless Network. London: JOHN WILEY & SONS, LTD. 2003

Patcha, A. e J.-M. Park. An Overview of Anomaly Detection Techniques: Existing Solutions and Latest Technological Trends. Computer Networks, v.51, n.12, p.3448-3470. 2007.

Perkins, C., E. Belding-Royer*, et al.* Ad hoc On-Demand Distance Vector (AODV) Routing: FRC 3561. 2003

Prasant, M. e K. V. Srikanth. Ad Hoc Networks Technologies and Protocols. Boston: Springer Science. 2005

Provos, N., M. A. Rajab*, et al.* Cybercrime 2.0: When the Cloud Turns Dark. Communications of the ACM, v.52, n.4, Abril 2009, p.42-47. 2009.

Ramin, H. Ad-Hoc Networks: Fundamental Properties and Network Topologies. Dordrecht: Springer. 2006

Saade, D. C. M., C. V. N. Albuquerque*, et al.* Redes em Malha: Solução de Baixo Custo para Popularização do Acesso à Internet no Brasil. XXV SIMPÓSIO BRASILEIRO DE TELECOMUNICAÇÕES - SBrT 2007 2007.

Sanchez, J. S., F. Pla*, et al.* Learning Vector Quantization With Alternative Distance Criteria. 10th International Conference on Image Analysis and Processing, 1999. 84-90 p.

Schonlau, M., W. Dumouchel*, et al.* Computer intrusion: Detecting masquerades. Statistical Science, p.58-74. 2001.

Singh, G., F. Masseglia*, et al.* Data Mining for Intrusion Detection: From Outliers to True Intrusions. The 13th Pacific-Asia Conference on Knowledge Discovery and Data Mining. Bangkok, Tailande: Springer, 2009. 891-898 p.

Soares, L. F. G., G. Lemos*, et al.* Redes de Computadores - Das LANs MANs e WANs à Redes ATM. Rio de Janeiro: Campus. 1995

Song, G., J. Zhang*, et al.* The Research of Dynamic Change Learning Rate Strategy in BP Neural Network and Application in Network Intrusion Detection, 2008. 513-513 p.

Spiegel, M. R., J. J. Schiller*, et al.* Schaum's Outline of Probability and Statistics: McGraw-Hill. 2000. 408 p.

Suebsing, A. e N. Hiransakolwong. Feature Selection Using Euclidean Distance and Cosine Similarity for Intrusion Detection Model. 2009 First Asian Conference on Intelligent Information and Database Systems. Dong Hoi City, Vietnam: CPS. 1-3 April 2009, 2009. 86-91 p.

Tanenbaum, A. S. Redes de Computadores. Rio de Janeiro: Campus. 1997

Tang, W., Y. Cao*, et al.* Study on Adaptive Intrusion Detection Engine Based on Gene Expression Programming Rules: IEEE Computer Society Washington, DC, USA, 2008. 959-963 p.

Teodoro, P. G., J. E. D. Verdejo*, et al.* Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers & Security, v.28, n.1-2, p.18-28. 2009.

The Matworks. Matlab 7 Getting Started Guide. Natick, MA: Matworks. 2008

Tolba, A. S., M. A. Elsoud*, et al.* LVQ for Hand Gesture Recognition Based on DCT and Projectiion Features. Journal of Electrical Engineering, v.60, n.4, p.204-208. 2009.

Tsai, C.-F., Y.-F. Hsu*, et al.* Intrusion Detection by Machine Learning: A Review. Expert Systems With Applications, v.36, n.10, p.11994-12000. 2009.

Vokorokos, L., A. Kleinová*, et al.* Network Security on the Intrusion Detection System Level. International Conference on`Intelligent Engineering Systems, 2006. 270-275 p.

Waharte, S., R. Boutaba*, et al.* Routing protocols in wireless mesh networks: challenges and design considerations. Multimedia Tools and Applications**:** 285 - 303 p. 2006.

Walke, B. H., S. Mangold*, et al.* IEEE 802 Wireless Systems - Protocols, Multi-hop Mesh/Relaying,Performance and Spectrum Coexistence. London: John Wiley & Sons Ltd. 2006

Wang, S. Y., C. L. Chou*, et al.* The Design and Implementations of the NCTUns 1.0 Network Simulator. Computer Networks. 2**:** 175-197 p. 2003.

Widrow, B. e M. A. Lehr. 30 Years of Adaptive Neural Networks: Perceptron, Madaline and Backpropagation. Proceedings of the IEEE, v.78, n.9, p.1415-1442. 1990.

Wood, A. D. e J. A. Stankovic. Denial of Service in Sensor Networks. Computer. 35**:** 54-62 p. 2002.

Xiang, C. e S. M. Lim. Design of Multiple-level Hybrid Classifier for Intrusion Detection System. 2005 IEEE Workshop on Machine Learning for Signal Processing, 2005. 117-122 p.

Xiao, H., F. Hong*, et al.* Instrusion Detection in Ad-Hoc Networks. Journal of Communication and Computer**:** 42-47 p. 2006.

Xiao, Y., X. Shen*, et al.* Wireless-mobile Network Security. New York: Springer. 2007

Yang, D., G. Chen*, et al.* Learning Vector Quantization Neural Network Method for Network Intrusion Detection. Wuhan University Journal of Natural Sciences, v.12, n.1, p.147-150. 2007.

Yu, K.-M., M.-F. Wu*, et al.* Protocol-Based Classification for Intrusion Detection. WSEAS Transactions on Computer Research, v.3, n.3, p.135-141. 2008.

Yu, L., B. Chen*, et al.* An Integrated System of Intrusion Detection Based on Rough Set and Wavelet Neural Network. Third International Conference on Natural Computation: IEEE Computer Society, 2007. 194 - 199 p.

Livros Grátis

(http://www.livrosgratis.com.br)

Milhares de Livros para Download:

Baixar livros de Administração
Baixar livros de Agronomia
Baixar livros de Arquitetura
Baixar livros de Artes
Baixar livros de Astronomia
Baixar livros de Biologia Geral
Baixar livros de Ciência da Computação
Baixar livros de Ciência da Informação
Baixar livros de Ciência Política
Baixar livros de Ciências da Saúde
Baixar livros de Comunicação
Baixar livros do Conselho Nacional de Educação - CNE
Baixar livros de Defesa civil
Baixar livros de Direito
Baixar livros de Direitos humanos
Baixar livros de Economia
Baixar livros de Economia Doméstica
Baixar livros de Educação
Baixar livros de Educação - Trânsito
Baixar livros de Educação Física
Baixar livros de Engenharia Aeroespacial
Baixar livros de Farmácia
Baixar livros de Filosofia
Baixar livros de Física
Baixar livros de Geociências
Baixar livros de Geografia
Baixar livros de História
Baixar livros de Línguas

Baixar livros de Literatura
Baixar livros de Literatura de Cordel
Baixar livros de Literatura Infantil
Baixar livros de Matemática
Baixar livros de Medicina
Baixar livros de Medicina Veterinária
Baixar livros de Meio Ambiente
Baixar livros de Meteorologia
Baixar Monografias e TCC
Baixar livros Multidisciplinar
Baixar livros de Música
Baixar livros de Psicologia
Baixar livros de Química
Baixar livros de Saúde Coletiva
Baixar livros de Serviço Social
Baixar livros de Sociologia
Baixar livros de Teologia
Baixar livros de Trabalho
Baixar livros de Turismo