Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Caracterização de Comportamento Anômalo em Redes Ad Hoc

Nicole Rodrigues Nagel
Ruzbeh Shokranian

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Orientador
Prof. Dr. Jacir Luiz Bordim

Brasília
2008

Universidade de Brasília – UnB
Instituto de Ciências Exatas
Departamento de Ciência da Computação
Curso de Bacharelado em Ciência da Computação

Coordenador: Prof. Dr. Guilherme Albuquerque Pinto

Banca examinadora composta por:

Prof. Dr. Jacir Luiz Bordim (Orientador) – CIC/UnB
Prof.ª Dr.ª Carla Denise Castanho – CIC/UnB
Prof. Ms. João José Costa Gondim – CIC/UnB

CIP – Catalogação Internacional na Publicação

Nagel, Nicole Rodrigues.

Caracterização de Comportamento Anômalo em Redes Ad Hoc / Nicole Rodrigues Nagel, Ruzbeh Shokranian. Brasília : UnB, 2008.
83 p. : il. ; 29,5 cm.

Monografia (Graduação) – Universidade de Brasília, Brasília, 2008.

1. reputação, 2. confiança, 3. ad hoc, 4. segurança, 5. cooperação, 6. camada MAC, 7. cross-layer

CDU 004

Endereço: Universidade de Brasília
Campus Universitário Darcy Ribeiro – Asa Norte
CEP 70910–900
Brasília – DF – Brasil

Universidade de Brasília
Instituto de Ciências Exatas
Departamento de Ciência da Computação

Caracterização de Comportamento Anômalo em Redes Ad Hoc

Nicole Rodrigues Nagel
Ruzbeh Shokranian

Monografia apresentada como requisito parcial
para conclusão do Bacharelado em Ciência da Computação

Prof. Dr. Jacir Luiz Bordim (Orientador)
CIC/UnB

Profª. Drª. Carla Denise Castanho
CIC/UnB

Prof. Ms. João José Costa Gondim
CIC/UnB

Prof. Dr. Guilherme Albuquerque Pinto
Coordenador do Bacharelado em Ciência da Computação

Brasília, 25 de junho de 2008

Dedicatória

Gostaríamos de dedicar este trabalho primeiramente à família, principalmente aos pais, pelo apoio e incentivo contante, aos professores, que nos deram subsídios para desenvolver este trabalho. E por fim, a todos os nossos amigos que nos apoiaram em todos os momentos.

Agradecimentos

Agradecemos a Deus, aos pais, a nossa família, amigos e a todos que auxiliaram direta ou indiretamente para a conclusão deste trabalho. Ao Professor Dr. Jacir Bordim pela dedicação, paciência, amizade, apoio e por ter acreditado em nosso potencial. Gostaríamos de agradecer também à banca examinadora, professora Dra. Carla Castanho e ao professor Ms. João Gondim por terem dedicado seu tempo ao nosso trabalho e pelas sugestões que muito contribuíram.

Resumo

As redes ad hoc oferecem vários desafios, como falta de infra-estrutura, recursos limitados, problemas de segurança, e confiança entre os nós. Como é preciso cooperação entre os nós para o bom funcionamento da rede, pode ocorrer que alguns tenham um comportamento egoísta e busquem formas de obter vantagens. Este tipo de comportamento é prejudicial e precisa ser evitado.

Existem vários sistemas de reputação que buscam detectar mau comportamento e estabelecer uma relação de confiança entre os nós. Nestes sistemas, nós com mau comportamento ao serem detectados são classificados, como não confiáveis e punidos. Entretanto, estes mecanismos consideram os eventos que ocorrem apenas na camada de rede. Alguns sistemas de detecção analisam o desvio de conduta na camada de enlace, mas não são sistemas de reputação. Por serem restritos a apenas uma camada, cada um destes sistemas é vulnerável aos nós que eventualmente decidam agir maliciosamente em outra camada.

A proposta deste trabalho é estudar o mau comportamento de um nó levando em consideração que isto pode ocorrer em qualquer camada. Neste trabalho será proposto um sistema de reputação *cross-layer* que não está restrito a uma única camada. Entretanto, procurou-se dar maior ênfase aos problemas de mau comportamento na camada MAC, que equivale a um módulo do sistema de reputação proposto. Por meio de várias simulações, procurou-se mostrar o quanto um nó pode ganhar sendo malicioso na camada MAC e o prejuízo que este pode causar. Mostrou-se também como podemos detectar um nó malicioso e quais parâmetros devem ser utilizados para evitar falsos positivos.

Palavras-chave: reputação, confiança, ad hoc, segurança, cooperação, camada MAC, cross-layer

Abstract

Ad hoc networks offers various challenges, for example, no infrastructure, limited resources, security problems, and trust between nodes. For being an environment where nodes need to cooperate for the network to function, it can happen that some nodes have a selfish behavior and seek ways to obtain advantages. This type of behavior is harmful and needs to be avoided.

There are many reputation systems that seek to detect misbehavior and establish a trust relation between nodes. In these systems when misbehaving nodes are detected they are classified as non trustful and punished. However, these systems only consider events that happens in the network layer. There are detection systems that analyze misconduct in the link layer, but they are not reputation systems. Since they are restrict to only one layer they are vulnerable to nodes that decide to act in a layer that they do not monitor.

Our proposal in this work is to study nodes misbehavior considering that this can happen in any layer. An ideal reputation system should not be restricted to a layer, but should collect information from all layers. This is only possible if there is interaction between layers, also called, cross-layer, which will be discussed in this work. Since MAC layer misbehavior is little explored, we focus our work in this problem. While in the network layer there are many related research. We seek to show how much a node can gain being malicious and how the network is affected. We also show how we can detect a malicious node and what parameters we need to avoid false positives.

Keywords: reputation, trust, ad hoc, security, cooperation, MAC layer, cross-layer

Sumário

Lista de Figuras **10**

Lista de Tabelas **12**

Lista de Acrônimos **15**

Capítulo 1 Introdução **16**
1.1 Justificativa 18
1.2 Objetivos 18
1.3 Estrutura da Monografia 18

Capítulo 2 Redes Ad Hoc **20**
2.1 Definições 20
2.2 Subdivisão em Camadas e sua Aplicação em Redes Ad Hoc 23
2.2.1 TCP/IP 24
2.2.2 *Cross-layer* 26
2.3 Padrão IEEE 802.11 28
2.3.1 DCF 30
2.3.2 PCF 33
2.3.3 802.11e 34
2.4 Roteamento 35
2.4.1 AODV 36
2.4.2 DSR 37
2.5 Segurança em Redes Ad Hoc 37
2.5.1 Premissas de Segurança 37
2.5.2 Problemas 38
2.5.3 Soluções Propostas na Literatura 39

Capítulo 3 Detecção de Desvio de Conduta **41**
3.1 Camada de Transporte 41
3.2 Camada de Rede 41
3.3 Camada MAC 42
3.4 Sistemas de Detecção na Camada MAC 43
3.4.1 DOMINO 43
3.4.2 Detecção de Mau Comportamento na Camada MAC (Kyasanur *et. al*) 45
3.5 Sistemas de Detecção *Cross-layer* 46

3.5.1 *Cross-layer Framework* 46

Capítulo 4 Sistemas de Reputação **49**
4.1 Definições 49
4.1.1 Reputação e Confiança 49
4.1.2 Transitividade de Confiança 49
4.1.3 Reputação Direta e Indireta 50
4.1.4 Reputação Global e Local 51
4.1.5 Segurança e Confiança 52
4.2 Aplicações 52
4.2.1 Sites de leilão eletrônico 52
4.2.2 Redes *Peer-to-Peer* (P2P) 53
4.2.3 Redes Ad Hoc 54
4.3 Desafios em Sistemas de Reputação 54
4.4 Trabalhos Relacionados 55
4.4.1 Incentivos Econômicos 55
4.4.2 Protocolos de Reputação e Confiança 56

Capítulo 5 Modelo Proposto **60**
5.1 Descrição dos Objetivos 60
5.2 Proposta: Sistema de Reputação Cross-layer 61
5.3 Definições Matemáticas e Estatísticas 63
5.4 Descrição do Modelo 65
5.4.1 Simulação de Uso do Canal 65
5.4.2 Simulação do Nível de Mal Comportamento 66

Capítulo 6 Resultados Experimentais e Análise **68**
6.1 Simulação de Uso do Canal 68
6.2 Simulação do Nível de Mal Comportamento 73

Capítulo 7 Conclusão e Trabalhos Futuros **78**

Referências **79**

Lista de Figuras

2.1 Exemplo de Redes Sem fio com infra-estrutura 21
2.2 Exemplo de Redes Ad Hoc 21
2.3 Pilha de protocolos TCP/IP 24
2.4 Correspondência entre as camadas dos modelos OSI e TCP/IP . . 25
2.5 Desafios técnicos encontrados em cada camada nas redes Ad Hoc (fonte: [4]) 26
2.6 Formas de Violação Cross-layer (fonte:[33]) 27
2.7 Formas de interação entre camadas nas propostas de cross-layer existentes (fonte:[33]) 28
2.8 (a) Problema do Terminal Escondido. (b) Problema da Estação Exposta. 30
2.9 Camada de Enlace 31
2.10 Diagrama explicativo do funcionamento dos pacotes CTS e RTS. Fonte: [29]. 32
2.11 Diagrama Explicativo do funcionamento do Backoff. Fonte: [29] . 33
2.12 Exemplo de Backoff Exponecial 33
2.13 Espaço de intervalo entre os frames 34
2.14 Coexistência entre os modos PCF e DCF 34
2.15 Filas de prioridades do padrão 802.11e 35
2.16 Processo de descoberta de rotas AODV. Setas azuis são pacotes RREQ, amarelas RREP. O nó vermelho é o remetente e o verde o destinatário. 37

3.1 Exemplo de uma topologia que ilustra o problema de observar o número de retransmissões de cada nó 45
3.2 Arquitetura do Framework *Cross-layer*. Fonte: [15] 47

4.1 Confiança transitiva derivada 50
4.2 Confiança transitiva incorreta 50
4.3 Combinação de confiança transitiva 51
4.4 Representação do funcionamento das redes P2P 53
4.5 Diagrama que representa os componentes básicos presentes na maioria dos sistemas de reputação e confiança 54

5.1 Componentes do Sistema de Reputação entre camadas. 62
5.2 Grafo completo K_6 64

6.1 Porcentagem de uso do canal com dois nós 69

6.2 Porcentagem de uso do canal com quatro nós 69
6.3 Porcentagem de uso do canal com oito nós 70
6.4 Porcentagem de uso do canal com dezesseis nós 70
6.5 Porcentagem de uso do canal com trinta e dois nós 71
6.6 Distribuição binomial da simulação de uso do canal 72
6.7 Nível de mal comportamento com dois nós 74
6.8 Nível de mal comportamento com quatro nós 74
6.9 Nível de mal comportamento com oito nós 75
6.10 Nível de mal comportamento com dezesseis nós 75
6.11 Nível de mal comportamento com trinta e dois nós 76

Lista de Tabelas

4.1 *Comparação entre os Sistemas de Reputação* 59

Lista de Acrônimos

AC_BE Access Control Best Effort

AC_BK Access Control Background

AC_VI Access Control Video

AC_VO Access Control Voice

ACK ACKnowledges

AIFS Access Category Inter Frame Spacing

AODV Ad Hoc On-Demand Distance Vector

CCA Clear Channel Assessment

CONFIDANT Cooperation Of Nodes: Fainess In Dynamic Ad hoc Network

CORE Collaborative Reputation Mechanism

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance

CSMA/CD Carrier Sense Multiple Access with Collision Detection

CTS Clear To Send

CW Contention Window

DCF Distributed Coordination Function

DIFS DCF Inter Frame Spacing

DOMINO System for Detection Of greedy behavior in the Mac layer of IEEE 802.11 public NetwOrks

DSR Dynamic Source Routing

DSSS Direct Sequence Spread Spectrum

EDCAF Enhanced Distributed Channel Access Function

FHSS Frequency Hopping Spread Spectrum

FTP File Transfer Protocol

HR-DSSS High Rate Direct Sequence Spread Spectrum

ICMP Internet Control Message Protocol

IEEE Institute of Electrical and Electronics Engineers

IGMP Internet Group Management Protocol

IP Internet Protocol

ISO International Standards Organization

LLC Logical Link Control

MAC Medium Access Control

MANET Mobile Ad hoc Networks

MDS - p Primary Misbehavior Detection System

MDS - s Secondary Misbehavior Detection System

MIB Management Information Base

NAV Network Allocation Vector

OFDM Orthogonal Frequency Division Multiplexing

OSI Open Systems Interconnection

P2P Peer-to-Peer

PCF Point Coordination Function

PC Point Coordinator

PLCP Physical Layer Convergence Protocol

PMD Physical Medium Dependent

PRB Pretictable Random Backoff

RREP Route Reply

RREQ Route Request

RRER Route Error

RTS Request To Send

SAP Service Access Point

SIFS Shorter Inter Frame Spacing

SMTP Simple Mail Transfer Protocol

SNMP Simple Network Management Protocol

SNR Signal-to-Noise Ratio

SNR Singal-to-Noise Ratio

SORI Secure and Objective Reputation-based Incentive Scheme

TCP Transmission Control Protocol

TELNET TELecommunication NETwork

TTL Time To Live

UDP User Datagram Protocol

Capítulo 1

Introdução

As redes sem fio cresceram e se desenvolveram intensamente nas últimas décadas e atualmente podem ser encontradas em escritórios, hospitais e nas casas de milhares de pessoas. Embora estejam presentes no cotidiano de muitas pessoas, a maioria das redes wireless depende de uma infra-estrutura centralizada e os custos associados ainda são empecilho para sua ampla adoção. As redes ad hoc, por sua vez, são constituídas por dispositivos móveis autônomos capazes de configurarem, entre si, uma rede. Assim, a criação de uma rede torna-se fácil e possui baixo custo. Esta facilidade, entretanto, exige soluções tecnológicas mais complexas.

Historicamente, as redes ad hoc foram primeiro utilizadas em aplicações militares. A presença de uma infra-estrutura fixa e centralizada para estabelecer a comunicação entre os soldados no campo de batalha não é viável devido ao dinamismo e à natureza das operações militares. Desde 1990, a importância das redes ad hoc estendeu-se para a área comercial e residencial, devido ao uso crescente de dispositivos móveis como laptops, PDAs e celulares, além do surgimento de padrões como Bluetooh, IEEE 802.11 e Hiperlan [14]. A facilidade de configuração e a falta de infra-estrutura são atrativos para o setor comercial e já existem inúmeras aplicações como redes de sensores, operações de resgate e emergência, aplicações educacionais, entretenimento, computação colaborativa e distribuída, redes mesh, redes híbridas entre outras [29].

Existem vários tipos de redes sem fio e estas podem ser categorizadas sob vários critérios, um deles é a formação e arquitetura da rede. Segundo este critério, existem duas categorias de redes wireless: redes baseadas em infra-estrutura e redes sem infra-estrutura ou ad hoc [4]. O primeiro tipo, mais conhecido, depende de um ponto de acesso ou uma torre central fixa que fornece uma área de cobertura para os aparelhos que estiverem ao alcance do sinal. O segundo tipo também chamado de redes ad hoc ou MANET (*Mobile Ad hoc Networks*) são redes formadas por uma coleção de nós sem uma infra-estrutura fixa ou centralizada, onde os nós são independentes e cada nó possui algumas funções básicas de rede como roteamento e encaminhamento de pacotes. Portanto, cada nó age como um host e um roteador ao mesmo tempo.

As redes ad hoc herdam os problemas tradicionais das redes wireless, como interferência, baixa confiabilidade, pouca largura de banda, alta influência do meio para o correto funcionamento da rede, recursos limitados como bateria e poder de processamento, cobertura de serviço limitada [4], e acrescentam ainda

novos desafios. Como o alcance de rádio dos nós é limitado, a comunicação entre vários saltos só é possível se houver cooperação entre eles. Entretanto, para que haja cooperação os nós precisam encaminhar e rotear o pacote de outros nós, o que incorre em maior consumo de bateria e poder de processamento. E uma vez que os nós possuem recursos de bateria e poder de processamento limitados, a cooperação pode ser bastante onerosa e alguns nós podem ter um comportamento egoísta. Os nós que se desviam do comportamento regular definido pelos protocolos em qualquer camada da pilha TCP/IP são chamados mal comportados. Além dos nós egoístas, o mau comportamento do nó pode ser intencional, para realizar um ataque na rede, conhecido como nó malicioso ou provocado por alguma falha do sistema. É importante que os nós estejam preparados para identificar o mau comportamento de um outro nó, pois sua presença degradará o desempenho da rede [27].

O dinamismo da rede e a falta de infra-estrutura dificultam o uso de mecanismos centralizados para controle de acesso, autenticação, ou mesmo controle de tráfego. [4] Qualquer uma dessas políticas deve ser realizada de forma distribuída entre os nós da rede. Existem algumas propostas de roteamento seguro que utilizam criptografia e fornecem meios para prevenir alguns ataques específicos que variam a cada proposta. Esses protocolos de roteamento seguro, entretanto, são limitados e contrariam a ausência de infra-estrutura e a escassez de recursos das redes ad hoc. A hipótese adotada por boa parte dos protocolos seguros é a existência de uma ligação segura entre todos os nós da rede. Esta hipótese, introduz a necessidade de um ambiente gerenciado por uma autoridade comum, que forneça uma chave compartilhada a cada par de nós ou uma entidade certificadora que emita certificados para todos os nós da rede. Entretanto, uma autoridade certificadora deve estar disponível durante toda a vida da rede para prover a revogação de certificados. Em ambientes ad hoc, isto não é viável [4]. Além disso, o uso de criptografia não é suficiente para garantir que um nó seja confiável.

O modelo de incentivos econômicos é uma tentativa de forçar a cooperação entre os nós por meio de uma espécie de crédito ou micro pagamento para compensar o serviço do nó, ou seja, o nó recebe um pagamento virtual para encaminhar as mensagens dos outros nós [9, 38]. O problema desse sistema de moedas virtuais é a sua dependência de um hardware imutável ou a necessidade de um servidor central para determinar o débito e o crédito de cada nó envolvido na transmissão da mensagem. Estas abordagens não são ideais em ambientes heterogêneos como o ad hoc.

Um outro mecanismo que incentiva a cooperação são os sistemas de reputação e confiança, semelhantes aos sistemas de sites de leilão como MercadoLivre e Ebay [22]. A cada transação comercial cada parte avalia positivamente ou negativamente a outra parte, que se refletirá em pontos para a reputação do usuário. Os mecanismos de reputação e confiança podem ser aplicados às redes ad hoc como uma forma de evitar o mau comportamento e incentivar a cooperação dos nós por meio do monitoramento dos nós vizinhos [5, 18, 28, 3].

Os sistemas de reputação precisam ser robustos o suficiente para identificarem e impedirem que os nós mal comportados participem ativamente e degradem o desempenho da rede. Esses sistemas devem tratar a perda e o ganho de pontos de

reputação de forma diferenciada, pois devem estar preparados para tratar nós que agem por um longo período de forma ativa e regular e ao conseguirem um bom valor de reputação comecem a agir maliciosamente. Estes nós, também chamados de bizantinos podem causar sérios danos até que o seu valor de reputação degrade significativamente e sejam excluídos da rede. Entretanto, existe a possibilidade de que alguns nós sejam valorados de forma incorreta, porque é difícil diferenciar nós egoístas e maliciosos dos nós defeituosos que desviam do comportamento regular devido a uma falha do sistema ou interferência do sinal. É importante que exista uma mecanismo de redenção para que os nós que agiram incorretamente, sejam perdoados e recebam uma segunda, ou terceira chance [8].

1.1 Justificativa

Os protocolos existentes observam o comportamento dos nós apenas em uma camada. Entretanto, um nó mal comportado pode agir em várias camadas. Por exemplo, um protocolo que observa o comportamento dos nós na camada de rede não seria capaz de identificar um nó mal comportado na camada de enlace. Existem vários problemas de mau comportamento na camada de enlace que são ainda pouco explorados. Por exemplo, na camada de enlace, os nós devem esperar um período de tempo pré-determinado para transmitir, também chamado de *backoff*, esse período é gerado de forma aleatória e independente por cada nó para evitar que haja colisões durante a transmissão. Assim, é possível que um nó mal comportado não respeite os limites de geração de tempo aleatório e sempre consiga ter acesso ao meio para transmitir.

1.2 Objetivos

Este trabalho tem como objetivo fazer um estudo e análise dos sistemas de reputação existentes e propor um novo sistema de reputação que identifique o mau comportamento em outras camadas, além da camada de rede. Além disso, o foco deste trabalho foi identificar as vantagens que um nó malicioso pode obter na camada de enlace e a partir de que momento os nós normais conseguem perceber a presença de um nó malicioso. Os resultados foram obtidos por meio de várias simulações no simulador desenvolvido especificamente para este trabalho.

1.3 Estrutura da Monografia

A monografia está dividida da seguinte forma: Introdução (capítulo 1), Redes Ad Hoc (capítulo 2), Detecção de Desvio de Conduta (capítulo 3), Sistemas de Reputação (Capítulo 4), Modelo Proposto (Capítulo 5), Resultados Experimentais e Análise (Capítulo 6) e Conclusão e Trabalhos Futuros (Capítulo 7). O capítulo 2 explica brevemente o funcionamento de uma rede ad hoc; o capítulo 3 mostra as formas de mau comportamento em várias camadas e como elas podem ser detectadas; no capítulo 4 discute-se sobre reputação e confiança e alguns sistemas de

reputação existentes; o capítulo 5 apresenta o modelo proposto; o capítulo 6 mostra os resultados e análise deste trabalho; e finalmente no capítulo 7 encerra-se este trabalho.

Capítulo 2

Redes Ad Hoc

Nesse capítulo iremos introduzir alguns conceitos de redes ad hoc e suas características. Em seguida, descreveremos a suíte de protocolos TCP/IP e o padrão 802.11. Por fim, discutiremos alguns problemas de segurança em redes ad hoc.

2.1 Definições

O termo redes sem fio refere-se a um tipo de redes onde a forma de comunicação entre os dispositivos é feita por meio de ondas eletromagnéticas como ondas de rádio ou infra-vermelho [4]. Podemos listar vários dispositivos sem fio, como por exemplo: PDAs, sensores sem fio, telefones celulares, laptops entre outros.

Existem vários tipos de redes sem fio e estas podem ser classificadas sobre diferentes aspectos. Nesse trabalho utilizaremos a classificação das redes sem fio quanto a formação e arquitetura da rede. Segundo este critério, as redes sem fio são classificadas em dois tipos distintos: redes dependente de infra-estrutura e redes sem infra-estrutura ou **ad hoc** [4]:

- **Com Infra-Estrutura**. Tipicamente uma rede sem fio desse tipo possui uma infra-estrutura pré-configurada e provê diversos serviços como, por exemplo, acesso a Internet, além da conectividade entre os outros dispositivos da rede. A figura 2.1 exemplifica um cenário de redes sem fio com infra-estrutura, nesse caso a conectividade entre os nós é intermediada por uma estação central fixa.

- **Sem Infra-Estrutura ou Ad Hoc**. As redes ad hoc são formadas de forma dinâmica por meio da cooperação entre nós heterogêneos e independentes. A figura 2.2 mostra a estrutura de um típico cenário ad hoc. Não existe nenhuma infra-estrutura previamente configurada, cada nó age de forma independente segundo as condições dos meios correntes. Os nós possuem dois papéis distintos: host e roteador. Como não existem roteadores dedicados e o alcance do rádio é limitado, a comunicação em vários saltos deve ser feita por meio da cooperação entre os nós, onde cada nó encaminha o pacote de outros nós. Os nós podem se mover livremente e é esperado que os nós se movam com frequência, o que justifica o dinamismo intenso da rede e as constantes alterações de rota entre os nós.

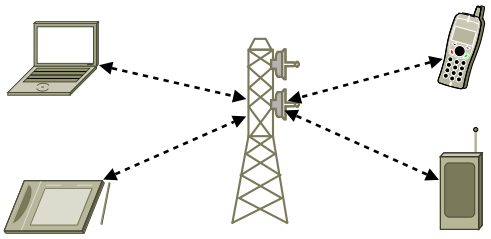

Figura 2.1: Exemplo de Redes Sem fio com infra-estrutura

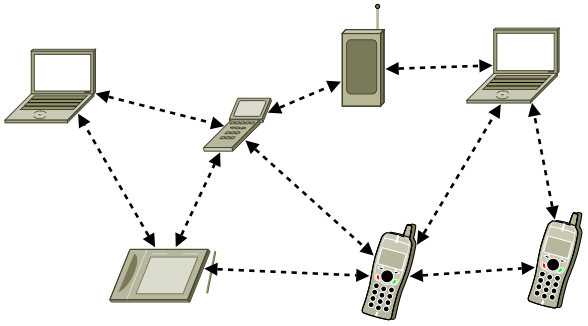

Figura 2.2: Exemplo de Redes Ad Hoc

As redes celulares são um tipo de rede sem fio, mas são classificadas como redes **com infra-estrutura** e não ad hoc, pois a comunicação é feita diretamente entre o celular e a estação base. Já a rede Bluetooth [34] é um tipo de rede ad hoc que permite a conexão de vários dispositivos distintos, desde telefones celulares a fones de ouvido, câmeras digitais e outros. A rede bluetooth utiliza o paradigma mestre e escravo, onde o mestre informa aos escravos que endereços usar, quando podem transmitir e por quanto tempo podem transmitir. Embora o nó mestre centralize a gerência da rede, não existe uma infra-estrutura fixa como uma torre de celular.

As características do meio de transmissão sem fio são bastantes distintas daquelas encontradas nas redes tradicionais (cabeadas):

- **Alta interferência do meio:** os sinais infra-vermelhos sofrem bastante com a interferência de raios solares e calor e ainda podem ser bloqueados ou absorvidos por alguns objetos e materiais. Os sinais de rádios, por sua vez, são menos suscetíveis a bloqueios, mas sinais elétricos emitidos por outros dispositivos como microondas podem interferir neste sinal. Os próprios dispositivos de rede interferem entre si devido a característica de comunicação em broadcast. Dessa forma a taxa de erro é relativamente superior àquela presente nas redes tradicionais. Enquanto a taxa de erro em fibras ópticas é da ordem de 10^{-9}, em redes sem fio a taxa de erro é da ordem de 10^{-4} [29].

- A **velocidade de comunicação** e a **capacidade do meio de transmissão** é inferior àquela observada nas redes com fio, logo a qualidade de serviço também será inferior.

- **Topologia Dinâmica e Conectividade Restrita:** a conectividade da rede pode ser parcial em alguns momentos devido a mobilidade dos nós, ou seja, o alcance do sinal entre os nós varia e, em alguns momentos pode não haver conectividade. Durante a transmissão também pode haver perda de pacotes devido a interferências do sinal.

- **Recursos Limitados:** pequeno poder de processamento, memória, capacidade de disco rígido e bateria limitados. A limitação do tamanho do dispositivo também implica em interface com usuário limitada e pequena área de visualização (ou display).

- **Recursos limitados do meio de transmissão:** o canal é compartilhado e o alcance do sinal não pode ser determinado com acurácia, porque depende de vários outros fatores como a potência do sinal, níveis de ruído presente no ambiente, ou seja, não é possível determinar com facilidade e precisão as fronteiras de alcance do meio de transmissão. Além disso, as frequências disponíveis são limitadas e sofrem com regulamentações restritivas que variam para cada país.

- **Vulnerabilidades:** como o sinal de rádio pode ser interceptado por qualquer um que esteja ao alcance do sinal, mecanismos de segurança são mais difíceis de serem implementados.

As redes ad hoc herdam boa parte das características citadas acima e acrescentam outras. Entretanto, as redes ad hoc eliminam as restrições de mobilidade impostas por uma infra-estrutura fixa. No ambiente ad hoc os nós podem se mover livremente e novas rotas e caminhos são estabelecidos com os nós da rede.

As redes ad hoc possuem inúmeros benefícios, como a facilidade de se estabelecer uma rede e configurá-la. Por exemplo, considere um grupo de pesquisadores que desejem compartilhar suas pesquisas e apresentações durante uma conferência, uma rede ad hoc pode ser facilmente criada para esse fim. Além da facilidade, os custos podem ser menores, visto que não existe a necessidade de dispositivos centralizados e dedicados à realização de uma função específica (roteadores, *firewall*). Por causa desses benefícios, essas redes tornaram-se um atrativo para a área comercial. Suas aplicações são inúmeras, englobam aplicações educacionais, entretenimento, serviços de emergência e muitas outras.

Como dito anteriormente, as redes ad hoc acrescentam novos desafios. A falta de infra-estrutura aliada à comunicação em vários saltos, acrescenta algumas dificuldades que não existiam nas redes sem fio dependentes de infra-estrutura. Além disso, a falta de infra-estrutura pressupõe que não existam dispositivos centrais, ou seja, a gerência da rede torna-se muito mais complexa. O controle de tráfego comumente realizado por firewalls ou agentes de gerência com repositório de dados como as MIBs (*Management Information Base*) [34], deve ser realizado de forma distribuída entre os nós da rede, e conseqüentemente torna-se muito mais complexo. O mesmo acontece com o controle de acesso, onde a autenticação dos dispositivos da rede, por exemplo, deve ser realizada de forma distribuída. Ainda, o próprio dinamismo da rede e os recursos limitados impedem a existência de nós centrais com papel de gerência e detecção de falhas.

A alta mobilidade dos nós e a comunicação em vários saltos resultam em mudanças freqüentes de rotas entre os nós e possivelmente perda de pacotes caso o nó destino esteja se movimentando durante a transmissão de um pacote. Os protocolos de roteamento, os mecanismo de detecção de intrusão ou ainda detecção de nós maliciosos devem estar preparados para esse dinamismo da rede. Como a comunicação entre os nós é feita por broadcast e o alcance de rádio desses nós é bastante limitado, alguns problemas surgem e devem ser tratados como o problema do terminal escondido e o problema do terminal exposto [4].

As redes ad hoc são bastante heterogêneas. Cada dispositivo pode, por exemplo, ser equipado com uma ou mais interfaces de rádio que podem operar em diferentes faixas de freqüência e conter capacidades de transmissão e recepção diferenciadas. Além disso, existe também heterogeneidade em relação ao software e hardware, onde cada nó pode conter configurações e capacidades de processamento distintos. Os protocolos de roteamento e os algoritmos podem ser bastante complexos devido a essa heterogenidade de recursos, e requerem mecanismos que estejam aptos a lidar com alterações do canal de transmissão, variações bruscas do tráfego da rede e balanceamento de carga [34].

Em decorrência da mobilidade dos nós, os dispositivos das redes ad hoc são portáteis e ficam limitados pela bateria, logo, as aplicações ficam limitadas por esse recurso. Este problema é agravado pelos papéis desempenhados pelos nós, de host e roteador, ou seja, uma quantidade de energia adicional torna-se necessária para encaminhar os pacotes para outros nós. É importante, também, que os nós fiquem sempre disponíveis (on-line) para que o roteamento dos pacotes possa ser realizado, aumentando ainda mais a cooperação do nó na rede como um todo.

Como descrito nas seções anteriores a conectividade entre os nó é estabelecida por meio do roteamento e do encaminhamento de pacotes. A falta de infra-estrutura acrescenta benefícios, mas alguns mecanismos devem ser reformulados. Um nó pode não encaminhar um pacote por vários motivos: sobrecarga, ligações entre os nós desfeitas, comportamento egoísta seja para tirar vantagem sobre outros nós, ou porque o sistema está em um estado crítico e outras funcionalidades foram cortadas para prover um pouco mais de energia para a função crítica do sistema. O mau comportamento dos nós e rotas inseguras podem causar sérios impactos no desempenho geral da rede. Novamente a falta de infra-estrutura dificulta e complica a detecção e o isolamento de forma eficiente e rápida dos nós mal comportados.

Muitas das aplicações de redes ad hoc envolvem redes em larga escala com milhares de nós, como por exemplo, redes de sensores e operações táticas militares. A escalabilidade é crítica nestes ambientes, assim os algoritmos e os protocolos devem estar preparados para essa possível expansão da rede.

2.2 Subdivisão em Camadas e sua Aplicação em Redes Ad Hoc

As redes de computadores são organizadas como uma pilha de camadas ou níveis, colocadas umas sobre as outras. As camadas são logicamente ordenadas de modo

que uma esteja em um nível hierárquico mais alto que a anterior. Cada camada possui uma responsabilidade diferente durante a comunicação e seu objetivo é oferecer determinados serviços às camadas superiores, isolando essas camadas dos detalhes de implementação desses recursos.

2.2.1 TCP/IP

A pilha de protocolos TCP/IP [34, 10] possibilita a comunicação entre computadores com arquiteturas distintas e sistemas operacionais completamente diferentes. Em meados de 1960, o Departamento de Defesa dos Estados Unidos financiou um projeto de comutação de pacotes, que em 1990 tornou-se a forma de comunicação mais utilizada entre redes de computadores. O TCP/IP é um sistema aberto em sua definição e muitas implementações estão publicamente disponíveis. O TCP/IP é dividido em quatro camadas (figura 2.3).

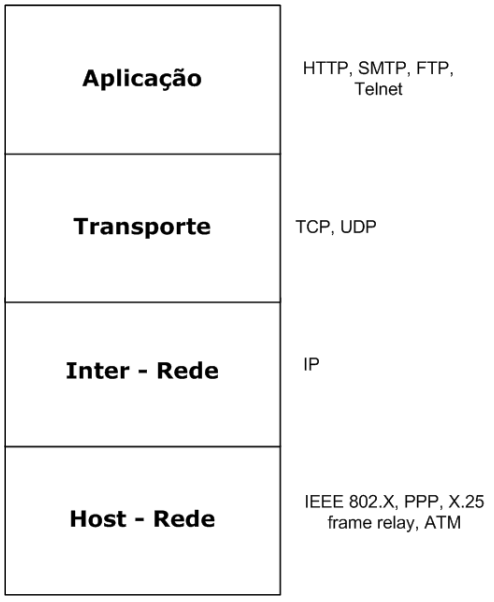

Figura 2.3: Pilha de protocolos TCP/IP

- A camada de **host-rede** é responsável pelos detalhes de hardware e a interface física com o meio de comunicação. Sua função básica é tratar da transmissão dos sinais elétricos por um canal de comunicação.

- A camada de **inter-rede** trata do movimento de pacotes. Sua função é garantir que o host envie pacotes em qualquer rede e permitir que os pacotes trafeguem de forma independente até o destino. O roteamento de pacotes é feito nessa camada. O IP (*Internet Protocol*), o ICMP (*Internet Control Message Protocol*) e o IGMP (*Internet Group Management Protocol*) provêem os serviços desta camada.

- A camada de **transporte** provê um fluxo de dados entre os hosts de origem e destino. Dois protocolos foram definidos: TCP (*Transmission Control*

Protocol) e UDP (*User Datagram Protocol*). O TCP é um protocolo orientado a conexões que provê um fluxo de dados confiável entre o nó destino e origem. Esse protocolo divide o fluxo de bytes de entrada em pedaços de tamanho apropriado e os repassa para a camada abaixo. Além disso, o protocolo TCP possui controle de fluxo para impedir a sobrecarga do nó destino com um volume de dados maior do que sua capacidade. Já o protocolo UDP é um protocolo sem conexão e não confiável que provê um serviço mais simples a camada de aplicação. O protocolo UDP também divide o fluxo de entrada em pedaços menores, chamados datagramas, e os envia de um ponto a outro, mas não existem garantias de que estes datagramas alcancem o outro lado. Qualquer confiabilidade deve ser provida pela camada de aplicação.

- A camada de **aplicação** é responsável por tratar dos detalhes da aplicação em particular. Ela contém os protocolos de nível mais alto como, por exemplo: o protocolo de terminal virtual (TELNET), protocolo de transferência de arquivos (FTP), protocolo para o gerenciamento de correio eletrônico (SMTP), protocolo para o gerenciamento de rede (SNMP) entre outros.

O modelo TCP/IP tornou-se referência depois do surgimento da suíte de protocolos TCP/IP. O modelo teórico desenvolvido pela ISO *(International Standards Organization)* foi criado com objetivo de padronizar os protocolos empregados nas diversas camadas e tornou-se um modelo de referência. Embora os protocolos tenham mudado bastante daquela época em relação aos protocolos atuais, a terminologia do modelo OSI [39, 34] persiste. A figura 2.4 mostra a correspondência entre as camadas dos dois modelos de referência.

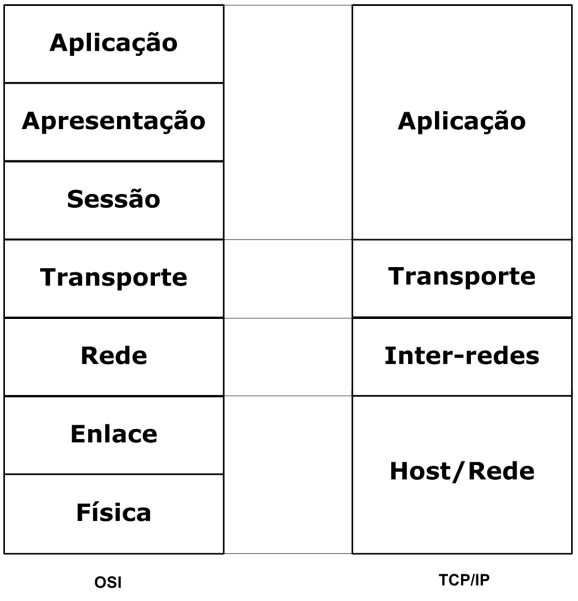

Figura 2.4: Correspondência entre as camadas dos modelos OSI e TCP/IP

Como descrito em seções anteriores as redes ad hoc acrescentam inúmeros desafios, e parte daquelas restrições podem, agora, ser relacionadas por camadas.

Os problemas em cada camada não serão abordados com detalhes, uma vez que o propósito desse trabalho é apenas identificar os dados que podem ser monitorados para distinguir o mau comportamento de um nó e definir a ação a ser tomada para que o nó seja excluído da rede. Observe a figura 2.5 para visualizar os principais problemas presentes em cada camada.

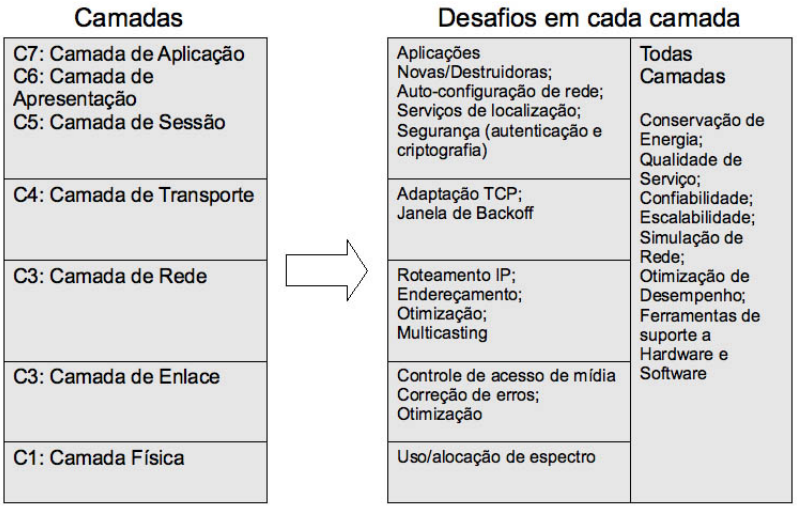

Figura 2.5: Desafios técnicos encontrados em cada camada nas redes Ad Hoc (fonte: [4])

2.2.2 *Cross-layer*

O protocolo TCP possui um bom desempenho nas redes *Ethernet*. E devido à baixa taxa de erros dessas redes, assume-se que todos os pacotes perdidos foram causados por congestionamento. Nas redes sem fio, todavia, o ambiente é mais instável. As taxa de erros são altas e a topologia é muito dinâmica, ou seja, como os nós estão em constante movimento a conectividade entre eles varia a todo instante. O TCP, entretanto, reage à perda de pacotes da mesma forma nos dois ambientes, ou seja, a janela de transmissão será reduzida antes da retransmissão dos pacotes e o controle de congestionamento será iniciado. Essas medidas irão causar uma redução desnecessária da utilização da largura de banda, reduzindo significativamente o desempenho da rede [2].

A estrutura rígida das camadas pode não ser flexível o bastante para ser utilizada em ambientes dinâmicos. A interação entre camadas (*cross-layer*), se usada de forma apropriada pode aumentar o desempenho da rede [25]. Como o meio físico varia constantemente, a troca de informações entre diferentes camadas pode ser utilizada para otimizar a vazão da rede.

Como vimos na seção anterior, na arquitetura em camadas, os protocolos devem prover serviços para as camadas acima e utilizar os serviços da camada abaixo, ou seja, a comunicação é restrita entre as camadas adjacentes e limita-se a chamada de procedimentos e respostas. Alternativamente, os protocolos poderiam violar a arquitetura em camadas permitindo a comunicação direta entre

camadas mais acima. Esta violação da arquitetura em camadas é chamada de *cross-layer*. A violação entre camadas pode ocorrer de várias formas:

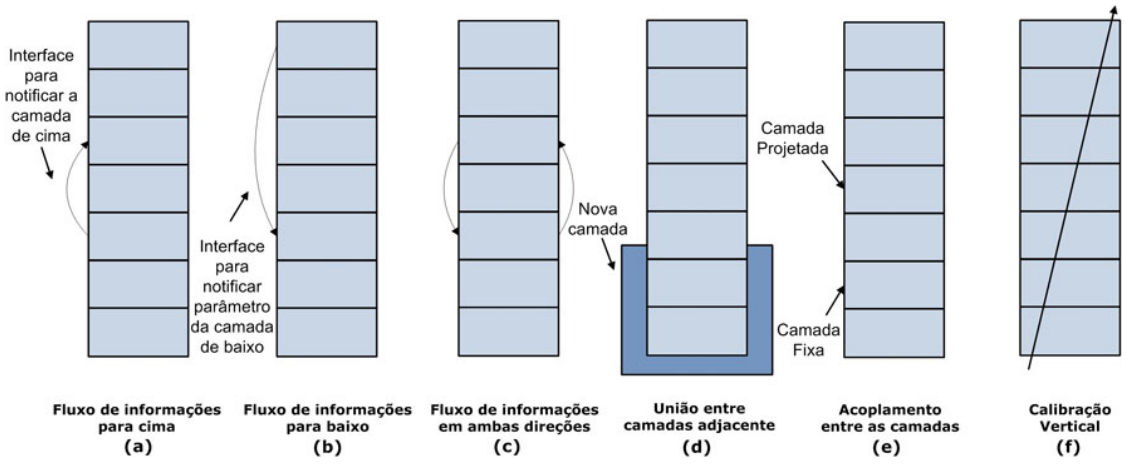

Figura 2.6: Formas de Violação Cross-layer (fonte:[33])

- **Criação de novas interfaces.** Alguns modelos de cross-layer requerem a criação de novas interfaces entre as camadas. A novas interfaces são utilizadas para compartilhar informações em tempo de execução. Esta categoria pode ser dividida de acordo com o fluxo de informações: das camadas de baixo para cima (Fig. 2.6(a)), das camadas de cima para baixo (Fig. 2.6(b)), e em ambas direções (Fig. 2.6(c)).

- **União entre camadas adjacentes.** Duas camadas adjacentes se unem formando uma nova camada que provê serviços das duas camadas (Fig. 2.6(d)). Não requer a criação de novas interfaces.

- **Acoplamento de camadas sem criar novas interfaces.** O número de camadas permanece o mesmo e não são criadas novas interfaces (Fig. (2.6(e)). Mas, o serviços prestados pelas camadas passam a depender de como é feita a implementação. Assim, a mudança de um serviço pode implicar mudanças na outra camada.

- **Calibração Vertical.** Ajuste de parâmetros entre as camadas (Fig. 2.6(f)). Por exemplo, o desempenho da camada de aplicação depende de parâmetros configurados na camadas mais abaixo.

Existem alguns trabalhos na literatura que sugerem um modelo de interação entre camadas. A maioria deles é evolucionária, ou seja, são compatíveis com as redes existentes. Em [1, 33] é possível encontrar um levantamento dos modelos existentes. As propostas existentes podem ser dividas em três categorias:

- **Comunicação direta entre camadas** (Fig. 2.7(a)). Permitir o compartilhamento de informações em tempo de execução.

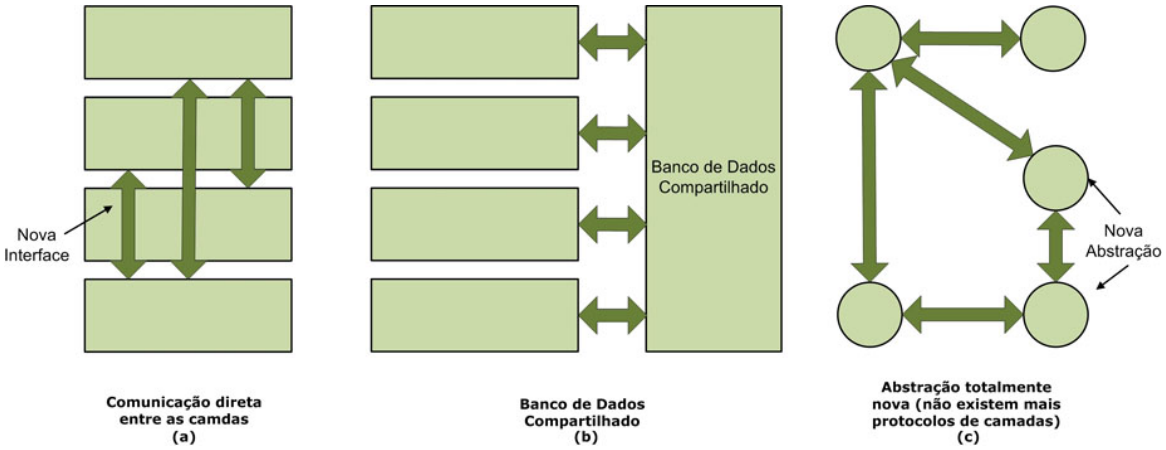

Figura 2.7: Formas de interação entre camadas nas propostas de cross-layer existentes (fonte:[33])

- **Banco de dados compartilhado entre as camadas** (Fig. 2.7(b)). As informações são compartilhadas entre as camadas por meio de um banco de dados, que poderia ser visto como uma nova camada que provê um serviço de armazenar informações de todas as camadas.
- **Nova abstração** (Fig. 2.7(c)). Um abstração completamente nova e diferente, que pode oferecer grande flexibilidade em tempo de execução, mas requer novas implementações. Além de perder a compatibilidade com outros sistemas.

2.3 Padrão IEEE 802.11

As redes podem ser divididas em duas categorias: as que usam conexão ponto a ponto, e as que utilizam canais de difusão (broadcast). Nas redes de difusão é essencial determinar quem tem direito de acessar o canal quando existe disputa por ele. Os protocolos que exercem esse papel pertencem a uma subcamada da camada de enlace de dados chamada de MAC (*Medium Access Control*).

O padrão 802.11 [21] foi elaborado pelo comitê IEEE (*Institute of Electrical and Electronics Engineers*) em meados da década de 1990 com objetivo de padronizar as redes sem fio. Na época em que o processo de padronização começou, o padrão *Ethernet* já havia dominado o mercado de redes locais, assim o comitê decidiu tornar o padrão 802.11 compatível com a *Ethernet* nas camadas acima da host-rede. Esse padrão especifica a camada física e a camada de enlace, segundo os requisitos específicos das redes sem fio. Na pilha TCP/IP, o padrão 802.11 corresponde a primeira camada host-rede.

O padrão 802.11 de 1997 suporta três opções de meio de transmissão para ser utilizado na camada física: uma delas é o infravermelho e as outras duas são baseadas em transmissão de rádio. O método infravermelho utiliza quase a mesma tecnologia que os controles remotos dos televisores. Os outros dois métodos empregam rádio de alcance limitado, utilizando as técnicas FHSS (*Frequency Hopping Spread Spectrum*) e DSSS (*Direct Sequence Spread Spectrum*).

Essas técnicas operam com baixa potência a fim de evitar conflitos e operam com 1 ou 2 Mbps. Em 1999, foram apresentadas duas novas técnicas para alcançar maior largura de banda: OFDM (*Orthogonal Frequency Division Multiplexing*) e HR-DSSS (*High Rate Direct Sequence Spread Spectrum*). Elas operam, respectivamente, em até 54 Mbps e 11 Mbps. Em 2001, uma segunda modulação de OFDM foi introduzida que opera com uma banda de freqüência diferente da primeira. A criação dessas novas técnicas de modulação estabeleceram três novos padrões. O padrão 802.11a utiliza a faixa de freqüência mais larga e opera em velocidades de 54Mbps (OFDM). O padrão 802.11b utiliza a mesma faixa de freqüência que o 802.11, mas emprega uma técnica de modulação diferente para alcançar 11 Mpbs. E o padrão 802.11g utiliza a técnica de modulação do 802.11a, mas emprega a faixa de freqüência do 802.11b.

A camada física é conceitualmente dividida em duas partes: subcamada dependente do meio físico PMD (*Physical Medium Dependent*) e protocolo de convergência da camada física PLCP (*Physical Layer Convergence Protocol*). A subcamada PMD trata da codificação, decodificação, modulação do sinal e das particularidades do meio físico. Já a subcamada PLCP abstrai as funcionalidades que a camada física deve oferecer para a camada MAC. A subcamada PLCP oferece um serviço independente da tecnologia de transmissão, chamado de SAP (*Service Access Point*) e um mecanismo, chamado de CCA (*Clear Channel Assessment*), que verifica se o canal está ou não ocioso.

O protocolo da subcamada MAC do 802.11 difere do protocolo *Ethernet* [34], devido à complexidade inerente ao ambiente sem fio em relação às redes cabeadas. No protocolo *Ethernet*, as estações precisam esperar o canal ficar ocioso para poder transmitir e caso não recebam de volta uma rajada de ruído, é praticamente garantido que os dados foram recebidos com sucesso. Nas redes sem fio, essa situação não ocorre. Por exemplo, na figura 2.8(a), suponha que o nó A esteja transmitindo dados para o nó B, mas que o alcance do rádio do transmissor A seja curto demais para alcançar C. Se C quiser transmitir para B, ele irá escutar o canal, mas o fato de não ouvir nada não significa que sua transmissão será bem-sucedida. Pois os frames transmitidos por C irão interferir com os frames de A destinados para B. Esse problema é conhecido como terminal escondido.

Além disso, existe o problema inverso, o problema da estação exposta, ilustrado na figura 2.8(b). Suponha que B quer transmitir para o nó A. Ao escutar o canal, o nó B conclui de forma errada que não pode transmitir dados para o nó A, embora o nó C esteja transmitindo para o nó D. A transmissão de C para D só geraria uma recepção de má qualidade na zona entre B e C, onde nenhum dos receptores desejados está localizado. Antes de iniciar uma transmissão a estação precisa saber se há alguma atividade no canal. Mas a única informação que pode ser obtida é se há atividade ou não no canal na zona entre o nó emissor e receptor. Nas redes cabeadas, os sinais se propagam para todas as estações, assim, somente uma transmissão pode ocorrer de cada vez. Enquanto, nas redes sem fio, podem existir várias transmissões simultâneas, desde que os destinos sejam diferentes e estejam longe do alcance um dos outros.

Além da subcamada MAC, a camada de enlace é composta por outra subcamada chamada de LLC (Logical Link Control). A figura 2.9 ilustra a divisão da camada de enlace em subcamadas. A subcamada LLC é responsável pelo con-

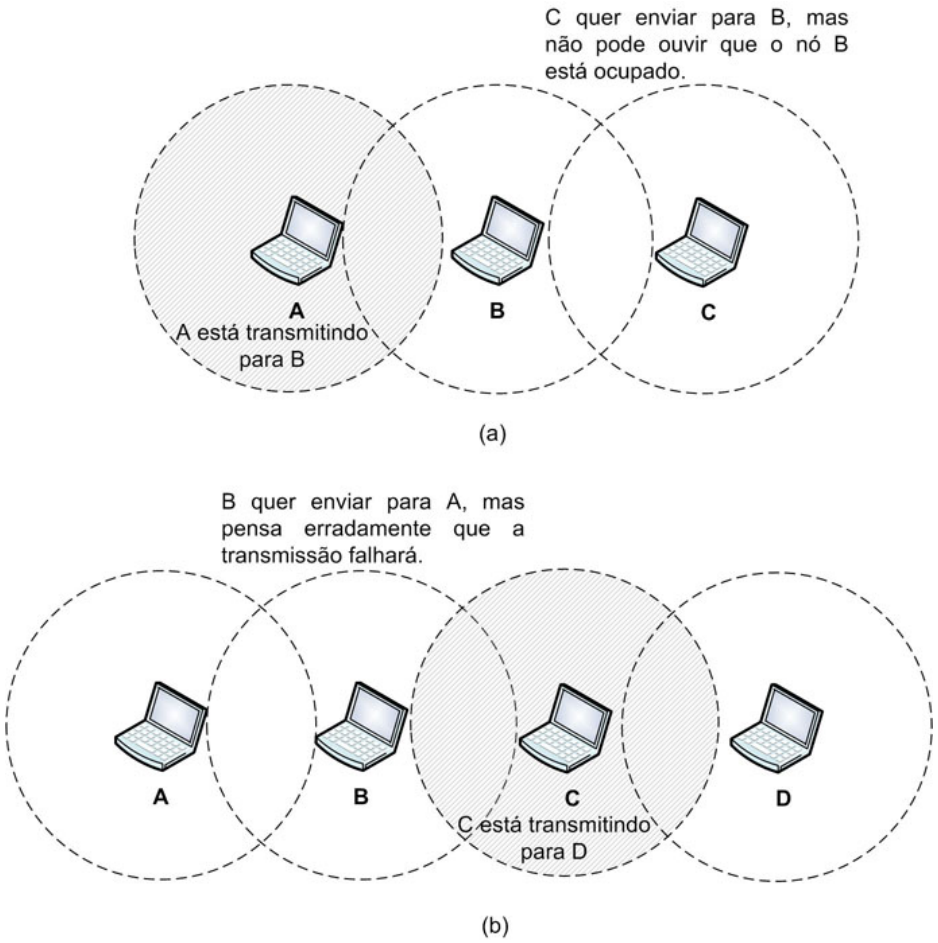

Figura 2.8: (a) Problema do Terminal Escondido. (b) Problema da Estação Exposta.

trole de erros e pelo controle de fluxo. Enquanto a camada MAC é responsável pelo endereçamento, divisão dos dados em frames, e o controle de acesso ao meio, mencionado anteriormente. Além dessas funcionalidades usualmente desenvolvidas pela camada MAC, no padrão 802.11 a camada MAC é responsável por funções tipicamente desempenhadas por protocolos da camada acima como fragmentação, retransmissão de pacotes e *acknowledgement* (ACK) [29].

A camada MAC define dois métodos de acesso: *Distributed Coordination Function* (DCF) e *Point Coordination Function* (PCF).

2.3.1 DCF

O mecanismo básico de acesso, chamado de DCF é na verdade o mecanismo *Carrier Sense Multiple Access with Collision Avoidance* (CSMA/CA). Os protocolos CSMA são bastante conhecidos, o mais popular é o protocolo *Ethernet* que é o protocolo CSMA/CD (CD acrônimo para *Collision Detection*). Este protocolo não pode ser usado no contexto das redes sem fio porque a taxa de erro nas redes sem fio é muito maior do que nas redes cabeadas e, permitir a ocorrência de colisões degradaria sensivelmente a vazão da rede. Além do mais, a detecção

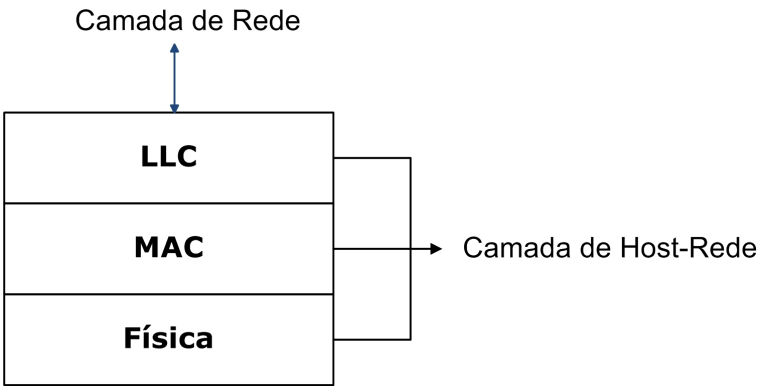

Figura 2.9: Camada de Enlace

de colisões nas redes sem fio nem sempre é possível. Como a maioria dos rádios é *half-duplex*, ou seja, os nós não podem transmitir e ouvir rajadas de ruído ao mesmo tempo em uma única freqüência, não é possível utilizar o protocolo CSMA/CD. O protocolo CSMA, resumidamente, funciona da seguinte forma: a estação que deseja transmitir verifica o canal. Se o canal estiver livre, por um determinado período de tempo, então a estação pode transmitir. Mas se o canal estiver ocupado, ou seja, se outra estação estiver transmitindo, o nó irá adiar a sua transmissão até que o canal esteja novamente livre.

Para diminuir a probabilidade de colisão entre duas estações que não conseguem se ouvir, o padrão define outro mecanismo chamado de *Virtual Carrier Sense* ou simplesmente RTS-CTS, ilustrado na figura 2.10. Em algumas situações um nó pode receber informações de dois outros nós, que não conseguem se ouvir, conhecido também como problema do terminal escondido, descrito anteriormente. Nestes casos o nó receptor é sobrecarregado pelos emissores, o que resulta em colisões e baixa vazão. Os emissores, por sua vez, têm a impressão que o receptor pode receber os pacotes sem nenhuma interferência.

A estação que deseja transmitir um pacote irá primeiro transmitir um pequeno pacote de controle chamado RTS (*Request To Send*). O pacote RTS contém o receptor, emissor e a duração esperada da transmissão do pacote de dados e o respectivo ACK (*Acknowledge*). Após esperar por um tempo pré-determinado chamado SIFS (*Shorter Inter Frame Spacing*), se o nó receptor estiver pronto para receber os dados, ele irá responder ao emissor com um pacote chamado CTS (*Clear To Send*). O pacote CTS também contém o tempo de duração esperado do restante da transação. Todas as estações que receberem o pacote RTS e/ou CTS deverão configurar o indicador *Virtual Carrier Sense*, mais conhecido como NAV (*Network Allocation Vector*) com a duração esperada da transmissão entre os dois nós. O NAV indica o tempo que uma estação deverá esperar até que possa tentar transmitir. O conjunto de estações que receberam o pacote CTS pode ser diferente do conjunto de estações que receberam o pacote RTS, o que indica a presença de terminais escondidos. Esse mecanismo reduz a probabilidade de colisões na área da estação receptora, porque mesmo que a estação não possa ouvir o RTS, quando o nó receptor enviar o CTS, o terceiro nó irá ouvir e reservar o canal como ocupado até o final da transmissão. Uma vez que o pacote RTS foi

enviado e o pacote CTS foi recebido com sucesso, os demais nós foram informados que o canal está ocupado e o nó emissor pode iniciar a transmissão do pacote de dados após esperar por um tempo SIFS. O receptor depois de receber o pacote de dados, espera por um tempo SIFS e envia o pacote ACK. Assim que a transmissão termina, o NAV de cada nó é marcado como livre.

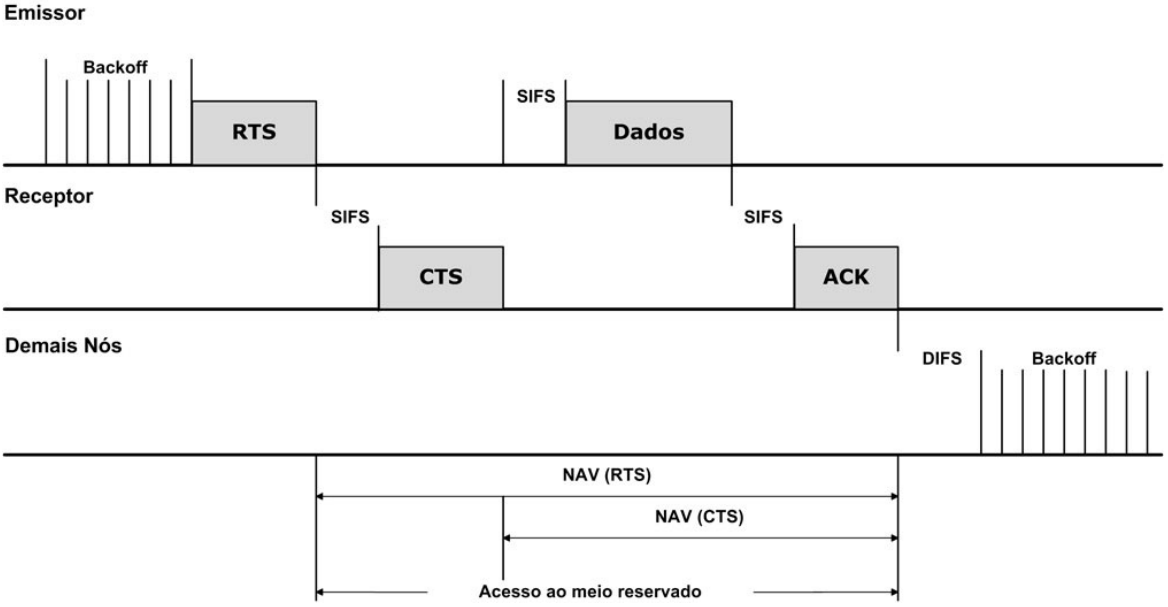

Figura 2.10: Diagrama explicativo do funcionamento dos pacotes CTS e RTS. Fonte: [29].

Uma estação que deseja realizar uma transmissão, como vimos, primeiro deve verificar se o canal está ocupado. Se o meio estiver ocupado, a estação deve ficar em silêncio até que o meio seja determinado como ocioso. Logo que o canal ficar ocioso, o nó deve esperar sem interrupção por um período de tempo igual à DIFS (*DCF Inter Frame Spacing*). Se o canal estiver livre por um tempo DIFS, antes de transmitir a estação deve esperar por um tempo aleatório também chamado de período de *backoff* ou janela de contenção. Cada estação gera um número aleatório de *backoff* que será decrementado a cada slot de tempo, quando o valor do *backoff* for zero a estação pode acessar o canal. Durante o processo de *backoff*, se a estação observar que o canal está ocupado, o contador de *backoff* é interrompido, e continuará a ser decrementado, após esperar um tempo DIFS, se o canal estiver ocioso.

A figura 2.11 ilustra o funcionamento do *backoff*. O nó emissor verifica que o canal está livre e espera por DIFS. Após, o término desse período, o nó percebe que o canal está ocupado e permanece em silêncio até que o canal fique livre novamente. Assim que o canal ficar livre e esperar por DIFS, ele irá decrementar o *backoff* gerado. Mas durante o período de *backoff*, outro nó detém a posse do canal e começa a transmitir. Nesse momento, o nó congela o *backoff* e espera o término da transmissão. Após esperar por um período igual a DIFS, o nó termina de decrementar o *backoff* e inicia sua transmissão.

O valor da janela de contenção (*Contention Window*) pode variar entre CW_{min}

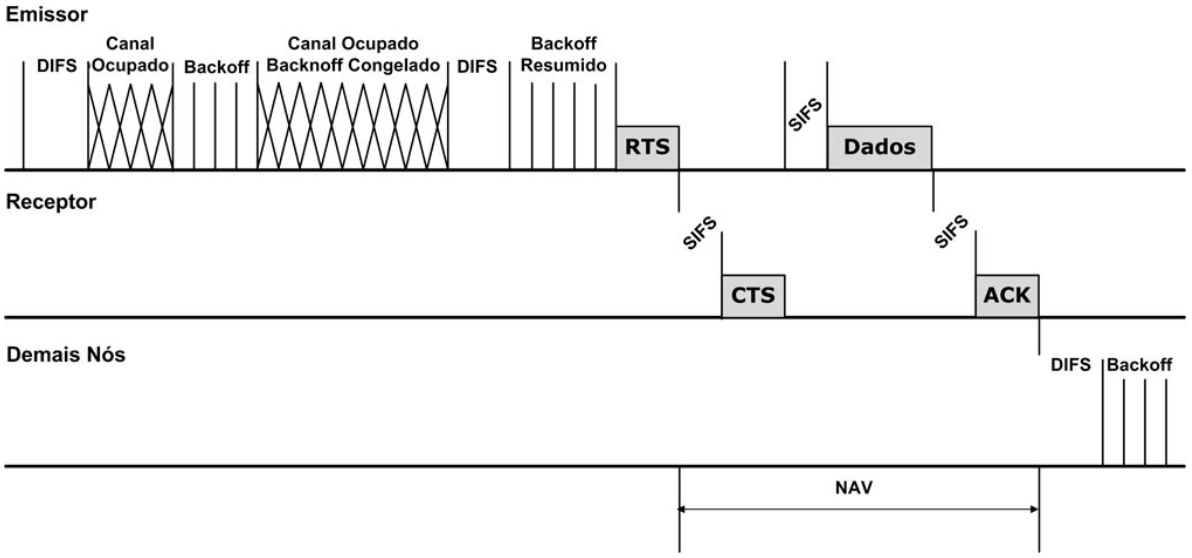

Figura 2.11: Diagrama Explicativo do funcionamento do Backoff. Fonte: [29]

e CW_{max}, esses valores por sua vez dependem do meio físico. O valor inicial da janela de contenção deve ser um número aleatório entre 0 e CW_{min}. A cada colisão o valor da janela de contenção dos nós envolvidos é dobrada até atingir o tamanho máximo da janela CW_{max}. Um exemplo de *backoff* exponencial pode ser visto na figura 2.12.

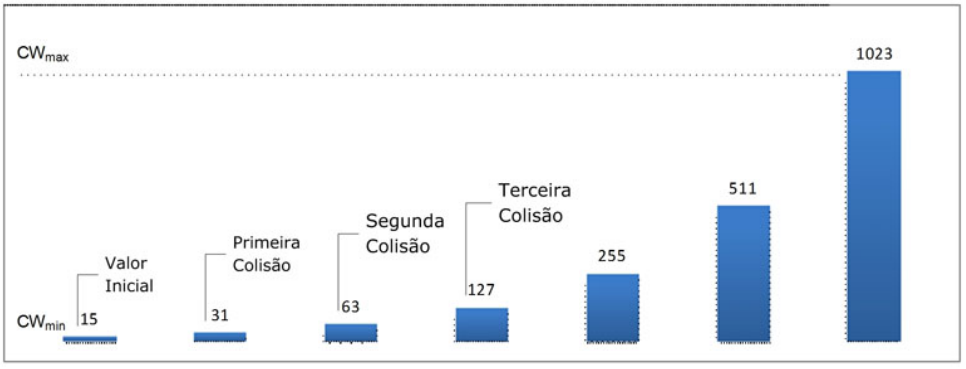

Figura 2.12: Exemplo de Backoff Exponecial

2.3.2 PCF

A camada MAC oferece outro método de acesso opcional chamado PCF que só pode ser utilizado em redes infra-estruturadas. Este método de acesso utiliza um PC (*Point Coordinator*) localizado no ponto de acesso que irá determinar qual estação tem o direito de transmitir em um dado momento. O PCF é basicamente um sistema de eleição onde o PC controla o sistema. A operação em modo PCF pode requerer coordenação adicional caso vários PCs estejam operando simultaneamente. O espaço de intervalo entre os frames utilizado pelo PCF é menor do

que o utilizado no DCF (DIFS). A figura 2.13 mostra a diferença entre os diferentes espaços de intervalo entre os frames. Dessa forma, o tráfego do PCF terá maior prioridade caso os dois modos estejam em ação simultaneamente. Observe na figura 2.14 que em redes infra-estruturadas uma parte do tempo é controlada pelo PC no modo PCF, mas a outra parte do tempo é reservada para permitir que os nós comuniquem-se, entre si, sem o controle do ponto de acesso.

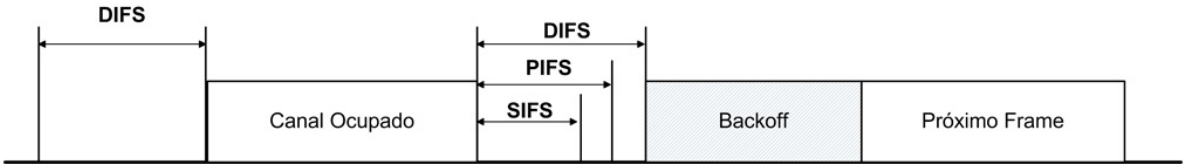

Figura 2.13: Espaço de intervalo entre os frames

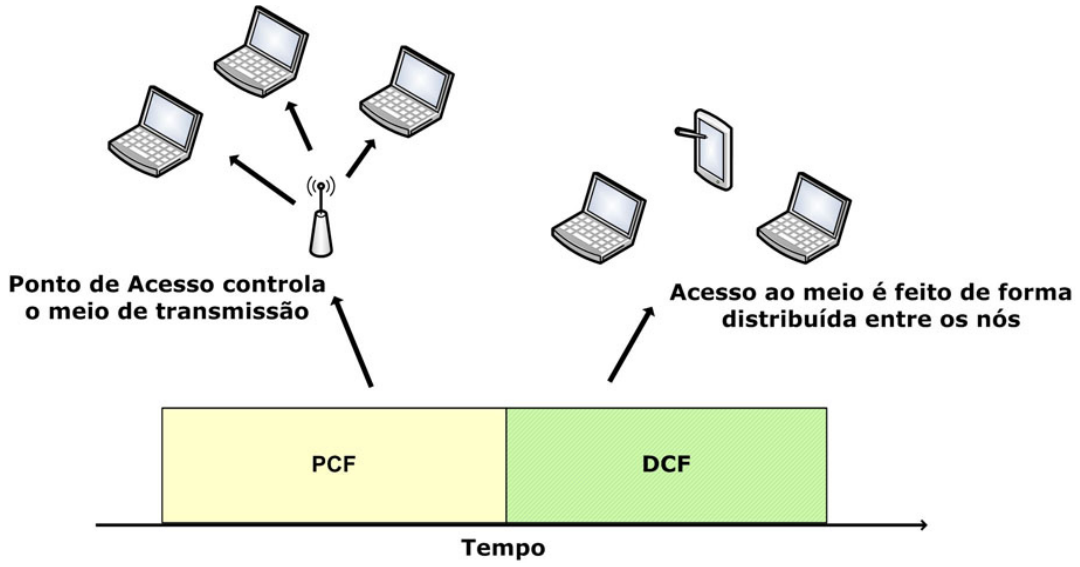

Figura 2.14: Coexistência entre os modos PCF e DCF

2.3.3 802.11e

O padrão 802.11e finalizado em 2005 provê algumas funcionalidades para garantir a qualidade de serviço ou QoS (*Quality of Service*) [36]. O 802.11e utiliza uma versão aprimorada do DCF, o EDCAF (*Enhanced Distributed Channel Access Function*). O EDCAF é bem semelhante ao DCF, ele possui os mesmos princípios do CSMA/CA e do *backoff*, mais inclui funcionalidades para oferecer qualidade de serviço. O tráfego pode ser definido em categorias de acesso, com filas diferentes para cada categoria de tráfego e janelas privilegiadas de transmissão ou TXOP (*Transmission Oportunity*).

O protocolo define quatro categorias de acesso com diferentes prioridades. As categorias, ordenadas de forma crescente (menor prioridade para maior prioridade), são: AC_BK (*Access Control Background*), AC_BE (*Access Control Best Effort*), AC_VI (*Access Control Video*) e AC_VO (*Access Control Voice*) para

os respectivos tipos de tráfego, *background*, melhor esforço, vídeo e voz. Cada uma destas categorias possuem valores para CW_{min} e CW_{max} diferenciados, além disto, é definido um tempo de espera AIFS (*Access Category Inter Frame Spacing*), semelhante ao DIFS, diferente para cada categoria. A figura 2.15 mostra como funciona as filas.

A janela privilegiada de tranmissão (TXOP) define um período de tempo onde uma estação pode acessar o canal de forma direta, ou seja, não é necessário deter o direito de aceso ao meio de transmissão. Esse período tem a finalidade de reduzir as filas com maior prioridade de transmissão.

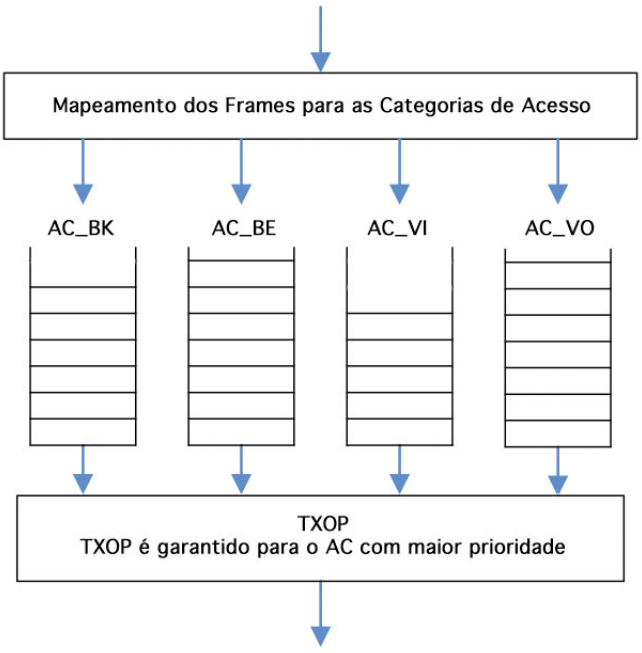

Figura 2.15: Filas de prioridades do padrão 802.11e

2.4 Roteamento

Os protocolos de roteamento nas redes ad hoc precisam ser robustos e eficientes, devido a comunicação em vários saltos (*multi-hop*). Entretanto, encontrar uma rota ideal de comunicação entre dois nós não é um problema trivial, principalmente por haver vários caminhos disponíveis e um grande dinamismo da rede. Existem vários protocolos de roteamento propostos para redes ad hoc. Em [4] pode-se encontrar um estudo de vários protocolos.

Os protocolos de roteamento podem ser divididos em dois tipos: reativos e proativos. A principal característica dos protocolos proativos é a existência de uma tabela com as rotas para todos os outros nós. Uma das vantagens desta abordagem é a disponibilidade imediata de uma rota para qualquer nó. Entretanto, ele pode ser ineficiente nas redes muito grandes e com alta mobilidade. Em uma rede grande, os nós precisam reservar um espaço de memória muito grande para armazenar a tabela de rotas, e em casos de muito dinamismo na rede a atua-

lização da tabela ocorreria constantemente. A simultaneidade de ambos cenários tornam os protocolos proativos inviáveis.

As técnicas de roteamento reativas buscam resolver o problema de roteamento com uma abordagem diferente. Nestes protocolos as rotas só são descobertas quando realmente necessárias. Nas redes grandes e com alta mobilidade, ao contrário dos protocolos proativos, este tipo abordagem é mais eficiente. Pois, não é necessário manter uma tabela com as rotas para cada nó da rede. Sua desvantagem ocorre quando uma rota é necessária, pois será preciso descobri-la primeiro. Enquanto, nos protocolos proativos, a rota já está previamente definida.

Nas redes cabeadas, entretanto, os protocolos proativos são mais eficientes, pois dificilmente os dispositivos desta rede estarão se movendo. Enquanto no ambiente ad hoc esta abordagem é inviável. A seguir será descrito o funcionamento dos protocolos reativos mais utilizados: AODV e DSR.

2.4.1 AODV

O protocolo de roteamento AODV (*Ad Hoc On-Demand Distance Vector*) é um protocolo reativo e basicamente tenta descobrir uma rota enviando uma mensagem em *broadcast* (para todos os nós) perguntando pelo nó destino. Ao receber a mensagem, o nó responde com uma mensagem *unicast* (somente para o nó emissor). A mensagem de resposta contém todas as rotas encontradas.

Cada nó possui uma tabela de roteamento onde é armazenado informações de roteamento dos nós vizinhos que estejam ao seu alcance (*single-hop*). Cada tabela possui uma validade, se a rota não for utilizada no período determinado pela validade, a rota é expirada. Caso seja utilizada, o período de validade é aumentado. Quando o nó remetente possui pacotes para enviar a um nó destinatário, primeiro ele verifica se o nó destinatário está presente na sua tabela de rotas, se estiver envia o pacote, senão inicia o processo de descoberta de rota. Para descobrir essa rota, o nó envia um pacote RREQ (*Route REQuest*) para todos os seus vizinhos. Quando um nó recebe o pacote RREQ ele marca que o nó de quem ele recebeu o RREQ é o próximo salto para o nó remetente e incrementa o contador de saltos presente no pacote RREQ. Depois disso os vizinhos o repassam para os seus respectivos vizinhos, e conseqüentemente todos os nós da rede receberão o pacote RREQ.

Quando o pacote chegar ao destinatário, o nó terá a informação de qual vizinho possui a menor distância em saltos ao nó remetente e responderá com um pacote RREP (*Route REPly*) para este nó, que irá repassar para o nó vizinho com menor distância, e assim sucessivamente. A figura 2.16 mostra como este processo funciona.

Quando a rota é descoberta, ela é utilizada pelo maior tempo possível. Entretanto, devido a alta mobilidade da rede, alguns nós podem mudar de posição ou serem desligados. Neste caso, o nó intermediário que não conseguir encontrar o próximo nó da rota irá enviar uma mensagem RRER (*Route Error*) para o nó remetente.

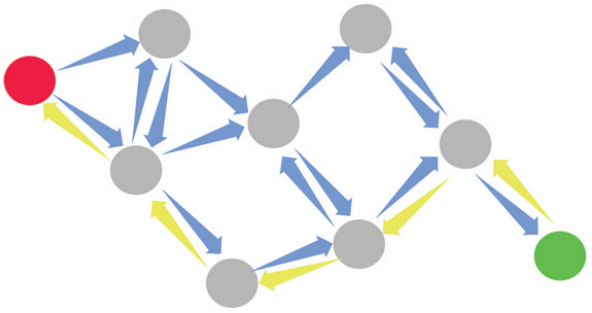

Figura 2.16: Processo de descoberta de rotas AODV. Setas azuis são pacotes RREQ, amarelas RREP. O nó vermelho é o remetente e o verde o destinatário.

2.4.2 DSR

O protocolo DSR (*Dynamic Source Routing*) é bem semelhante ao AODV, com algumas diferenças importantes. Dentre elas, a mais importante, ocorre ao enviar o pacote RREQ. Os nós intermediários colocam no pacote RREQ a rota completa pela qual o pacote passou. Assim, o nó destinatário saberá exatamente a rota utilizada, e irá utilizá-la para enviar os pacotes subsequentes.

Ao invés de manter uma tabela de rotas, o DSR mantém um cache de rotas e também permite a existência de múltiplas rotas entre dois nós. Dessa forma, caso alguma delas seja desativada, será possível utilizar a outra rota rapidamente. A outra diferença é em relação a validade de rotas. No protocolo DSR, a rota que for utilizada por um longo período não será apagada.

2.5 Segurança em Redes Ad Hoc

Esta seção descreve os problemas de segurança nas redes ad hoc, em decorrência de suas características: canal compartilhado e falta de infra-estrutura. Existem inúmeros ataques possíveis, apenas os mais relevantes para esse trabalho serão apresentados.

2.5.1 Premissas de Segurança

A segurança em comunicação possui quatro objetivos principais: disponibilidade, integridade, legitimidade e o irreputabilidade. Qualquer protocolo que tente proteger alguma informação enquanto for transmitida por um canal de comunicação deve permitir que os usuários dos dispositivos móveis alcancem estes objetivos. Estes protocolos devem garantir confiabilidade, que é a garantia de que a informação não será acessada por indivíduos não autorizados. Também é necessário garantir a integridade, isto é, garantir que a informação não seja alterada durante a transmissão. O protocolo deve também garantir a legitimidade dos nós, ou seja, os usuários precisam ter certeza que estão se comunicando com quem acreditem estar. E por fim, a irreputabilidade deve garantir que o emissor de uma mensagem não negue a não autoria da mensagem, essa garantia é comumente realizada por meio de assinaturas digitais.

2.5.2 Problemas

Como descrito nas seções anteriores, o canal de transmissão não possui a proteção física dos fios presentes nas redes tradicionais. Dessa forma, as redes são mais suscetíveis a ataques, pois podem surgir de inúmeras direções. Alguns dos possíveis ataques são: vazamento de informações, modificação, fabricação de mensagens e nós que se disfarçam afirmando ser um outro nó.

Os ataques em redes ad hoc podem ser classificados em duas categorias: passivos e ativos. Um ataque passivo não altera o funcionamento da rede, o nó adversário apenas escuta o canal para tentar descobrir alguma informação valiosa sem alterar os dados. Nesse caso o sigilo será violado caso o nó adversário consiga interpretar os dados coletados. Se todos os dados transmitidos forem cifrados, derivar as informações úteis torna-se um pouco mais complicado. A detecção de ataques passivos é mais difícil, uma vez que a operação da rede não é afetada.

Um ataque ativo, por sua vez, tem como objetivo prejudicar o funcionamento da rede. Os ataques ativos envolvem ações como a replicação, modificação, e remoção dos dados transmitidos na rede. Podem ser agrupados em: personificação, negação de serviço e ataque de revelação *(disclosure attack)*. Os nós que realizam ataques ativos são chamados de nós maliciosos.

No ataque de personificação, o nó adversário assume a identidade e os privilégios de um outro nó da rede com o intuito de consumir os recursos da rede que podem não estar disponíveis ou atrapalhar o normal funcionamento da rede acrescentando outras informações. Neste caso, o nó poderia adivinhar a identidade e os detalhes de autenticação de um outro nó autorizado (nó alvo), ou poderia escutar o canal para encontrar alguma informação de identidade e detalhes de autenticação do nó alvo em uma comunicação anterior com outro nó, ou ainda desabilitar o sistema de autenticação do nó alvo. Um outro tipo de ataque de personificação é o *man-in-the-middle*, onde o nó adversário lê e modifica as mensagens entre dois nós sem deixar que ambos os nós envolvidos na comunicação saibam que estão sendo atacados.

Outros tipos de ataques podem servir como subsídio para o ataque de personificação como o ataque do buraco negro *(blackhole)* e o ataque de *wormhole*. No ataque de *blackhole*, nós maliciosos anunciam rotas falsas (menores ou supostamente mais estáveis) durante o processo de descoberta de rotas, para interceptar os pacotes do nó alvo ou para não fazer parte de nenhuma rota na rede. Esse ataque possui como objetivo evitar o consumo dos seus recursos como bateria. Os nós que realizam esse tipo de ataquem são chamados de nós egoístas. No ataque de *wormhole* o nó adversário recebe os pacotes em um local da rede e faz uma espécie de túnel desses pacotes para outro local da rede, isso ocorre tipicamente se houver conluio de nós na rede.

O ataque de negação de serviço consiste em impedir que os nós acessem os serviços da rede. O ataque clássico de negação de serviço é a inundação de pacotes em um algum ponto da rede, tipicamente os nós centrais da rede são os alvos. Nas redes ad hoc um nó com papel central seria um nó presente em todas as rotas (gargalo da rede). Devido as características próprias das redes ad hoc, outras técnicas de negação de serviço que não poderiam ser realizadas nas redes cabeadas, podem ser aplicadas. O ataque de negação de serviço pode ser realizado em

qualquer camada. Na camada física, por exemplo, um nó adversário pode emitir um sinal qualquer e provocar interferência em alguma transmissão em algum ponto da rede. Na camada de rede, o nó adversário pode fazer parte de uma rota e explorar o protocolo de roteamento para atrapalhar o funcionamento da rede, como o simples *drop* de um determinado número de pacotes, ou ainda consumir a bateria dos outros nós com inúmeras requisições desnecessárias, conhecido como *sleep deprivation torture*.

O ataque de revelação *(disclosure attack)* consiste em descobrir ou revelar informações importantes da rede como topologia e localização geográfica dos nós. Nesse caso dois mecanismos de segurança podem ser aplicados, um mecanismo preventivo baseado em algoritmos criptográficos com distribuição de chaves e um mecanismo reativo como os sistemas de detecção de intrusão.

As características únicas das redes ad hoc requerem que os mecanismos de segurança tenham alguns requisitos. Os mecanismos de segurança devem:

1. Ter como objetivo incentivar a cooperação entre os nós, devem ser distribuídos e devem-se auto-organizar.

2. Consumir poucos recursos para não degradar o desempenho das redes ad hoc, uma vez que os nós possuem recursos limitados.

3. Ser confiáveis e possuir escalabilidade.

2.5.3 Soluções Propostas na Literatura

Algumas soluções para os problemas de segurança em redes ad hoc podem ser encontradas na literatura. Uma solução estática para os problemas de segurança não é suficiente devido ao dinamismo dessas redes. Assim, as soluções não podem conter entidades centrais de gerenciamento como, autoridades certificadoras, firewalls e agentes de gerência baseados em MIBs. Como os mecanismos de segurança devem ser realizados de forma distribuída, a complexidade dos algoritmos aumenta.

Para manter o sistema funcionando com a presença de nós com mau comportamento algumas soluções foram propostas:

- Roteamento seguro utilizando criptografia. Exemplo de protocolos: Ariadne [20], SRP [30] e ARAN [32].

- Uso de incentivos econômicos, ou seja, nós egoístas são pagos para fazer roteamento [19]. Exemplo de protocolos: Nugglets [9] e Sprite [38].

- Sistemas de Reputação e Confiança [19, 5, 18, 28].

As duas primeira soluções não resolvem todos os problemas, pois, mudanças silenciosas de uma rota não poderiam ser detectadas. A idéia por trás dos sistemas de reputação e confiança é monitorar os nós da rede e verificar o seu comportamento em relação a algum protocolo previamente definido e permitir que nós vizinhos troquem suas observações com outros nós. Além disso, esses sistemas devem tomar uma medida coletiva contra o nó mal comportado.

Neste capítulo vimos as principais características do ambiente ad hoc e seus problemas associados. A alta mobilidade dos nós e a facilidade de configuração são os principais benefícios das redes ad hoc. Porém, suas características como falta de infra-estrutura, recursos limitados, topologia dinâmica, comunicação em *broadcast* acrescentam alguns desafios principalmente em relação a segurança que têm sido bastante explorado no meio científico. Além disso, vimos alguns dos problemas em utilizar a arquitetura em camadas nas redes ad hoc e algumas propostas existentes de *cross-layer*. Foi apresentado também como ocorre o funcionamento do padrão 802.11e e como é feito o controle de acesso ao meio por meio do uso de RTS/CTS e *backoff*. Finalmente, vimos os principais protocolos de roteamento nas redes ad hoc e alguns dos problemas de segurança em redes ad hoc.

Capítulo 3

Detecção de Desvio de Conduta

Neste capítulo veremos alguns dos tipos de desvio de conduta, ou mau comportamento existentes nas camadas de transporte, rede e enlace. Depois veremos, ainda neste capítulo alguns mecanismos existentes na literatura para identificar nós maliciosos que agem na camada de enlace. E no próximo capítulo veremos alguns mecanismos existentes para minimizar o efeitos de nós maliciosos na camada de rede.

3.1 Camada de Transporte

Mau comportamento na camada de transporte em redes ad hoc, que sejam únicos deste contexto, ainda não foram identificados [26]. Os problemas que podem ocorrer nesta camada são basicamente os mesmos que podem ocorrer com o mecanismo de congestionamento do TCP em redes cabeadas.

Uma forma de mau comportamento nesta camada, seria um nó não obedecer ao algoritmo de congestionamento, e enviar dados com uma taxa maior que os demais, forçando que o tráfego concorrente seja atrasado ou descartado. Por outro lado, poderíamos ter o nó receptor como malicioso. Por exemplo, o nó receptor poderia ao receber um pacote de N bytes, dividir o pacote resposta em M diferentes ACK, onde $M \leq N$, cada um cobrindo os $\frac{M}{N}$ bytes do pacote recebido, e consequentemente o emissor irá aumentar a janela de congestionamento e irá transmitir para o nó receptor a uma taxa M vezes maior [26].

Em redes ad hoc os problemas em camadas inferiores, como da camada de rede, podem afetar o protocolo TCP. Uma falha de rota pode ser interpretada como problema de congestionamento e alterações de rotas podem fazer com que os pacotes TCP cheguem fora de ordem. Para resolver estes problemas é necessário que o mau comportamento das camadas inferiores seja impedido.

3.2 Camada de Rede

Na camada de rede os nós mal comportados podem agir de várias maneiras [13]:

- **Não participar do processo de descoberta de rotas.** Um nó malicioso poderia não encaminhar os pacotes de *route request* e *route reply*. O nó

poderia, também, alterar o TTL (*Time To Live*), ou seja o número de nós pelos quais o pacote pode passar, para o menor valor possível. Com isso o nó prejudicaria o funcionamento do protocolo de roteamento. Caso a melhor rota passasse pelo nó atacante, o nó que gerou o RREQ não iria poder usá-la.

- **Modificar a topologia de roteamento** O nó poderia modificar o pacote *route request* alterando a rota percorrida adicionando saltos inexistentes. Conseqüentemente, o nó gerador do RREQ receberia uma rota inexistente.

  O nó poderia também modificar os pacotes *route reply* trocando sua identidade no pacote por outro nó. Também poderia causar um desvio, ou retornar uma rota inexistente, ou inserir saltos adicionais, como consequência o nó que pediu a rota receberá uma rota inexistente ou uma rota que não seja a mais a curta.

- **Não participar mais de uma rota vigente.** Se o nó estiver participando de uma rota, poderia não confirmar o recebimento de pacotes não enviando ACKs, causando um erro no roteamento. Com isso o nó remetente iria ter que descobrir outra rota.

- **Não encaminhar pacotes.** O nó poderia não encaminhar nenhum pacote.

3.3 Camada MAC

Alguns mecanismos existentes para minimizar o efeito do nó mal comportado, que serão descritos no próximo capítulo, concentram-se em observar o mau comportamento de um nó apenas na camada de rede. Isto é um problema pois um nó egoísta pode também agir em outras camadas, como por exemplo, na camada de enlace. Veremos na próxima seção alguns sistema de detecção de mau comportamento na camada MAC. Um nó malicioso pode agir na camada MAC de várias formas [15]:

- **Negação de serviço**. Para causar negação de serviço na rede, basta que o nó malicioso provoque colisões propositais no canal, de tal forma que nenhum nó consiga transmitir. Pois, tanto o nó malicioso quanto os demais nós seriam impedidos de transmitir. Neste caso, muito pouco pode ser feito para impedir estes ataques e identificar o nó malicioso. A menos que o nó atacante não deseje transmitir, para ele seria vantajoso causar essas colisões. A única solução seria localizar o nó e desativá-lo manualmente, mas essa solução torna-se inviável, pois não é possível obter do canal os nós que sofreram colisões.

- **Colisões nos frames CTS, ACK e de dados**. O nó malicioso pode provocar colisões com os frames CTS, ACK e de dados para forçar o nó emissor a dobrar sua janela de *backoff*. Como consequência, o nó atacante aumentaria a sua probabilidade de obter o canal e transmitir os seus dados.

- **Início da transmissão antes do término do DIFS**. Como vimos, nas seções anteriores, antes de transmitir o nó deve esperar por um tempo DIFS. Um nó malicioso pode esperar um tempo menor que o DIFS. Assim, ele obterá o canal bem antes das outras estações e poderá transmitir os seus dados. Este problema pode ser facilmente identificado. Como o DIFS é padrão, basta observar o tempo que o nó suspeito esperou antes de transmitir.

- **Redução da janela de contenção máxima(CW_{min})**. O nó malicioso pode alterar o valor padrão de CW_{min} definido pelo padrão 802.11. Assim, o valores de *backoffs* gerados serão menores que os valores obtidos pelos demais nós. Consequentemente, ele teria maior probabilidade de obter o canal. O foco do nosso trabalho é o estudo desse problema.

- **Manipulação do NAV**. O nó malicioso poderia alterar o valor do NAV para um período maior que sua transmissão, consequentemente os outros nós esperarão um tempo maior para voltar a decrementar o *backoff*.

3.4 Sistemas de Detecção na Camada MAC

3.4.1 DOMINO

O DOMINO (*System for Detection Of greedy behavior in the Mac layer of IEEE 802.11 public NetwOrks*) [31] é um sistema de detecção de mau comportamento na camada MAC. Basicamente, o seu enfoque é detectar e identificar as estações egoístas que aumentam a sua largura de banda em detrimento de outros usuários na rede. O DOMINO leva em consideração a existência de um ponto de acesso onde estará presente o sistema de detecção dos nós maliciosos. Este sistema de detecção é realizado em duas etapas: coleta de dados e identificação do nó malicioso. A primeira etapa, chamada de período de monitoramento, é feita em intervalos de tempo regulares onde o tráfego das estações emissoras é coletado. A segunda etapa verifica a existência de seis tipos diferentes de mau comportamento na camada MAC em cada estação, a partir dos dados coletados no primeiro estágio.

O primeiro teste verifica se o número de retransmissões é menor do que a média de retransmissões realizada pelas outras estações. Uma estação maliciosa poderia provocar a colisão de frames CTS para forçar que o canal fique ocioso logo após a colisão e transmitir os seus dados, ou ainda provocar a colisão dos frames de dados e ACK para forçar o nó a dobrar a sua janela de *backoff*. Assim o número de retransmissões do nó malicioso seria muito menor do que das demais estações. Os demais testes verificam se houve algum tipo de manipulação nos parâmetros do protocolo.

Logo após o término da transmissão de dados de uma estação qualquer, todos os nós devem esperar por um período fixo de tempo chamado DIFS antes de transmitir. Caso o nó transmita antes desse período de tempo sucessivas vezes, podemos considerá-lo malicioso. É este comportamento que o segundo teste irá identificar, ou seja, se o período de tempo ocioso após o último ACK ser enviado

é menor do que o DIFS. O terceiro teste compara o valor do NAV no cabeçalho dos frames RTS e/ou dos frames de dados com o valor real da transmissão. Dessa forma, o nó malicioso deixaria o canal reservado por mais tempo e obteria o canal novamente logo após a transmissão do ACK. O valor de *backoff* é gerado de forma aleatória no intervalo $[0, CW]$, segundo o padrão 802.11. É esperado que o valor máximo de *backoff* gerado por uma estação esteja próximo de CW após um grande número de amostras, desconsiderando as colisões. Assim, um nó que possua o valor máximo de *backoff* menor que um valor limite, pode ser considerado suspeito. Esse teste pode ser facilmente enganado caso o nó malicioso gere um *backoff* aleatório próximo do valor limite estabelecido.

O teste do valor atual de *backoff* consiste em estimar o valor de *backoff* gerado por uma estação entre duas transmissões consecutivas e verificar se a média dos valores de *backoff* estimados desse nó é menor do que um valor de *backoff* nominal. Esse valor de *backoff* nominal pode ser obtido pela média de *backoffs* do ponto de acesso caso haja tráfego suficiente ou de forma analítica, baseada em probabilidades. A estimativa desse valor de *backoff* é a soma de todo o intervalo ocioso entre duas transmissões consecutivas de um nó N. As colisões não são consideradas, uma vez que não é possível identificar os nós emissores dos frames que sofreram colisão. O último teste é bem semelhante ao anterior, a diferença está na forma de estimar o valor do *backoff*. Se o tráfego for TCP é provável que exista o controle de congestionamento, então nesses casos existem alguns atrasos entre os frames que podem acabar sendo contabilizados no valor do *backoff* de forma errada. Dessa forma, além do teste do valor atual de *backoff*, existe o teste do *backoff* consecutivo que considera valor de *backoff* como sendo o tempo ocioso entre dois frames consecutivos sem frames intercalados de um nó N.

O DOMINO foi desenvolvido para redes sem fio infra-estruturadas. As soluções propostas por ele podem ser aplicadas para um ambiente ad hoc, mas nem todas são adequadas.

Em relação ao ataque de colisão nos frames RTS e CTS, o DOMINO identifica um atacante observando o número de retransmissão de cada nó. Se um nó fizer este ataque, ele terá menos retransmissão que os demais. Entretanto, essa política é falha, visto que, dependendo da topologia da rede alguns nós poderão ser classificados de forma errônea. Por exemplo, suponha uma topologia que possua uma área pequena com muitos nós e um nó afastado dos demais. Na área concentrada de nós haverá muitas colisões, logo cada nó fará muitas retransmissões. O nó afastado, entretanto, terá poucas retransmissões e poderá ser classificado como malicioso. A figura 3.1 ilustra esse cenário, na área próximo ao nó **c** o número de colisões será bem maior do que nas regiões próximas ao nó **a** e **b**.

Para estudar o problema de manipulação do *backoff* é preciso saber quanto um nó pode ganhar reduzindo sua janela de contenção do *backoff*. A forma mais intuitiva para analisar se um nó está agindo maliciosamente modificando seu *backoff* é observando a vazão de dados do nó e comparando com a vazão dos nós normais. A solução proposta pelo DOMINO para este problema, consiste na estimativa do *backoff* de cada nó. Mas, essa abordagem não é adequada para um ambiente ad hoc onde recursos são limitados. Neste trabalho propomos que a melhor solução é observar a vazão dos nós. Entretanto, tanto DOMINO quanto o framework proposto em [15] criticam o uso de vazão. O argumento utilizado, por

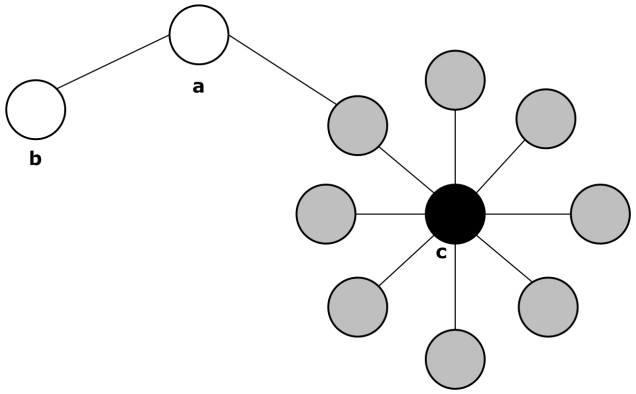

Figura 3.1: Exemplo de uma topologia que ilustra o problema de observar o número de retransmissões de cada nó

eles, é que diferentes tipos de tráfego possuem taxas de transmissão diferentes, e por isso seria dificil comparar um nó com outro sem saber o tipo de tráfego de cada um. Todavia, estimar o *backoff* do nó não é eficiente e possui um custo muito alto para o nó observador.

O uso da vazão para analisar o comportamento de um nó pode ser usado com eficiência se soubermos o tipo de tráfego de cada nó. O protocolo 802.11e oferece exatamente isso.

O 802.11e ainda é muito recente e não está amplamente difundido. Mas a tendência atual é que o 802.11e se espalhe e seja adotado pelas redes ad hoc. Quando isso acontecer será possível identificar o tipo de tráfego de cada estação e assim, será viável utilizar a vazão dos dados para comparar o comportamento dos nós. Entretanto, existe uma probabilidade do nó malicioso caracterizar propositalmente um tipo de tráfego não prioritário com uma categoria de maior prioridade. Uma provável solução para esse problema seria a abertura dos pacotes do nó suspeito, para verificar se o conteúdo é coerente com a prioridade definida, pelo nó observador. Porém, em um ambiente ad hoc isto é inviável. Considerando que o nó malicioso está comunicando com outro nó, o nó destino poderá facilmente saber se o nó emissor utilizou a categoria de prioridade correta. Se ele verificar que a prioridade está incorreta, ele poderia enviar um alerta para os nós da rede. Essa solução se inviabilizaria, caso o nó destino agisse em conluio com o nó malicioso. Outra solução poderia ser feita quando uma estação, ao verificar que outro nó está sempre transmitindo pacotes com alta prioridade por muito tempo, capturasse alguns pacotes para verificar se o conteúdo é coerente com a prioridade.

3.4.2 Detecção de Mau Comportamento na Camada MAC (Kyasanur *et. al*)

Kyasanur *et. al* em [24] propõe um esquema de detecção de nós egoístas na camada MAC que exige modificações no modo de operação DCF definido pelo padrão 802.11. Nesse modelo, assume-se que os nós receptores possuem bom comportamento e não existe conluio entre os nós emissores e receptores. O es-

quema proposto é divido em três componentes. Primeiro, o nó receptor verifica se houve desvio no protocolo. Caso tenha ocorrido algum desvio, o nó receptor penaliza o nó emissor de acordo com a magnitude do desvio realizado para aquela transmissão. No decorrer do tempo, o nó receptor caracteriza o nó emissor como egoísta baseado na magnitude dos desvios realizados durante as transmissões.

Nesse modelo, o nó receptor determina o valor de *backoff* do nó emissor, que deve ser enviado junto com os pacotes CTS ou ACK. O valor de *backoff* é gerado de forma aleatória entre $[0, CW_{min}]$. Mas, é possível que o nó emissor não respeite o valor determinado e utilize um valor de *bakoff* menor que o estipulado sendo, portanto, considerado um nó egoísta. O nó receptor R irá observar o canal e contar os slots de tempo ocioso do nó emissor S, durante o intervalo de tempo entre o envio do ACK por R e a próxima recepção do frame RTS por S. Um desvio do protocolo é caracterizado caso o valor observado seja menor que o valor esperado, ou seja, $B_{act} < \alpha \times B_{exp}$, onde $0 < \alpha \leq 1$, onde B_{act} é o valor do *backoff* atual e B_{exp} o valor do *backoff* experado. A adição da fração α é necessária para minimizar falsos positivos, pois as condições do canal podem ser diferentes para o nó emissor e receptor.

Além disso, foi acrescentado no pacote RTS um campo que indica o número de tentativas de retransmissões. O valor contido nesse campo é incrementado pelo nó emissor a cada transmissão mal sucedida. E será utilizado pelo nó receptor para aumentar o espaço amostral dos possíveis valores de *backoff* e reduzir o número de colisões. O esquema de correção é utilizado para penalizar os nós que desviam do protocolo, de tal forma que o próximo valor de *backoff* determinado pelo nó emissor será a soma do valor aleatório gerado mais uma penalização P que está em função da magnitude do desvio. Os desvios observados são coletados durante o tempo e se a soma dele for maior do que um limiar, o nó será considerado mal comportado.

3.5 Sistemas de Detecção *Cross-layer*

3.5.1 *Cross-layer Framework*

Lei Guang *et. al.* [15] propõem um *framework* com interação entre as camadas (cross-layer) com o objetivo de facilitar a detecção e a reação contra o mau comportamento dos nós na camada MAC.

O *framework* proposto atua basicamente em três camadas: MAC, rede, aplicação. A figura 3.2 ilustra a arquitetura proposta por Lei Guang *et. al.*. A arquitetura desse *framework* é constituída por três componentes. A camada MAC possui um sistema de detecção chamado de *Primary Misbehavior Detection System (MDS - p)* para detectar o mau comportamento segundo algumas métricas e para criar uma lista de *trust*. A camada de rede, por sua vez, utiliza a lista de *trust* gerada pelo *MDS - p* para descobrir rotas confiáveis e prevenir eventuais danos provocados por um nó malicioso. A camada de aplicação é composta por outro sistema de detecção, chamado de *Secondary Misbehavior Detection System (MDS - s)* que faz o controle de fluxo e complementa o primeiro método, *MDS – p*.

Além disso, na camada MAC, Lei Guang *et. al.* propõe a utilização do

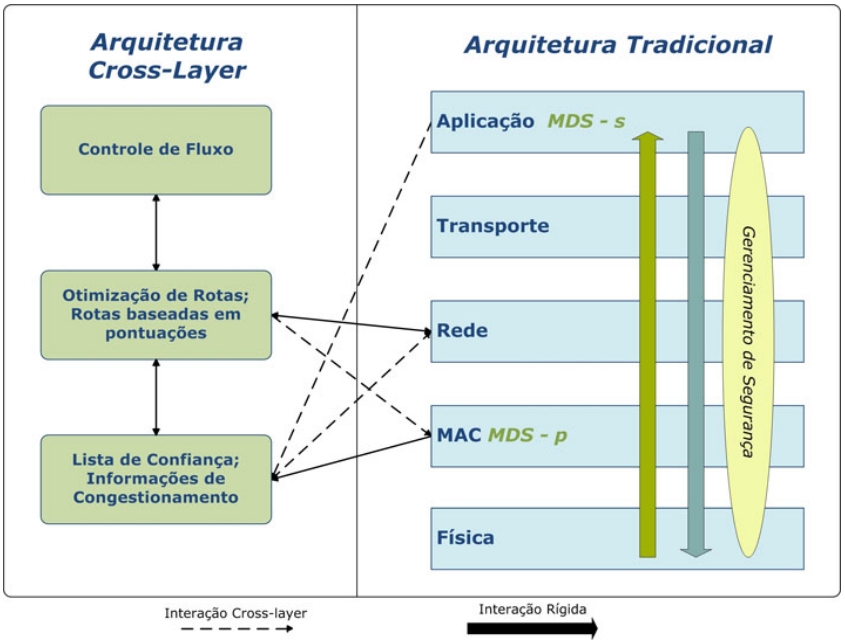

Figura 3.2: Arquitetura do Framework *Cross-layer*. Fonte: [15]

protocolo denominado *Pretictable Random Backoff (PRB)* [16, 17] para minimizar os danos causados pela alteração do intervalo de *backoff* do nó malicioso, visto anteriormente. No protocolo PRB, o nó gera um valor aleatório de *backoff* entre o intervalo $[CW_{lb}, CW]$, onde CW_{lb} é o menor valor possível da janela de *backoff* obtido em função do valor anterior de *backoff* gerado. Logo, o CW_{lb} do próximo *backoff*, $cw_i + 1$, onde cw_i representa o valor de *backoff* gerado pelo nó, será $CW_{lb} = \alpha \times cw_i$. Os componentes do MDS são:

- ***Monitor Module***. Esse módulo monitora as transmissões e coleta os dados relevantes, como *timeout* e duração do tempo ocioso.
- ***Pattern Module***. Verifica se o tráfego observado é normal ou anormal.
- ***Detection Module***. Identifica o nó malicioso.
- ***Trust Module***. Gera uma lista segundo os diferentes tipos de mau comportamento que pode ser posteriormente utilizado em conjunto com o módulo de reação.
- ***Reaction Module***. Depois de identificar o nó malicioso, esse módulo irá minimizar o impacto de sua presença na rede. Para uma reação ativa o nó normal poderá descobrir rotas mais confiáveis, e para uma reação passiva o nó pode implementar um esquema de prevenção com a utilização do protocolo PRB.

Lei Guang *et. al.* em [15] critica o uso da vazão como métrica para analisar o mau comportamento do nós. Eles utilizam o mesmo argumento do DOMINO, ou seja, que a existência de diferentes tipos de tráfego com taxas de transmissão

diferentes não é confiável, visto que algumas aplicações como voz, vídeo possuem maior vazão que outras. Assim, um nó que esteja rodando essas aplicações poderia ser classificado de forma errônea como mal comportado. Além disso, Lei Guang *et. al.* afirma que a vazão também pode ser afetada por outros fatores como injustiça, SNR (*Singal-to-Noise Ratio*), potência do dispositivo, implementação do protocolo no sistema operacional entre outros. Entretanto também afirmam ser possível utilizar essa métrica caso seja conhecido os tipos de aplicação rodando em cada nó.

Esse framework com interação entre camadas utilizam duas métricas para identificar os nós maliciosos: *backoff* e *timeout*. O *backoff* é medido de forma semelhante ao DOMINO. Segundo a percepção do nó receptor, o *backoff* do nó emissor pode ser calculado contando o período de tempo ocioso a partir da última transmissão bem sucedida desse nó. Caso o nó continue selecionando valores de *backoff* pequenos, o nível de confiança do emissor mantido pelo nó receptor será decrementado até atingir um limiar pré-definido. A outra métrica utilizada é *timeout*. Segundo o padrão 802.11, o nó emissor deve configurar um intervalo de *timeout* para o *frame* esperado a fim de evitar a espera sem fim por uma resposta do nó receptor. Por exemplo, um nó receptor irá computar o valor de *timeout* esperado pelo *frame* de dados, assim que enviar o *frame* CTS. Caso o *frame* de dados chegue depois de espirado o tempo de *timeout* definido, esse comportamento será considerado anormal e o valor de confiança do nó emissor será decrementado.

Cada nó mantém uma lista de confiança para todos os nós que tenham uma ligação direta. O valor de confiança de cada nó é iniciado com T_0. O nó monitora a transmissão dos *frames* MAC entre ele e os outros nós da rede. E após o período de monitoramento, realiza o *update* da lista de *trust*. Para cada transmissão considerada anormal o valor de confiança é decrementado por m e para cada transmissão bem sucedida é incrementado por $\alpha \times m$, onde $0 < \alpha \leq 1$. O valor de trust, $T[SN_i]$, após n transmissões é definido por: $T[SN_i]^n = T[SN_i]^{n-1} - Num_{fail} \times m + Num_{succ} \times m$, onde Num_{succ} representa o número de transmissões realizadas com sucesso e Num_{fail} o número de transmissões que fracassaram. Se o valor de $T[SN_i]$, atingir um valor limiar mínimo, o nó é marcado como não confiável.

O *MDS – s* localizado na camada de aplicação foi criado para poder compensar o controle de fluxo da camada MAC. Assim, a camada de transporte pode realizar o encaminhamento de pacotes por rotas menos congestionadas e a camada de aplicação pode ter uma visão global do estado da rede atual para chamar o controle de fluxo, por exemplo: pausar o tráfego de saída ou negar a entrada de tráfego.

Descrevemos neste capítulo os principais tipos de mau comportamento que podem ocorrer em cada camada, além de alguns dos sistemas de detecção existentes que tratam do mal comportamento na camada MAC e que utilizam a idéia de *cross-layer* para melhorar o desempenho da rede. Fizemos algumas críticas ao DOMINO que também foi utilizado como comparação em relação ao nosso trabalho realizado na camada MAC.

Capítulo 4

Sistemas de Reputação

Este capítulo irá descrever alguns dos sistemas de reputação existentes na literatura. Mas antes de detalhar cada um dos sistemas de reputação, é necessário definir alguns conceitos que serão descritos logo na primeira seção deste capítulo.

4.1 Definições

4.1.1 Reputação e Confiança

Os conceitos de confiança e reputação estão bastante relacionados, e por isso muitas vezes são confundidos como sinônimos. É importante distinguir o significado de ambos. Embora a idéia de confiança e de reputação sejam fáceis de compreender, pois são palavras cujo significado podem ser aplicados no cotidiano, é complexo formular uma definição para cada um destes conceitos [23].

A reputação pode ser definida como a percepção que um nó tem sobre o desempenho de outro nó ao realizar alguma operação na rede [37]. O valor de reputação é utilizado como uma predição da qualidade de serviço. A confiança, por sua vez, é interpretada como uma relação entre entidades que participam de vários protocolos [35]. A medida de confiança está relacionada com a interação anterior entre os nós. Segundo Sonja Buchegger *et. al.* [6], a medida de reputação representa o quanto um nó se comporta bem, já a medida de confiança representa o quanto honesto um nó é. Então, o valor de reputação é utilizado para decidir se um nó é mal comportado enquanto o valor de confiança é utilizado para decidir se um nó é confiável. A maioria dos sistemas de reputação utiliza o valor de reputação como métrica para a confiança [28, 5].

4.1.2 Transitividade de Confiança

Na vida real o conceito de confiança muitas vezes pode ser considerado transitivo. Por exemplo na figura 4.1, Ana confia em Bruno que confia em Carla. Dependendo da situação e do grau de confiança que Ana têm em relação a Bruno, Ana pode ter uma confiança derivada da confiança que Bruno tem por Carla [23].

Mas nem sempre a relação de transitividade pode ser aplicada. Caso Ana confie em Bruno para indicar um bom advogado, e Bruno confia em Carla para

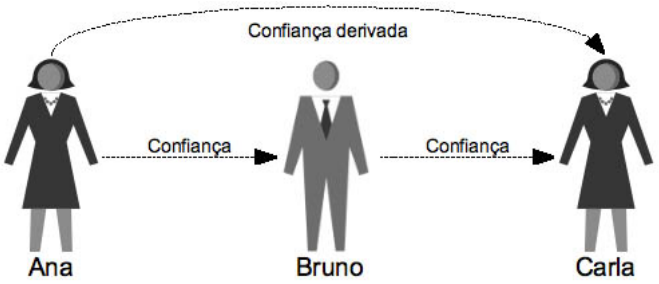

Figura 4.1: Confiança transitiva derivada

fazer faxina em seu lar, isto não significa que Ana confiará em Carla para fazer uma faxina, pois a relação de confiança entre as partes está relacionada a fatos diferentes. Entretanto, a transitividade poderia ser aplicada se todas as relações de confiança fossem para indicar um advogado.

A transitividade de confiança pode ter mais pessoas envolvidas. Todavia, quanto maior o número de pessoas para gerar a confiança derivada, maior será a incerteza quanto a esta confiança. Exemplificando, na figura 4.2, suponha que Ana confia em Bruno, Bruno confia em Carla que confia em Davi. Mas Ana não confia em Eduardo, enquanto Davi confia. A confiança derivada de Davi para Ana indicaria que Eduardo é confiável, entretanto, Ana não confia em Davi.

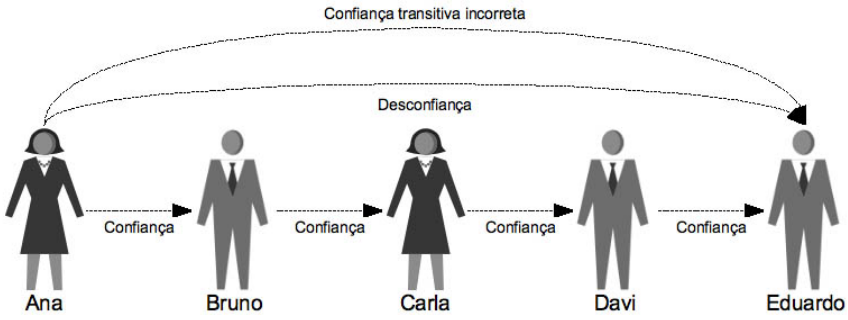

Figura 4.2: Confiança transitiva incorreta

Observe agora um outro caso na figura 4.3, Ana confia em Bruno e confia em Carla. Bruno e Carla confiam em Davi. Neste caso duas pessoas poderiam indicar Davi para Ana e a incerteza quanto a confiança que Ana poderia ter em relação a Davi é menor caso a indicação fosse, apenas de uma pessoa.

O conceito de transitividade deve ser aplicado com cautela em redes ad hoc, uma vez que existem restrições e nem sempre as relações entre os nós são iguais as relações entre as pessoas.

4.1.3 Reputação Direta e Indireta

Segundo Jiangyi Hu [19] a forma de se obter uma informação para calcular reputação pode ser classificada como direta e indireta. A reputação direta é baseada nas observações que o próprio nó faz de seus vizinhos. A reputação indireta

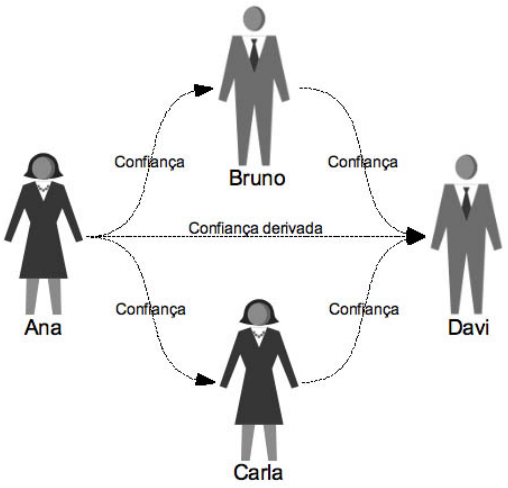

Figura 4.3: Combinação de confiança transitiva

refere-se a informação proveniente por meio de um terceiro nó sobre outro nó. Essa informação pode ser sob a forma de uma lista negra, lista de amigos, tabela de reputação. Além disso, essas informações podem ser dados de algum evento ocorrido em alguma sub-área da rede repassada nó a nó, até chegar ao nó que deseja obter alguma informação sobre aquele nó. Ao invés dos dados, as informações trocadas entres os nós podem ser apenas avisos como uma forma de alertar a existência de nós mal comportados naquela sub-área.

4.1.4 Reputação Global e Local

Um sistema de reputação, segundo Jiangyi Hu [19] pode ser classificado como global ou local. Em um sistema de reputação global os nós utilizam os dados trocados entre os outros nós da rede para calcular a reputação de cada nó, a reputação desses nós é conhecida como reputação indireta. Para isso, um nó precisa ter alguns dados dos nós conhecidos, e no pior caso precisará conhecer a reputação de **todos** os nós da rede, por isso a origem do nome global. Essa abordagem possui alguns problemas que podem prejudicar o sistema de reputação com medições erradas de reputação, acusações falsas e elogios falsos. Deve-se lembrar que os valores de reputação podem ser trocados, mas devem ser evitados e substituídos pela troca de dados brutos dos eventos observados para dificultar a fabricação e modificação de valores, mas mesmo assim os sistemas não estão livres de acusações falsas e elogios excessivos.

Um outro problema, é a quantidade de informação que deverá ser armazenada em cada nó além dos valores de reputação e confiança associados aos nós. Existe também, um outro problema relativo ao custo de troca de informações indiretas entre os nós, que, segundo Jinagyi Hu, o período de disseminação de informações é da ordem $O(N^2)$. Os sistemas globais, geralmente possuem um valor de confiança associado aos nós, que irá garantir se a informação recebida é realmente confiável. Essa verificação implica em computação adicional. Nos sistemas de reputação local as informações de terceiros não são consideradas, é levado em conta apenas as observações locais dos nós vizinhos, e o cálculo de reputação também é feito

localmente para cada um dos seus nós vizinhos.

4.1.5 Segurança e Confiança

Os termos segurança e confiança estão interligados. Por exemplo, o uso de criptografia e a distribuição de chaves entre duas partes, necessita que as partes tenham confiança uma na outra. Semelhantemente, duas partes que confiam uma na outra dependem da criptografia para realizar a troca de chaves em um canal inseguro.

O propósito da segurança é proteger alguém contra agentes maliciosos. Os mecanismos tradicionais de segurança geralmente protegem usuários de outras partes maliciosas restringindo o acesso ou utilizando-se a criptografia. A criptografia oferece várias formas de proteção, entretanto existem casos onde somente a criptografia não é suficiente. Existem situações onde o usuário precisa se proteger contra informações e serviços que não oferecem aquilo que foi proposto. Por exemplo, um nó em uma rede peer-to-peer que diz ter determinada música, entretanto o arquivo corresponde a outra canção [23].

Em redes ad hoc, só existe uma relação prévia de confiança entre os nós de uma rede em alguns cenários específicos, como as redes militares, onde uma autoridade confiável gerencia e pode-se acrescentar um mecanismo forte de autenticação no hardware dos dispositivos. Em redes abertas, onde não existem hardwares imutáveis nem uma infra-estrutura forte de autenticação, a confiabilidade entre os nós se perde. Com a falta de confiança prévia entres os nós da rede, a forma mais eficiente de lidar com os nós egoístas é por meio de mecanismos que incentivem a cooperação entre os nós [4].

4.2 Aplicações

Existem várias áreas da computação onde é necessário estabelecer confiança entre as partes que se interagem no sistema, dessa forma, alguns mecanismos chamados de sistemas de reputação e confiança foram propostos. Nas seções abaixo, algumas aplicações desse sistema são descritas.

4.2.1 Sites de leilão eletrônico

Sites de leilão eletrônico, como o *Ebay* e o Mercado Livre, não são responsáveis pela venda dos produtos mas hospedam anúncios de terceiros, são os membros da comunidade que vendem seus produtos. Alguns destes vendedores possuem intenções maliciosas e vendem produtos com descrições diferente da anunciada, ou após receberem o dinheiro não enviam o produto para o comprador. Os vendedores também precisam ter cuidado com os compradores, é possível que compradores não sejam honestos e enviem um cheque sem fundo ou ainda desistam da compra, após fazerem um lance no site.

Para tentar evitar parte desses problemas os sites *Ebay* e Mercado Livre implementam um sistema de reputação bastante simples. Embora sejam sites diferentes, o sistema de reputação de ambos é bastante semelhante. Ao final de cada

transação as partes podem classificá-la como positiva, neutra, ou negativa e podem também escrever um comentário sobre a outra parte. A pontuação de cada usuário será a soma dos pontos referentes à classificação de cada usuário com a qual a transação foi realizada, onde um valor positivo soma um ponto, negativo diminui um ponto e neutro não altera a pontuação existente daquela parte.

Este sistema, porém, possui algumas falhas. Um usuário que possua 100 transações classificadas como positiva e 10 como negativa possui a mesma reputação que um usuário com 90 transações positivas e nenhuma negativa. Além disso foi observado que geralmente as classificações nesse sites de leilão são positivas, e chegou-se a conclusão que existe uma reciprocidade entre os vendedores e compradores, assim sugere-se que apenas uma das partes classifique a transação [22]. Um outro problema ocorre quando um vendedor com uma pontuação muito alta aproveita-se da sua boa reputação para praticar atitudes maliciosas. Esse comportamento, porém, não irá interferir, a curto prazo, no seu valor de reputação.

Embora esse sistema de reputação seja primitivo, para o contexto tanto do Mercado Livre e do *Ebay* ele têm-se mostrado eficiente e gerado um efeito positivo nas comunidades.

4.2.2 Redes *Peer-to-Peer* (P2P)

Redes *peer-to-peer* consistem de nós que agem simultaneamente como servidores e clientes. Esse tipo de rede é muito utilizada por usuários que compartilham arquivos e conseguem carregar estes em sua máquina com rapidez. O número de usuários das redes P2P cresceu bastante e as mais conhecidas são o *Gnutella, KaZaA, iMesh, Morpheus, Torrent* e muitas outras.

O funcionamento básico destas redes pode ser visualizado na figura 4.4. O nó A deseja carregar o arquivo X. Os nós B e C possuem este arquivo. O nó A irá baixar uma parte do arquivo do nó B e outra do C. Antes que a transferência do arquivo termine para o *peer* A, o nó D também solicita o arquivo X, assim A envia parte de seu arquivo para D, enquanto D também se conecta com B e C.

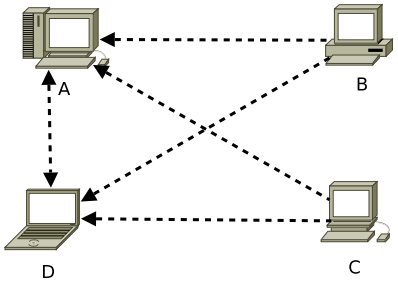

Figura 4.4: Representação do funcionamento das redes P2P

Os sistemas de reputação são bastante úteis nessas redes. Entretanto, alguns problemas podem ocorrer: alguns *peers* podem afirmar possuir um determinado arquivo, mas na verdade o arquivo corresponde a outro que não têm relação com o que foi descrito; os *peers* podem ser egoístas, ou seja, fazem *download* dos arquivos mas não fazem *uploads*, o que prejudica o funcionamento da rede. Além

destes problemas, os sistema de reputação e confiança das redes P2P devem ser robustos, o suficiente para evitar ataques dentro do próprio sistema, como por exemplo, a alteração indevida dos valores de reputação.

4.2.3 Redes Ad Hoc

As redes ad hoc dependem da cooperação entre os nós, e é muito importante que seja implementado um protocolo que garanta e incentive a cooperação entre os nós. A cooperação em redes ad hoc é necessária para o roteamento de pacotes entre nós que não sejam vizinhos próximos. Todavia, como descrito em seções anteriores, alguns nós podem ter um comportamento egoísta com finalidade de economizar bateria. Dessa forma, o desempenho da rede pode ser severamente prejudicado pelo mau comportamento dos nós. Vários protocolos foram propostos [5, 18, 28, 3]. Os problemas que estes protocolos precisam resolver são similares aos problemas enfrentados pelas redes P2P.

Os sistemas de reputação e confiança em redes ad hoc são compostos basicamente por quatro componentes os quais estão representados e descritos na figura 4.5.

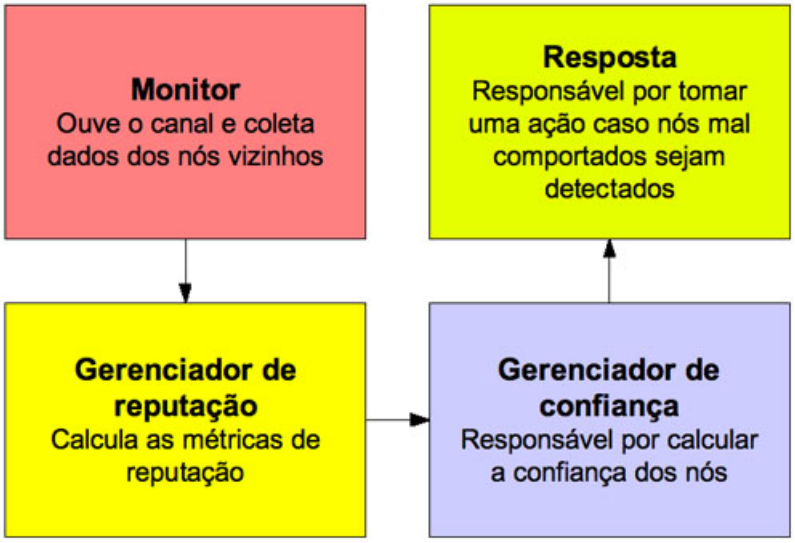

Figura 4.5: Diagrama que representa os componentes básicos presentes na maioria dos sistemas de reputação e confiança

4.3 Desafios em Sistemas de Reputação

Um bom sistema de reputação precisa ser robusto o suficiente para impedir que nós maliciosos se aproveitem de brechas presentes no sistema e as explorem. É preciso que os nós possuam identidade, e consequentemente as três propriedades associadas: persistência, unicidade e distinção. Além destas características, deve exisitir uma forma de identificar e impedir que os nós alterem com facilidade sua identidade, para evitar que nós que possuam baixa reputação mudem de

identidade, passando-se por um nó novo que acabou de entrar na rede, ou roube a identidade de outro nó presente na rede. É possível que exista um comportamento anômolo, onde um nó teria múltiplas identidades, conhecido como ataque de *Sybil* [7].

Outra questão que precisa ser evitada são as propagações de valores de reputação falsos, sejam eles negativos ou excessivamente positivos. O nós que apresentam um testemunho divergente dos demais nós da rede devem ser tratados com cautela, porque eventualmente o nó mais próximo do nó mal comportado irá testemunhar primeiro a ocorrência do evento, e o valor de reputação do nó supostamente mal comportado será distinto daqueles registrados na tabela de reputação dos demais nós, ou seja, o desvio do valor de reputação de um único nó, não significa que o nó seja mentiroso e deva ser punido. Entretanto não é trivial identificar inicialmente se uma informação é verdadeira ou falsa. Mas mesmo assim, é preciso que os nós mentirosos sejam identificados e punidos de alguma forma.

Nem sempre a falta de cooperação do nó é resultado de um comportamento egoísta ou malicioso. O nó pode estar em uma localização geográfica desfavorável ou os recursos de bateria podem estar acabando. Identificar o motivo do mau comportamento do nó, entretanto, é complexo. É, portanto, necessário que o sistema de reputação forneça mecanismos de redenção, ou seja, fornecer novas chances para que um nó com baixa reputação possa se recuperar e usufruir dos recursos da rede. É importante que o mecanismo de redenção consiga identificar a recorrência do mau comportamento e dificulte cada vez mais a recuperação dos valores de reputação, para evitar que um nó cause inúmeros prejuízos na rede, mas sempre retorne a ela por meio do mecanismo de redenção.

4.4 Trabalhos Relacionados

O encaminhamento de mensagens incorre em custos (bateria, processamento) para o nó e como os recursos são escassos faz-se necessário o uso de incentivos para que o nó possa encaminhar as mensagens pertencentes a um outro nó. Existem dois tipos de mecanismos para estimular a cooperação entre os nós: aqueles baseados em 1) incentivos econômicos e em 2) reputação.

4.4.1 Incentivos Econômicos

Os sistemas de Incentivos Econômicos usam uma espécie de crédito ou micro pagamento para compensar o serviço do nó, ou seja, o nó recebe um pagamento virtual para encaminhar as mensagens dos outros nós. *Nuglets* [9] e *Sprite* [38] são exemplos desse tipo de sistema.

L. Butty *et. al.* [9] criaram uma moeda virtual chamada *nuglets* e propuseram um mecanismo de recompenssa pelos serviços prestados que estimula a cooperação dos nós nas redes ad hoc. Existem dois modelos para o uso do *nuglets*: bolsa de pacotes e comércio de pacotes. No primeiro modelo, ao enviar o pacote o nó origem carrega-o com a quantidade necessária de *nuglets* e cada nó intermediário retira alguns *nuglets* pelo encaminhamento do pacote. Já no segundo modelo, os

pacotes são negociados pelos nós intermediários. Cada nó intermediário compra o pacote do nó anterior e o vende para o próximo nó da rota. Dessa forma, os nós intermediários lucram com o encaminhamento de pacotes e o nó destino paga pelo serviço obtido.

Zhong *et. al.* [38] propuseram um sistema de crédito. Uma entidade central denominada Serviço de Declaração de Crédito (SCC) é responsável por determinar o crédito e o débito de cada nó envolvido na transmissão de uma mensagem. Ao receber uma mensagem cada nó mantém um recibo que depois será reportado ao SCC e assim, o nó será recompensado pelo serviço. O crédito envolvido na transação depende do êxito da operação. Se o nó seguinte da rota reportar um recibo válido ao SCC, significa que o encaminhamento da mensagem foi realizado com sucesso. O modelo que estabelece o pagamento e os créditos é baseado na Teoria de Jogos.

O problema do sistema de moedas virtuais é a dependência de um hardware imutável (no caso do *Nuglets*) ou ainda a necessidade de um servidor central para determinar o débito e o crédito de cada nó envolvido na transmissão da mensagem (no caso do *Sprite*). As duas abordagens, entretanto, não se enquadram em ambientes puramente ad hoc.

4.4.2 Protocolos de Reputação e Confiança

Os sistemas de Reputação são mecanismos que têm como objetivo evitar a não-cooperação dos nós por meio do monitoramento dos nós vizinhos.

4.4.2.1 CORE

O protocolo CORE *(Collaborative Reputation Mechanism)* proposto por Michiardi *et. al.* [28] força a cooperação entre os nós em uma rede ad hoc. A idéia principal do protocolo é fazer com que os nós não obtenham vantagens sobre outros nós e apresentem um comportamento egoísta, pois esse comportamento implicaria em negação de serviço. O protocolo também evita outros tipo de ataque, chamados de acusações falsas, que ocorrem quando um nó malicioso envia em broadcast uma reputação negativa falsa sobre um outro nó. O funcionamento básico do protocolo baseia-se em um mecanismo de monitoração do canal e uma tabela de reputação presente em cada nó da rede.

O CORE define três tipos de reputação: reputação subjetiva, funcional e indireta. A reputação subjetiva corresponde às observações realizadas diretamente pelo nó. Essa reputação é calculada de tal forma que as observações mais antigas possuem maior relevância. A reputação indireta é baseada em informações provenientes de nós terceiros. A reputação funcional é utilizada para calcular o valor final da reputação, ela se baseia nos diferentes critérios ao qual a reputação foi observada, como por exemplo, se o nó encaminhou pacotes e como participou do protocolo de roteamento, levando também em consideração os valores da reputação subjetiva e indireta. Os nós compartilham os valores da reputação com os seus vizinhos, entretanto, se a reputação for negativa ela não será compartilhada.

Os valores de reputação calculados não são constantes. Se o valor for positivo, com o tempo será decrementado até chegar a um valor neutro. A razão disto é

evitar um possível ataque, caso um nó tenha uma reputação muito alta poderá entrar em estado ocioso quando não precisar mais se comunicar e, assim não encaminhará os pacotes de outros nós. A reputação de um nó está diretamente relacionada ao seu comportamento e a cooperação com os outros dispositivos. Se o valor de reputação final for negativo o nó será classificado como mal comportado e qualquer serviço que o nó venha a solicitar será negado. Caso o valor seja positivo o nó será classificado como confiável. Nesse sistema não há vantagens em ser mal comportado, pois qualquer recurso da rede será negado.

4.4.2.2 SORI

O SORI *(Secure and Objective Reputation-based Incentive Scheme)* proposto por Q. He *et. al.* [18] busca solucionar o problema de confiança incentivando a cooperação dos nós por meio do encaminhamento de pacotes de forma segura e objetiva. Para isso, a reputação do nó é calculada por meio de medidas objetivas. O SORI é um sistema de reputação global e para garantir que o valor da reputação de nós terceiros não seja interceptado por outros nós, essa informação será cifrada por um sistema de autenticação baseado em uma função hash. Além disso, a reputação de um nó terceiro não é transmitido por todos os nós da rede, apenas para os vizinhos do nó ao qual a reputação foi calculada.

Para calcular a reputação de um nó, os nós monitoram seus vizinhos e observam seu comportamento ao encaminhar os pacotes. Os nós observam o número de pedidos de encaminhamento de pacotes que foram feitos para seus vizinhos e verificam quantos pacotes foram realmente encaminhados. O valor da reputação é a razão entre estes dois números. A confiança, por sua vez, é proporcional ao número de pacotes que foram pedidos para serem encaminhados. Cada nó faz uma avaliação global dos outros nós, ou seja, para calcular a confiança final os valores de reputação e confiança de seus vizinhos são utilizados, levando em consideração a confiança que o nó têm de seus vizinhos.

A decisão de como tratar um nó é baseada na avaliação global, caso esse valor seja menor que um valor previamente definido, o nó y não irá encaminhar os pacotes do nó x por uma probabilidade n relacionada à avaliação global. Esta abordagem é utilizada, pois o não encaminhamento de um pacote pode ocorrer por outras razões diferentes de egoísmo, como falhas do sistema ou colisão, por isso o nó avaliado poderá eventualmente recuperar sua reputação e voltar a participar da rede.

4.4.2.3 CONFIDANT

S. Buchegger *et. al.* [5] propõe um protocolo denominado CONFIDANT *(Co-operation Of Nodes: Fainess In Dynamic Ad hoc Network)* com o objetivo de detectar e isolar os nós mal comportados. No CONFIDANT, cada nó é composto por quatro componentes que se interagem: monitor, gerenciador de confiança, gerenciador de reputação e gerenciador de rotas.

O monitor identifica os desvios do nó seguinte por meio de observações do canal de transmissão. Assim que um desvio é encontrado o gerenciador de reputação é chamado. O gerenciador de confiança envia uma mensagem de alerta para avisar

um nó da existência de um nó malicioso. A detecção do nó pode ser realizada se o nó viveu uma experiência com o nó malicioso ou recebeu uma alerta de outro nó. Os receptores da mensagem são chamados de amigos. A cada alerta recebido é feito um filtro para verificar o nível de confiança da mensagem de alerta. Os valores de confiança são importantes para (1) prover ou aceitar informações de roteamento; (2) aceitar um nó como parte da rota de um pacote; (3) ser aceito em uma rota destinado a um outro nó qualquer. Cada nó gera a sua própria lista de valores de reputação dos outros nós que eventualmente poderá ser trocada com os vizinhos. Um valor só é alterado quando há evidência suficientes de mau comportamento. O gerenciador de rotas é responsável por decidir uma ação ao detectar a presença de um nó malicioso, uma dessas ações seria, por exemplo, a remoção do nó malicioso do caminho

Os nós monitoram a vizinhança e reportam a ocorrência de eventos suspeitos para o sistema de reputação. Mantém-se um controle dos eventos significantes para distinguir o comportamento malicioso de simples coincidências, realizado a partir da definição de um valor mínimo, que deve ser superior o bastante para tornar possível a distinção dos dois comportamentos. Esse valor varia de acordo com os requisitos de segurança de cada nó. Quando esse valor for ultrapassado, a reputação do nó é alterada. E se o valor de reputação atingir um certo limite o gerenciador de rotas é acionado, as ações desse componente variam e pode, por exemplo, resultar na remoção de todas as rotas contidas no cache que contenham o nó malicioso.

Além do valor de reputação, cada nó mantém um valor de confiança associado que representa o quanto o nó é confiável. No CONFIDANT, os nós cooperam entre si com a troca de informações. Essas informações são os dados coletados por cada nó, resultantes da observação dos nós vizinhos, nenhum valor de reputação ou confiança são publicados, mas são alterados com o recebimento daqueles dados. O CONFIDANT implementa um mecanismo de redenção, onde os valores sofrem um decaimento ao longo do tempo, dessa forma um nó não poderá aproveitar o seu bom comportamento anterior para ludibriar outros nós, e os nós que tenham sido injustiçados poderão voltar a participar ativamente na rede. Somente os valores de reputação compatíveis são considerados e só exercem uma influência segundo um peso w, esse peso será maior para a experiência de próprio nó com o nó mal comportado ou por observação própria.

O cálculo da reputação e confiança de cada nó é realizado por meio de uma abordagem probabilística baseado nos sistemas Bayesianos. Os cálculos não foram descritos em detalhes, pois a breve descrição dos componentes e a relação entre eles é suficiente para compreender o funcionamento desse protocolo, as fórmulas e os cálculos podem ser verificados em [5, 7, 6].

A tabela 4.1 mostra as principais diferenças entre os protocolos de reputação apresentados em relação a: (1) representação da informação e classificação, isto é, quais tipos de dados são coletados e como é feito o cálculo de reputação; (2) uso de informação de terceiros, ou seja, se as reputações são trocadas ente os nós vizinhos; (3) confiança, ou seja, se os protocolos utilizam confiança; (4) perdão e resposta, se existe algum mecanismo de redenção para que os nós classificados como maliciosos possam voltar a participar da rede se melhorarem seu comportamento.

Neste capítulo, descrevemos algumas aplicações dos sistemas de reputação e

Tabela 4.1: *Comparação entre os Sistemas de Reputação*

	Sori	**Core**	**Confidant**
Representação de informação e classificação	Mantêm contadores para o número de pacotes solicitados para encaminhamento e os que realmente foram encaminhados. O sistema de reputação usa esses dados junto com outras medidas de confiança.	Representa os valores de reputação em um tabela de reputação. A tabela captura reputação subjetiva, indireta e funcional separadamente.	Usa uma aproximação Bayesiana onde a confiança de um nó i para um nó j é atualizada a cada observação para estimar a probabilidade de mal comportamento. Nós são classificados como mal comportados quando o valor passa do aceitável.
Uso de informação de terceiros.	Usa informações de terceiros localmente e as considera usando uma medida de credibilidade.	Permite apenas informação de terceiros que seja positiva. Que protege de acusações suspeitas mas não de pontuações positivas suspeitas.	Usa uma combinação de dois mecanismos para detectar informações suspeitas. Primeiro, apenas informação compatível é considerada. Segundo a informação só pode influenciar a avaliação de credibilidade um pouco baseado num peso w.
Confiança.	Baseia a confiança em informações de terceiros baseado no comportamento de encaminhamento.	Não possui noção de confiança.	Usa confiança adaptativa sem avaliações pré-determinadas, onde a confiança é baseada na honestidade do nó.
Perdão e resposta secundária	Não possui perdão e resposta secundária.	Um nó recebe perdão se ele se comporta bem de novo. Entretanto, não pode provar ser bem comportado, quando estiver isolado. Nesse caso, ele continuará isolado.	Desconta todas as avaliações periodicamente e desta forma fornece uma forma de redenção.

várias definições importantes relativas à reputação e ao estabelecimento de confiança entre os nós. Além disso, descrevemos os principais sistemas de reputação que minimizam o efeito de nós maliciosos na camada de rede.

Capítulo 5

Modelo Proposto

Neste capítulo iremos descrever o objetivo do nosso trabalho, o modelo de sistema de reputação proposto e o modelo criado para simulação e testes.

5.1 Descrição dos Objetivos

Vimos no capítulo 3 que existem várias técnicas possíveis para obter vantagem do canal de transmissão em relação aos demais nós da rede. Entretanto, a maioria dos sistemas de reputação trata apenas do mau comportamento na camada de rede. Poucos trabalhos lidam com o mau comportamento na camada MAC. É possível criar um sistema de reputação robusto que integre as melhores técnicas de identificação de mau comportamento nas diversas camadas com o uso de *cross-layer*. Neste trabalho fazemos um proposta de sistema de reputação *cross-layer*.

O foco do nosso trabalhom entretanto, foi verificar o desempenho de uma rede ad hoc com a presença de um nó malicioso na camada de enlace. Supondo que os nós utilizam o padrão 802.11 para gerar o número aleatório de *backoff* dentro do mesmo intervalo de $[0, CW_{min}]$, podemos esperar que a probabilidade de acessar o canal de transmissão será a mesma para todos os nós da rede. Mas um nó malicioso, pode utilizar um valor máximo de *backoff* (CW_{min}) menor que os demais. Dessa forma, a probabilidade de obter um número menor de *backoff* e conseqüentemente ter acesso ao canal de transmissão, será muito maior. Entretanto, se observarmos a taxa de transmissão dos nó por um determinado período, é possível determinar a porcentagem de transmissões de cada nó. Assim torna-se possível verificar quanto o nó malicioso pode ganhar em termos de uso do canal, e também, quando um nó não malicioso é capaz de perceber a presença de um nó egoísta na rede.

Construímos um modelo de simulação para avaliar a partir de que momento o nó normal consegue identificar um nó malicioso e quanto um nó malicioso pode ganhar em relação a vazão utilizando um intervalo de *backoff* menor que os demais nós da rede.

5.2 Proposta: Sistema de Reputação Cross-layer

Nesta seção iremos propor um modelo de sistema de reputação com interação entre camadas.

A arquitetura deste modelo está dividida da seguinte forma:

- **Módulo Monitor.** Responsável por monitorar o canal e coletar os dados dos nós vizinhos.
- **Módulos de Reputação.** Os dados coletados pelo módulo monitor são repassados para quatro módulos distintos: Reputação da Camada de Enlace, Reputação da Camada de Rede, Reputação da Camada de Transporte, Reputação da Camada de Aplicação. Cada um desses módulos é responsável por calcular um valor de reputação diferente, que irão compor o índice geral de reputação, calculado pelo Gerenciador de Reputação.
- **Gerenciador de Reputação.** O gerenciador de reputação recebe os índices de reputação e calcula o índice geral de reputação. Esse valor será armazenado na tabela de confiança.
- **Gerenciador de Confiança.** Responsável por receber alertas de nós maliciosos e enviá-los. Além disso, também é responsável por trocar listas de reputação com outros nós.
- **Módulo de Reação.** Este módulo é responsável por tomar alguma ação quando um nó malicioso é identificado. Várias ações podem ser tomadas, como não encaminhar os pacotes provenientes do nó malicioso, ou procurar rotas outras rotas que não passem pelo nó malicioso, entre outras.

A figura 5.1 ilustra a relação entre os componentes deste modelo.

Cada nó monitora o canal e observa as informações que contemplam as camadas de transporte, de rede e MAC. As informações da camada de aplicação podem ser coletadas caso esteja sendo executada a mesma aplicação em todos os nós da rede, como em uma rede sensores. O nó monitora o canal por um período pré-determinado, que irá depender do número de nós presentes na rede. Uma vez coletadas essas informações, o módulo monitor repassa os dados coletados para seus respectivos módulos de reputação, ou seja, os dados coletados relativos à camada MAC são repassados para o módulo de reputação da camada MAC e assim por diante. Os módulos de reputação calculam o índice de reputação parcial e repassam para o módulo Gerenciador de Reputação que irá calcular o índice geral de reputação baseado na média harmônica dos índices.

$$I_G = \frac{4}{\frac{1}{I_E} + \frac{1}{I_R} + \frac{1}{I_T} + \frac{1}{I_A}} \tag{5.1}$$

A fórmula 5.1 mostra como é calculado o índice geral de reputação (I_G), onde I_E é o índice de reputação da camada de enlace, I_R é o índice de reputação da camada de rede, I_T é o índice de reputação da camada de transporte e I_A é o índice de reputação da camada de aplicação. Caso não seja usada reputação na camada de aplicação o numerador da fórmula muda para 3.

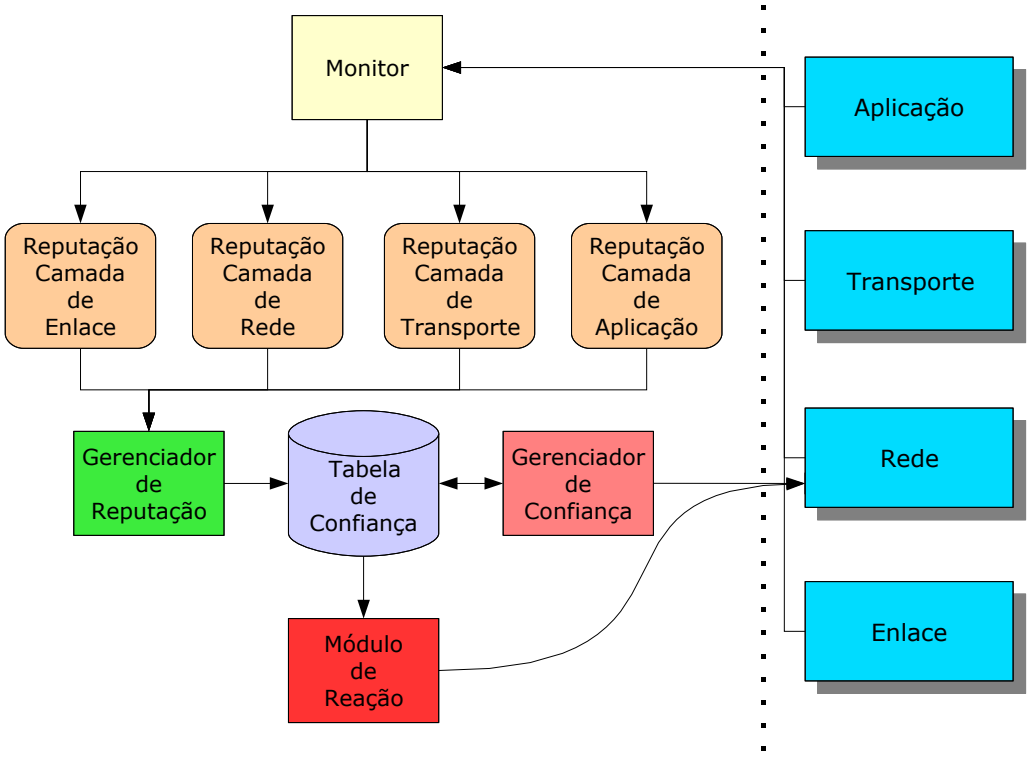

Figura 5.1: Componentes do Sistema de Reputação entre camadas.

Os índices gerais de reputação são repassados para o módulo Gerenciador de Confiança que irá atualizar o nível de confiança do nó. Os índices de reputação são baseados na própria observação do nó, ou seja, ele observa o canal e tira suas próprias conclusões em relação ao nó. O nível de confiança pode ser alterado por meio da troca de listas de reputação entre os nós. Quando o nó identifica algum nó como malicioso, o Gerenciador de Confiança repassa essa informação para os amigos, que podem ser os nós que possuem nível alto de confiança. Ao receber algum alerta de nó malicioso na rede, primeiro o nó verifica o nível de confiança do nó remetente do alerta. E baseado nesse nível de confiança poderá ou não atualizar o nível de confiança do nó supostamente indicado como malicioso. É necessário ponderar o alerta de mau comportamento baseado no nível de confiança, para evitar que nós maliciosos propaguem informações erradas de outros nós com intenção de prejudicá-los. Por exemplo, um nó malicioso, poderia propagar informações erradas do nó vizinho para diminuir sua reputação e evitar que ele participe de alguma rota. Com isso, o nó malicioso poderia diminuir o número de pacotes encaminhados para ele, e reservar seus recursos de bateria.

Além disso, os nós podem trocar listas de reputação de tempos em tempos com seus amigos e atualizarem o nível de reputação dos nós. Os dados trocados são sempre baseados na observação direta, ou seja, não são propagados dados obtidos por terceiros. As informações trocadas afetam o nível de confiança, mas os índices gerais de reputação obtidos da observação direta terão sempre maior peso, para minimizar a troca de informações falsas. Mesmo que as listas de reputação sejam trocadas entre os nós amigos, ainda existe a possibilidade de que informações

falsas sejam propagadas caso existam nós bizantinos na rede. Os nós bizantinos se comportam bem por um período de tempo para conquistar um nível de confiança alto e depois passam a atuar maliciosamente. Mas, caso o nível de confiança obtido seja muito alto, o decaimento será devagar. Para minimizar o efeito desse comportamento, utilizamos a técnica de *aging* nos índices de reputação.

Quando um nó for identificado como malicioso, o módulo de reação será chamado. Os principais tipos de reação a ser tomados são:

- Não encaminhar os pacotes provenientes do nó malicioso
- Procurar outras rotas que não envolvam o nó malicioso.

A troca de informações é importante para permitir que os nós possam ter uma visão global da rede e procurar rotas mais seguras. Entretanto, as trocas de informações podem gerar falsos positivos. É importante que haja um mecanismo de redenção, para permitir que os nós classificados de forma errônea possam voltar a participar da rede. Além disso, os nós que estejam presentes em áreas instáveis, com muita interferência, podem ser classificados de forma errônea como maliciosos. O mecanismo de redenção é importante para permitir que esse nós possam voltar a participar da rede, caso mudem de lugar ou a interferência do sinal diminua.

5.3 Definições Matemáticas e Estatísticas

Nesta seção, encontra-se termos matemáticos e estatísticos que serão utilizadas nas próximas seções.

Definição 5.3.1. (Grafo) *Seja G um grafo, V um conjunto finito e não vazio e A uma relação binária em V. O conjunto V é chamado de conjunto de vértices do grafo G, e o conjunto A é chamado de conjunto das arestas de G [11].*

Existem dois tipos de grafos: direcionados e não direcionados [11]. No grafo não direcionado G, o conjunto de arestas A é constituído por pares de vértices **não** ordenados. Enquanto, nos grafos direcionados os pares são ordenados. Logo, uma aresta é um conjunto $\{u, v\}$, onde $\{u, v\} \in V$ e $u \neq v$.

Definição 5.3.2. (Grafo Completo) *Um grafo completo é um grafo não direcionado, onde todos os vértices são adjacentes, isto é, cada vértice está conectado a todos os outros vértices. [11]. O grafo completo de n vértices será representado por K_n.*

A figura 5.2 ilustra um grafo completo K_6 com seis vértices. Nas redes ad hoc, cada vértice do grafo K_6 representa um nó da rede, onde todos os nós podem comunicar-se entre sim de forma direta, ou seja, com apenas um salto de distância.

Definição 5.3.3. (Espaço Amostral) *Para cada experimento ε, o espaço amostral Ω consiste de uma enumeração (finita ou infinita) de todos os resultados possíveis do experimento ε, ou seja, $\Omega = \{\omega_1, \omega_2, ..., \omega_n, ...\}$. [12].*

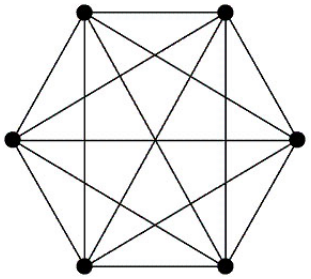

Figura 5.2: Grafo completo K_6

Suponha um experimento ε_1 que consista em jogar uma moeda quatro vezes e observar o número de caras obtido. O espaço amostral Ω_1 para esse experimento é $\Omega_1 : \{0, 1, 2, 3, 4\}$.

Definição 5.3.4. (Variável Aleatória) *Seja ε um experimento e Ω um espaço amostral associado ao experimento. Uma função X, que associe cada elemento $\omega \in \Omega$ um número real, $X(\omega)$, é denominada variável aleatória [12].*

Por exemplo, suponha que atiremos duas moedas. O espaço amostral associado ao experimento é dado por $\Omega_1 = \{HH, HT, TH, TT\}$. Podemos definir a variável aleatória da seguinte forma: X é o número de caras obtidas nas duas moedas. Assim, $X(HH) = 2$, $X(HT) = X(TH) = 1$ e $X(TT) = 0$.

Definição 5.3.5. (Variável Aleatória Contínua) *Uma função X, definida sobre o espaço amostral Ω é dita uma variável aleatória contínua, se X puder tomar todos os valores em algum intervalo de número reais (a, b). [12].*

Definição 5.3.6. (Esperança) *Esperança, ou valor média de uma variável aleatória contínua, é definida como:*

$$E(X) = \int_a^b xf(x)dx, \tag{5.2}$$

onde $a \leq x \leq b$ e $f(x)$ é a função de densidade de x [12].

Se X for igual a uma constante real b, teremos que $f(x) = 1$ e $x = b$, logo $E(x) = b$. Se quisermos calcular o valor esperado de uma função $g(X) = X$, com $0 \leq x \leq 1$ e $f(x) = 1$ teremos que $E(g(X)) = \int_0^1 g(x)f(x)dx = \int_0^1 xdx = \frac{1}{2}$.

Definição 5.3.7. (Distribuição Binomial) *Distribuição binomial é a distribuição de probabilidade discreta do número de sucessos em uma série de n eventos independentes, onde cada evento possui uma probabilidade p, p é sempre igual e $0 < p < 1$. A variável aleatória X, correspondente ao número de sucessos em um experimento binomial, e tem distribuição binomial $b(n, p)$, definida pela função de probabilidade abaixo [12].*

$$b(k; n, p) = P(X = k|n, p) = \binom{n}{k} p^k q^{n-k}, k = 0, 1, ..., n. \tag{5.3}$$

Por exemplo, suponha que atiremos uma moeda quatro vezes. Podemos utilizar a distribuição binomial para calcularmos a prioridade de conseguirmos duas caras e duas coroas. Considerando cara sucesso, temos quatro testes, $n = 4$, onde $p = 0,5$, e queremos dois sucessos, logo $k = 2$. Teremos então que $b(2; 4, 0, 5) = 0,375$.

5.4 Descrição do Modelo

Nesta seção será descrito o modelo de simulação desenvolvido.

O foco do nosso trabalho é o módulo de reputação na camada MAC, um dos módulos do sistema de reputação proposto. Foi criado, portanto, um modelo de simulação para avaliar a partir de que momento o nó normal consegue identificar um nó malicioso e quanto o nó malicioso pode ganhar em relação ao número de transmissões.

No modelo de simulação desenvolvido, consideramos as seguintes premissas:

- Os nós são estáticos.
- Os nós podem se comunicar com todos os nós, ou seja, a comunicação é *single hop*. Graficamente a rede pode ser representada por um grafo completo, como na figura 5.2.
- Os nós podem monitorar a comunicação de todos o nós e compará-las.
- Em cada cenário há apenas um nó malicioso, os outros tem comportamento normal.
- Todos nós estão continuamente tentando transmitir.

As simulações foram realizadas em um simulador desenvolvido especificamente para este trabalho, isto é, não foi utilizado nenhum simulador existente no mercado. O simulador foi desenvolvido na linguagem java e teve como base o padrão 802.11. Os nós com comportamento padrão geram um número aleatório dentro do intervalo de $[0, Cw_{min}]$. O valor de CW_{min} depende do meio físico, e nesta simulação consideramos o valor de CW_{min} igual a 15. Caso haja uma colisão, esta faixa de números pode ser dobrada sucessivamente até atingir o valor máximo de CW_{max} que também depende do meio físico. Consideramos o valor de CW_{max} igual a 1023, o que equivale a ocorrer seis colisões sucessivas. Quando não houver mais colisões a janela de contenção volta ao valor inicial.

A simulação foi realizada em duas etapas: Simulação de Uso do Canal e Simulação do Nível de Mal Comportamento.

5.4.1 Simulação de Uso do Canal

Nesta primeira etapa da simulação, chamada de simulação de uso do canal, foi observada a porcentagem de uso do canal por cada nó. Desta forma é possível comparar o uso real do nó em relação ao canal de transmissão, se considerarmos o desvio padrão, com a esperança de uso do canal de cada nó. A esperança é

calculada como $100/n$, onde n é o número de nós. A utilização do desvio padrão é importante para termos um limite do que seria aceitável, se um nó transmitisse mais ou menos que outro.

O objetivo desta simulação é descobrir para quais valores de CW_{min}, um nó egoísta poderia não ser detectado e com quais valores ele seria facilmente identificado. Uma questão importante é o quanto um nó malicioso poderia transmitir a mais que outro nó normal sem ser considerado mal comportado.

5.4.2 Simulação do Nível de Mal Comportamento

O objetivo desta etapa da simulação, chamada de simulação do nível de mal comportamento, é tentar identificar um nó malicioso. Em cada teste, o nó observador, com comportamento normal, monitora um nó suspeito, e a cada segundo compara o seu número de transmissões com o do nó suspeito utilizando a seguinte fórmula:

$$T_m(t) \geq T_n(t) + S(t) \times \delta, \tag{5.4}$$

onde t é o segundo atual, $T_m(t)$, é o número de transmissões do nó malicioso, $T_n(t)$ é o número de transmissões do nó normal. $S(t) \times \delta$ é o limiar considerado, onde $S(t)$ é o desvio padrão do nó observador, e δ é definido como tolerância e $0 \leq \delta \leq 1$. Nós podemos entender tolerância como uma forma de escolhermos o limiar ideal, quanto menor for a tolerância mais rígido seremos.

É importante sabermos a tolerância ideal para cada cenário. Pois, em alguns casos uma tolerância baixa pode ser suficiente, mas ao mesmo tempo pode permitir que um nó malicioso não seja detectado em outros cenários. Uma tolerância alta pode ser necessária quando for difícil detectar os nós mal comportados, mas também pode significar que nós normais sejam erroneamente identificados como maliciosos.

Definimos a pontuação do nó malicioso como Mp_n, onde $0 \leq Mp_n \leq 1$ e n é o nó observado. Se a fórmula 5.4 for verdadeira, $Mp_n(t) = 1$ sendo marcado como mal comportado, senão, $Mp_n(t) = 0$ sendo marcado como bem comportado; onde t é o segundo atual.

Além disso, utilizamos um mecanismo de *aging*, ou seja, os dados mais recentes possuem peso maior que os mais antigos, por meio da média ponderada da pontuação do nó malicioso, chamada de Nível de Mal Comportamento, calculada pela fórmula 5.5, onde a fórmula 5.6 é o somatório dos pesos da média ponderada e igual a fórmula 5.5. O uso da técnica de *aging* é importante para oferecermos redenção aos nós mal comportados que tenham tornado-se bem comportados, e também para evitarmos que nós bem comportados se tornem maliciosos sem serem detectados, conhecidos também como nós bizantinos.

$$M_n(t) = \frac{Mp_n(0) + 2 \times Mp_n(1) + ... + t \times Mp_n(t-1)}{1 + 2 + ... + t} \tag{5.5}$$

$$M_n(t) = 2 \times \frac{\sum_{i=0}^{t-1} (i+1) \times Mp_n(i)}{t \times (t+1)} \tag{5.6}$$

Se tivéssemos uma situação onde um nó observou um nó suspeito por três segundos e obteve para cada segundo $Mp_n(0) = 1$, $Mp_n(1) = 1$ e $Mp_n(2) = 0$, teríamos, usando a fórmula 5.6 que o nível de mal comportamento para o nó suspeito é $0,5$.

Vimos neste capítulo a descrição do modelo de simulação desenvolvido e a nossa proposta de sistema de reputação entre camadas. A idéia desse sistemas de reputação surgiu, pois vimos, que parte das técnicas que utilizamos no nosso modelo de simulação na camada MAC poderiam ser estendidas para detectar mau comportamento dos nós nas outras camadas.

Capítulo 6

Resultados Experimentais e Análise

Neste capítulo serão apresentados os resultados das simulações. Os cenários das simulações realizadas foram feitos com dois, quatro, oito, dezesseis, e trinta e dois nós. Para cada cenário foram feitos cem testes. A duração de cada teste varia com o tipo de simulação. Após são apresentados os resultados e uma análise dos mesmos.

6.1 Simulação de Uso do Canal

Nesta simulação cada teste teve uma duração de $4 \times n$ segundos, onde n é o número de nós. Os gráficos 6.1, 6.2, 6.3, 6.4, e 6.5 apresentam os resultados para cada cenário na simulação do uso de canal, sendo que o eixo x mostra o valor de CW_{min} para o nó malicioso, que varia de 14 a 0, e o eixo y mostra a porcentagem de uso do canal. A curva dos nós normais representa a média de todos os nós com comportamento normal. Estes gráficos nos mostram a porcentagem do canal utilizada pelo nó malicioso e pelos nós normais. Podemos observar que à medida que o nó mal comportado diminui o valor de sua CW_{min}, ele passa a utilizar uma parcela maior do canal e os nós normais vão utilizando menos, sendo que quando ele utiliza $CW_{min} = 14$, ambas curvas estão próximas da esperança.

O gráfico 6.1 mostra a porcentagem de uso de canal para a simulação com dois nós. Com $CW_{min} = 14$ para o nó egoísta, a porcentagem de uso de canal para ambos nós é perto de 50%. A medida que CW_{min} vai diminuindo o nó bem comportado passa a ser prejudicado e o nó malicioso vai ganhando vantagem. Com $CW_{min} = 10$ o nó normal utiliza 39,89% e o malicioso 60,11% do canal. Já com $CW_{min} = 4$ temos aproximadamente 20,26% e 79,74%, respectivamente, e com $CW_{min} = 0$ o nó normal praticamente não consegue transmitir.

Se observamos o gráfico 6.2, referente a simulação de quatro nós, veremos uma situação parecida, mas um pouco pior. O nó malicioso ganha vantagem mais rapidamente e o nó normal é prejudicado em um ritmo menor, pois fica diluído entre os nós normais. Se o nó malicioso utilizar $CW_{min} = 8$, irá obter 39,66% do canal, e com $CW_{min} = 6$ utilizará 47,78% do canal, enquanto a outra parte do canal fica compartilhada pelos três nós normais.

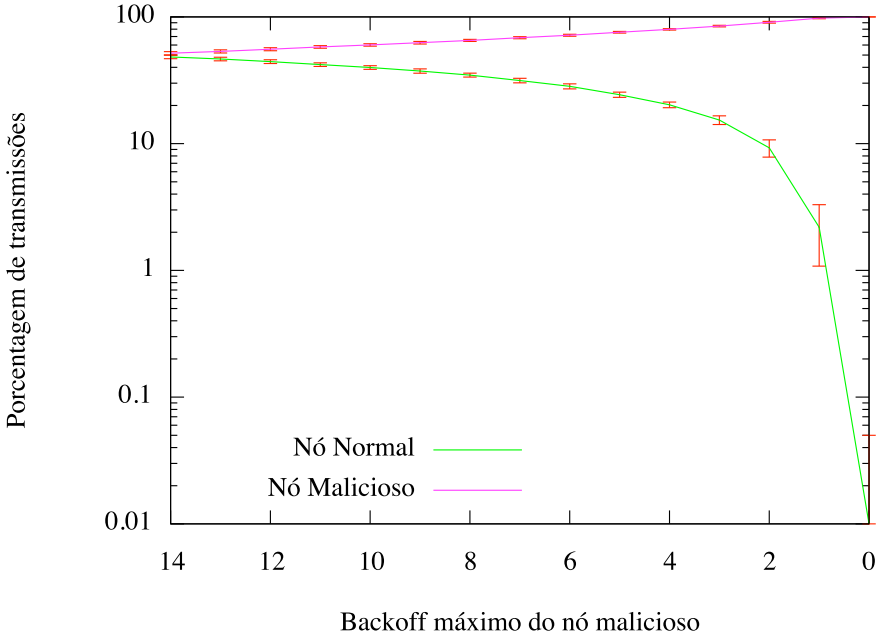

Figura 6.1: Porcentagem de uso do canal com dois nós

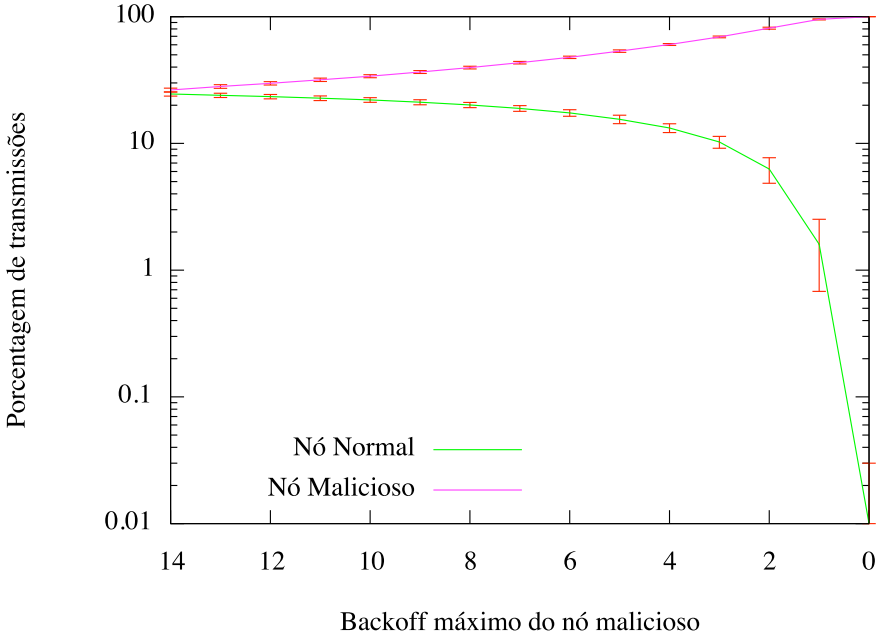

Figura 6.2: Porcentagem de uso do canal com quatro nós

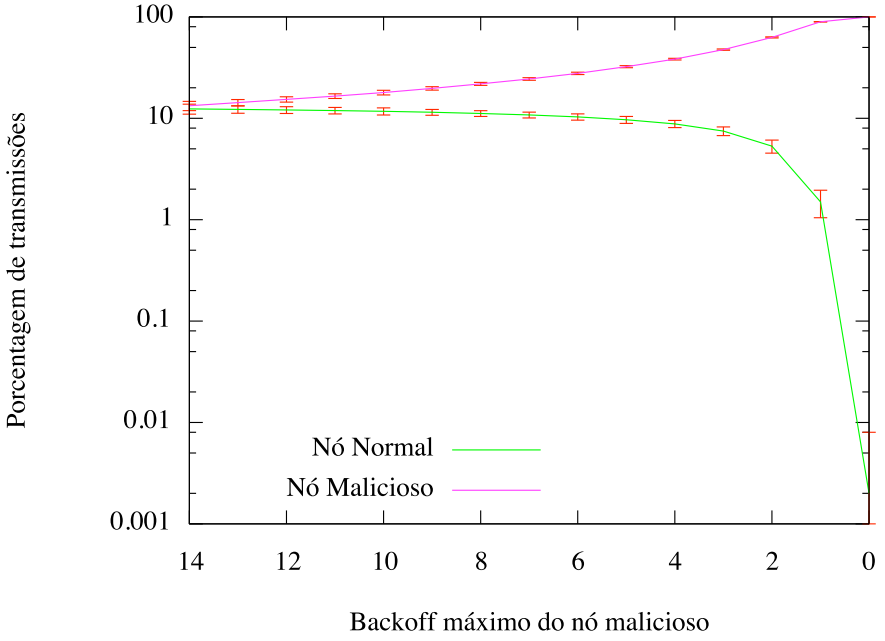

Figura 6.3: Porcentagem de uso do canal com oito nós

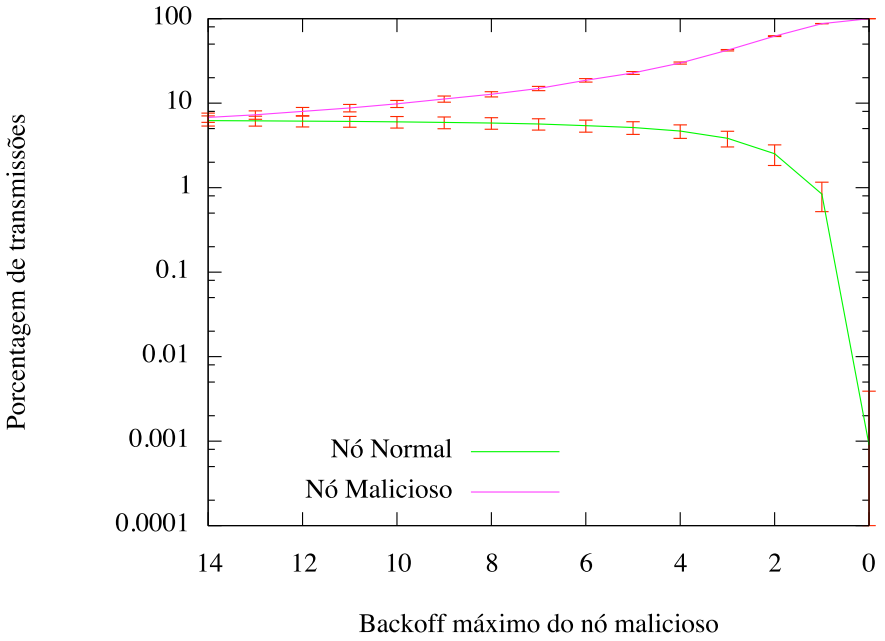

Figura 6.4: Porcentagem de uso do canal com dezesseis nós

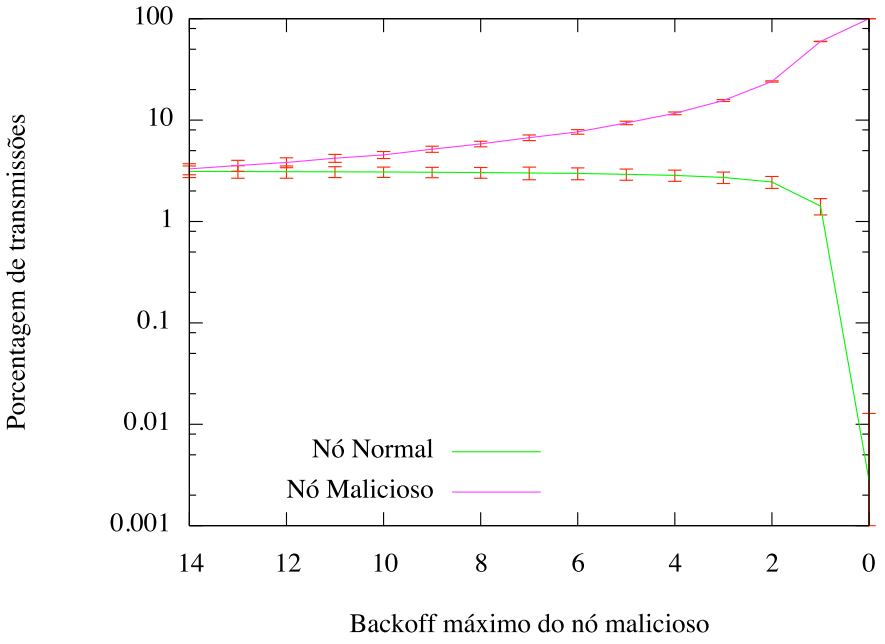

Figura 6.5: Porcentagem de uso do canal com trinta e dois nós

Na simulação de oito nós, mostrada no gráfico 6.3, com $CW_{min} = 9$, o nó malicioso usa 19,73% do canal, enquanto os nós normais possuem 11,47%. Podemos observar que neste gráfico a curva dos nós normais começa a se assemelhar com uma reta e decresce muito pouco até $x = 6$.

Com dezesseis nós, no gráfico 6.4, o nó malicioso consegue utilizar 18,64% do canal com $CW_{min} = 6$, enquanto os nós normais tem 5,42% cada, comparando com o caso anterior, ele precisa de uma CW_{min} menor para conseguir uma quantidade semelhante do canal. Isto é esperado, pois o número de nós dobrou e a competição pelo canal está mais acirrada.

Na simulação com trinta e dois nós, gráfico 6.5, a curva do nó normal se assemelha a uma reta até a posição $x = 4$. A esperança neste caso é de 3,12% para cada nó, entretanto, com $CW_{min} = 5$, o nó malicioso consegue transmitir 9,38%, enquanto os outros nós transmitem em média 2,92%, o que não é muito longe da esperança.

Para melhor compreendermos os gráficos de uso do canal, utilizamos uma ferramenta da estatística: a distribuição binomial. Fazemos uma análise de cada cenário utilizando a distribuição binomial. Consideramos para esta análise, o intervalo de cada ponto junto com o desvio padrão de um dos nós normais. Desta forma podemos saber qual seria a probabilidade real do nó usar o canal na mesma medida que ele utilizou caso não houvesse nós mal comportados. Se a probabilidade for pequena significa que o nó não está transmitindo de acordo com o esperado. É importante observar que uma baixa probabilidade não significa necessariamente que o nó esteja transmitindo mais que o normal mas, pode ser que o nó esteja transmitindo menos que o normal. Logo, uma baixa probabilidade não significa que o nó seja malicioso, mas sim que há algo errado na rede, e que possivelmente seja a presença de um nó malicioso afetando negativamente os ou-

tros nós. O eixo x do gráfico nos mostra o valor de CW_{min} utilizado pelo nó malicioso e o eixo y nos mostra a probabilidade da quantidade de transmissões do nó estar dentro do que seria aceitável em um ambiente onde todos os nós são normais, sendo que o intervalo aceitável seria o valor de transmissões do nó mais e menos o desvio padrão.

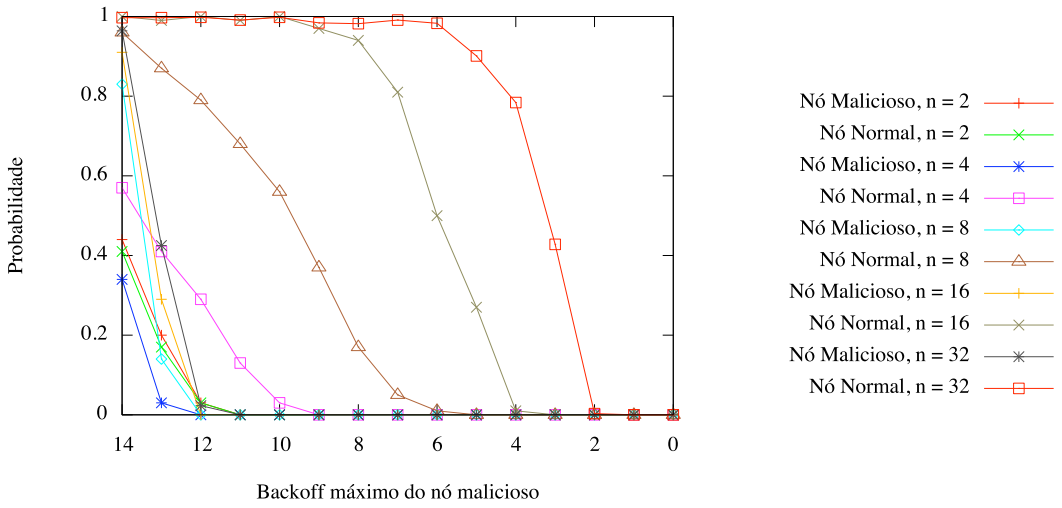

Figura 6.6: Distribuição binomial da simulação de uso do canal

Observando o gráfico 6.6 e comparando-o com os gráficos 6.1, 6.2, 6.3, 6.4, e 6.5 podemos perceber que a curva representando o nó malicioso cai para uma probabilidade perto de zero no mesmo ponto que as curvas dos nós normais e do nó malicioso se separam nos gráficos de uso do canal.

Analisando as curvas do gráfico 6.6, observamos que no caso da simulação com dois nós a curva do nó normal e do nó malicioso estão praticamente juntas. Isto ocorre pois na mesma medida que um nó ganha vantagem o outro perde. Com o aumento de nós bem comportados em relação aos mal comportados, estas curvas se separam, com a tendência da curva do nó malicioso cair rapidamente e a do nó normal demorar um período maior para cair.

Se tivermos um caso onde um nó normal não seja capaz de monitorar especificamente outro nó, o nó normal não terá como comparar suas transmissões, mas caso consiga contar os pacotes que passam por sua região, poderá usar a distribuição binomial e assim ter algum indicativo de presença de um nó malicioso na rede. Em casos com poucos nós a distribuição binomial é eficiente, entretanto, em redes com muitos nós um nó normal só é capaz de perceber algo errado se o nó egoísta usar um valor CW_{min} muito baixo. No caso da simulação com 32 nós, isso aconteceria somente se o nó mal comportado usasse $CW_{min} = 3$. Se o nó tiver a capacidade de monitorar os outros nós, a distribuição binomial se torna inútil, pois traz a mesma informação que o gráfico de uso de canal traz. E isto é coerente com os outros gráficos, no gráfico 6.5 a curva do nó normal demora muito para decrescer consideravelmente, e isto acontece no mesmo ponto que no gráfico da distribuição binomial.

Podemos perceber com estes gráficos que um nó malicioso pode ser facilmente detectado em uma rede com poucos nós, sendo que os nós normais serão bastante

prejudicados, porém, em uma rede com muitos nós a detecção será mais difícil, e os nós normais praticamente não serão afetados. Podemos também entender que quanto maior for a relação do número de nós bem comportados com o número de nós mal comportados, menos afetados serão os nós normais.

6.2 Simulação do Nível de Mal Comportamento

Nesta simulação cada teste teve uma duração de $8 \times n$ segundos, onde n é o número de nós. Para a tolerância utilizamos os seguintes valores: 0, 0,1, 0,2, 0,3, 0,4, 0,5, 0,6, 0,7, 0,8, 0,9 e 1. Nos gráficos não apresentamos todos os valores de tolerância, mas somente as mais importantes para cada caso para termos um gráfico claro e legível. Esta simulação não foi feita com todos os valores possíveis da CW_{min} do nó egoísta, pois com uma CW_{min} muito baixa o nível de mal comportamento tende para 1 muito rapidamente, chegando a um ponto que ela é sempre 1. Abaixo deste ponto ela sempre terá o valor 1, sendo desnecessário simular para estes valores.

Esta simulação teve início com o valor normal para CW_{min} igual a 15, e foi sendo decrescida até o necessário, que depende do número de nós da simulação. Apresentamos a seguir os resultados obtidos para $CW_{min} = \{15, 14, 13\}$. O nosso objetivo é encontrar uma tolerância ideal onde a curva para o comportamento normal seja decrescente e para um comportamento anômalo seja crescente. Nos gráficos encontrados logo abaixo, o eixo y mostra o nível de mal comportamento, descrito na fórmula 5.6 e o eixo x representa o tempo. Os gráficos 6.7, 6.8, 6.9, 6.10, e 6.11 mostram nosso resultado, onde o eixo x representa o tempo em segundos e o eixo y o nível de mal comportamento. Observando-os veremos como o nó observador vai modificando sua visão em relação ao nó suspeito no decorrer do tempo, podendo, depois de um determinado tempo, classificar um nó como normal ou mal comportado.

A simulação do nível de mal comportamento tinha como objetivo tentar identificar um nó mal comportado. Analisando os gráficos veremos que quanto maior o número de nós, maior será a dificuldade em identificar um nó malicioso, principalmente se este estiver usando um valor de CW_{min} próximo do valor padrão. Por exemplo, no gráfico 6.11 um nó malicioso utilizando $CW_{min} = 14$ será rapidamente identificado erroneamente como normal, se usarmos a tolerância $\delta = 1$. Mas por outro lado, se usarmos $\delta = 0$, ele será identificado como malicioso, entretanto não será tão rápido quanto no primeiro caso. Se formos para o caso com dois nós, gráfico 6.7, com $CW_{min} = 14$ e $\delta = 1$, veremos que a curva do nível de mal comportamento ficará estável e por volta de 0,7.

Quando o nó malicioso usa $CW_{min} = 13$, na simulação de dois nós ele é rapidamente identificado, independente da tolerância usada. Entretanto, com quatro nós, no gráfico 6.8, o nó será identificado com qualquer tolerância, mas o crescimento da curva varia mais com tolerâncias diferentes. Porém com oito nós, no gráfico 6.9, o nível de mal comportamento só passará de 0,9 após aproximadamente 20 segundos. Com dezesseis nós, no gráfico 6.10, após 120 segundos, a curva não chegou ainda a 0,9. Com trinta e dois nós a situação é pior, após 250 segundos a nível de mal comportamento ainda não chegou a 0,9. Isto ocorre por

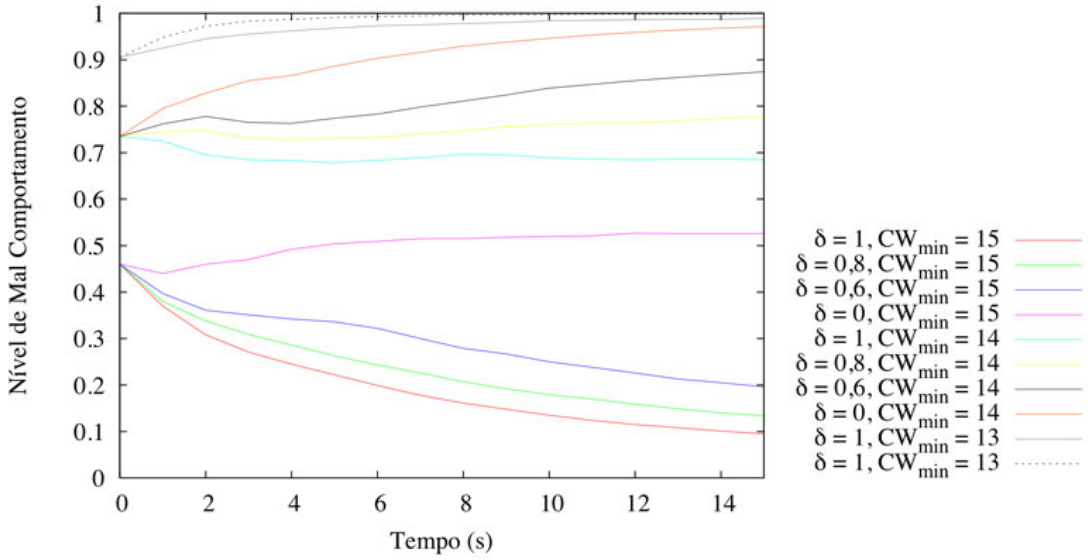

Figura 6.7: Nível de mal comportamento com dois nós

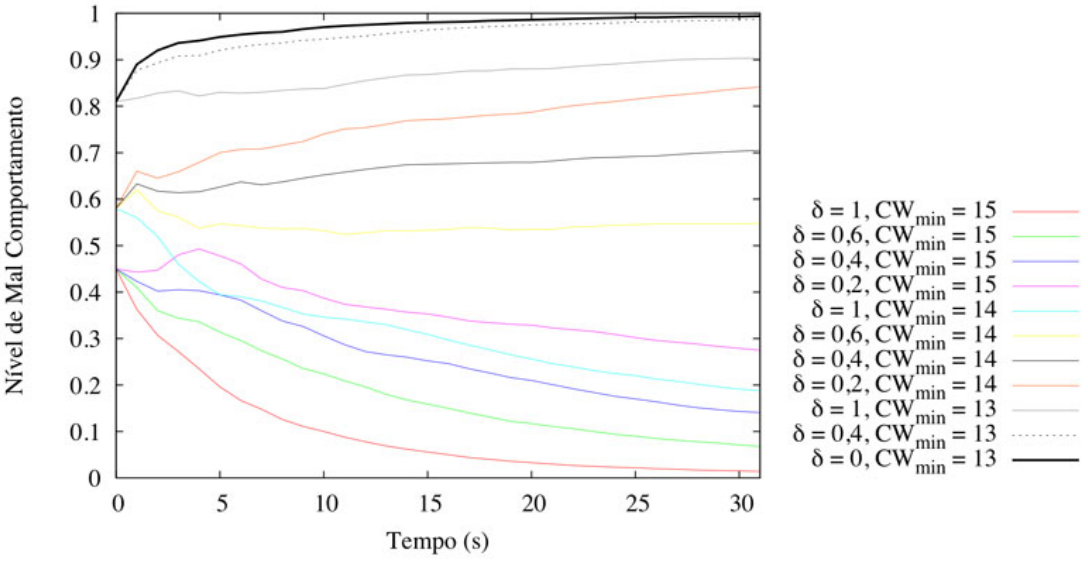

Figura 6.8: Nível de mal comportamento com quatro nós

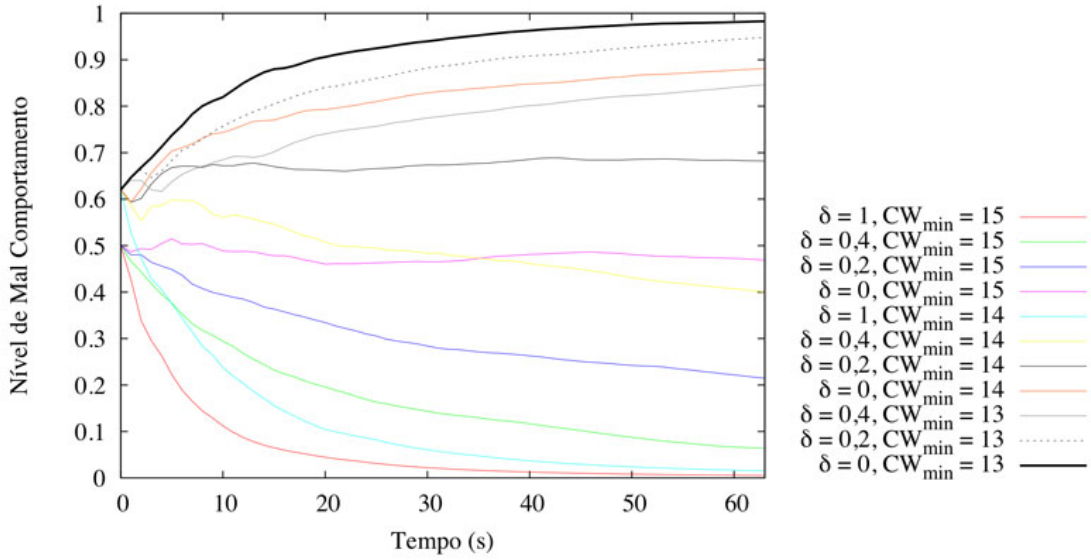

Figura 6.9: Nível de mal comportamento com oito nós

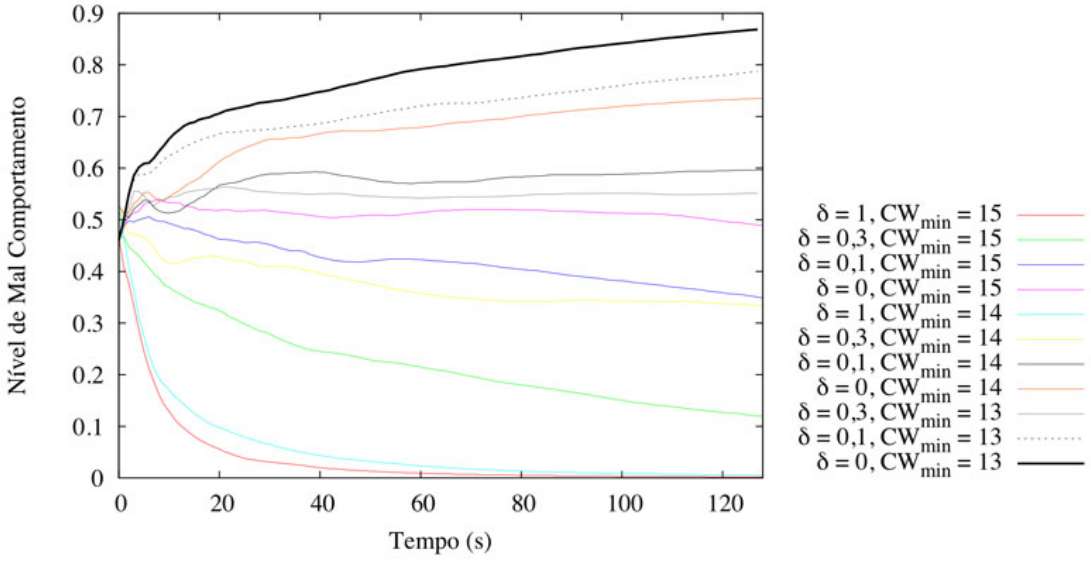

Figura 6.10: Nível de mal comportamento com dezesseis nós

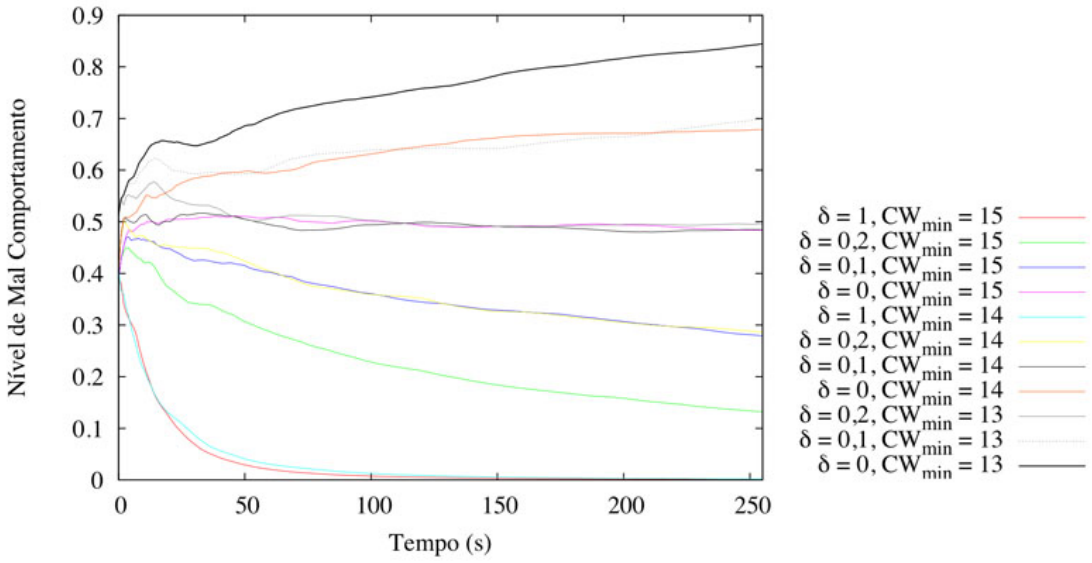

Figura 6.11: Nível de mal comportamento com trinta e dois nós

causa do número de nós na rede, mas embora o nível de mal comportamento não chegue a 1 rapidamente, ele é crescente para estes casos e em todos eles o nível passa de 0,6 bem no começo da simulação.

Se observarmos o comportamento de um nó normal com $CW_{min} = 15$, nas simulações de dois e quatro nós o nó sempre será identificado como normal, mesmo com tolerâncias diferentes a curva é sempre decrescente. Já com oito, dezesseis, e trinta e dois nós, se usarmos tolerância zero, a curva ficará estável e próxima de 0,5. Não desejamos falsos positivos, e vemos que estas curvas embora dêem um nível de mal comportamento de 0,5 para um nó normal, não passam muito deste valor. Isto não implica que não possa ocorrer falsos positivos com nível de mal comportamento próximos de 1, pois estas curvas representam a média de vários testes.

Podemos perceber que dependendo do número de nós precisamos de valores de δ diferente, é preciso então, definir o valor ideal para δ. No início da simulação, o nó observador não tem muitas informações sobre o nó suspeito e é natural que o valor do nível de mal comportamento não seja muito realista, é preciso um tempo para o nó chegar a uma conclusão efetiva sobre o comportamento do nó suspeito, e este tempo aumenta com o aumento do número de nós. Entretanto, queremos garantir que a curva para nós normais seja decrescente e para nós maliciosos seja crescente, independente do valor de CW_{min}. Logo, queremos que a curva para $CW_{min} = 15$ seja decrescente, e para $CW_{min} = 14$ seja crescente. Se garantirmos que para $CW_{min} = 14$ a curva seja crescente, logicamente as curvas com $CW_{min} < 14$ também serão.

Nós propomos que a fórmula para a tolerância ideal seja:

$$\delta(n) = \frac{2C}{n}, \tag{6.1}$$

onde n é o número de nós e C é uma constante. O valor de C pode depender de vários fatores, como as aplicações que estão sendo executados nos nós, o número de nós maliciosos, interferência de sinal na rede e outros. Para identificar o valor de C na nossa simulação, podemos considerar o caso de um cenário simples, como o cenário com dois nós. Podemos observar no gráfico que a tolerância ideal gira em torno de 0,8. Então para $n = 2$ podemos usar $\delta(2) = 0,8$, implicando em $\delta = 0,8$.

Utilizando $\delta = 0,8$ para os outros cenários teremos as seguintes tolerâncias: $\delta(4) = 0,4, \delta(8) = 0,2, \delta(16) = 0,1, \delta(32) = 0,05$. Se observarmos os gráficos, podemos verificar que estes valores garantem que as curvas de nós normais sejam decrescentes e de nós mal comportados sejam crescentes.

Neste capítulo, apresentamos os resultados da simulação e a análise dos dados. Nossos resultados são limitados pois foram baseados em um modelo simplificado. Entretanto, eles são válidos pois dão indícios do ganho de um nó malicioso e do prejuízo que este causa da rede. Nossos resultados da simulação do Nível de Mal Comportamento nos mostram as dificuldades em detectarmos um nó malicioso e quanto tempo é gasto para que tenhamos certeza que um nó é mal ou bem comportado, não encontramos nenhum resultado na literatura científica que nos desse uma direção para este problema. É preciso que estes resultados sejam confirmados em uma rede ad hoc real, mas com eles já é possível termos uma direção de como podemos analisar o problema de mau comportamento na camada de enlace em uma rede real.

Capítulo 7

Conclusão e Trabalhos Futuros

Neste trabalho foram discutidos vários tipos de desvio de conduta que um dispositivo poderia ter em uma rede ad hoc, em várias camadas. Como dito anteriormente, nós que não desejam cooperar devem ser identificados e punidos, independente de qual camada estejam atuando. Pois, não é viável para a rede, a presença de nós egoístas.

Um sistema de reputação ideal para redes ad hoc deve-se preocupar em verificar o comportamento dos nós em todas camadas. Ainda não existe um sistema de reputação que seja *cross-layer*. A maior parte dos sistemas propostos analisam apenas uma camada específica. Estes sistemas são falhos, pois nem consideram a hipótese que o nó poderia atacar em outra camada. Eles, portanto, possuem vunerabilidades, pois nós maliciosos podem agir em outras camadas sem serem identificados.

Não foi encontrado na literatura nenhum sistema de reputação *cross-layer*. Os sistemas existentes apenas observam a camada de rede ou realizam a detecção de mau comportamento na camada de enlace como o DOMINO. Entretanto, o DOMINO não é um sistema de reputação e nem é específico para redes ad hoc. Pois, como vimos ele considera a existência de um ponto de acesso.

É necessário que um sistema de reputação *cross-layer* seja robusto para conseguir tratar os problemas de cada camada e ao mesmo tempo seja simples e econômico, causando o menor impacto em bateria e processamento para os nós. O foco deste trabalho foi na camada de enlace, entretanto, um sistema de reputação deve-se preocupar também com a camada de rede, de transporte, e em alguns casos a camada de aplicação. Todavia, muitos problemas de uma camada superior podem ser evitados, se os problemas da camada inferior forem tratados.

Este trabalho mostrou que o uso da vazão para analisar o mal comportamento dos nós é um parâmetro válido e pode ser usado com eficiência. A simulação realizada nos fornece algumas direções em como um nó egoísta poderia agir. Em uma rede com poucos nós uma diferença muito grande da janela de contenção com os nós normais levaria o nó a ser rapidamente detectado. Mas em uma rede com muitos nós isto poderia ser difícil de ocorrer. Além disso, com o aumento do número de nós a dificuldade em chegar a uma conclusão precisa do comportamento do nó aumenta, exigindo um tempo maior de observação.

Para trabalhos futuros seria importante verificar nossa simulação de mau comportamento na camada MAC em um ambiente real, e testá-lo em cenários mais

complexos como por exemplo, ambientes com interferência, com a presença de terminais escondidos e terminais expostos, e alta mobilidade dos nós. Depois disto seria importante implementar o sistema de reputação *cross-layer* proposto e validá-lo nos cenários e situações descritos acima.

Referências

[1] Aune, F. (2004). Cross-layer design tutorial.

[2] Balakrishnan, H., Seshan, S., A., E., and Katz, R. H. (1995). Improving tcp/ip performance over wireless networks. In *MobiCom '95: Proceedings of the 1st annual international conference on Mobile computing and networking*, pages 2–11, New York, NY, USA. ACM.

[3] Bansal, S. and Baker, M. (2003). Observation-based cooperation enforcement in ad hoc networks. Technical report, Stanford University, CA.

[4] Basagni, S., Conti, M., Giordano, S., and Stojmenovac, I. (2003). *Mobile Ad Hoc Networking*. Wiley-IEEE Press.

[5] Buchegger, S. and Boudec, J. L. (2002). Performance analysis of the CONFIDANT protocol: Cooperation of nodes — fairness in dynamic ad-hoc networks. In *Proceedings of IEEE/ACM Symposium on Mobile Ad Hoc Networking and Computing (MobiHOC)*, Lausanne, CH. IEEE.

[6] Buchegger, S. and Boudec, J. L. (2004). A robust reputation system for peer-to-peer and mobile ad-hoc networks. In *P2PEcon 2004*.

[7] Buchegger, S. and Boudec, J.-Y. L. (2003). A robust reputation system for mobile ad hoc networks. Technical Report IC/2003/50, EPFL-DI-ICA, Lausanne, Switzerland.

[8] Buchegger, S. and Boudec, J. Y. L. (2005). Self-policing mobile ad hoc networks by reputation systems. *Communications Magazine, IEEE*, 43(7):101–107.

[9] Buttyán, L. and Hubaux, J. (2001). Nuglets: a virtual currency to stimulate cooperation in self-organized ad hoc networks. Technical Report DSC/2001, Swiss Federal Institute of Technology – Lausanne, Department of Communication Systems.

[10] Comer, D. E. (1999). *Redes de Computadores*. Artmed Editora Ltda., São Paulo, 2 edition.

[11] Cormen, T. H., Leiserson, C. E., and Rivest, R. L. (1997). *Introduction to Algorithms*. MIT Press, 18 edition.

[12] de O. Bussab, W. and Morettin, P. A. (2002). *Estatística Básica*. Editora Saraiva, São Paulo, 5 edition.

[13] F. Kargl, A. Klenk, S. S. and Weber, M. (2004). Advanced detection of selfish or malicious nodes in ad hoc networks. Technical report, University of Ulm, Dep. of Multimedia Computing, Ulm, Germany.

[14] Gast, M. S. (2002). *802.11 Wireless Networks: The Definitive Guide*. O'Reilly.

[15] Guang, L. and Assi, C. (2006a). Cross-layer cooperation to handle MAC misbehavior in ad hoc networks. *Electrical and Computer Engineering, 2006. CCECE '06. Canadian Conference on*, pages 219–222.

[16] Guang, L. and Assi, C. (2006b). Mitigating smart selfish MAC layer misbehavior in ad hoc networks. *Wireless and Mobile Computing, Networking and Communications, 2006. (WiMob'2006). IEEE International Conference on*, pages 116–123.

[17] Guang, L., Assi, C., and Benslimane, A. (2006). Modeling and analysis of predictable random backoff in selfish environments. In *MSWiM '06: Proceedings of the 9th ACM international symposium on Modeling analysis and simulation of wireless and mobile systems*, pages 86–90, New York, NY, USA. ACM.

[18] He, Q., Wu, D., and Khosla, P. (2004). SORI: A secure and objective reputationbased incentive scheme for ad-hoc networks. In *Proceedings of IEEE Wireless Communications and Networking Conference (WCNC2004)*, volume 2, pages 825–830. IEEE.

[19] Hu, J. (2005). Cooperation in mobile ad hoc networks. Technical Report TR-050111, Florida State University.

[20] Hu, Y., Perrig, A., and Johnson, D. B. (2005). Ariadne: a secure on-demand routing protocol for ad hoc networks. *Wirel. Netw.*, 11(1-2):21–38.

[21] IEEE 802.11 Standard (2007). IEEE Standard for Information technology-Telecommunications and information exchange between systems-Local and metropolitan area networks-Specific requirements - Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications. *IEEE Std 802.11-2007 (Revision of IEEE Std 802.11-1999)*, pages C1–1184.

[22] Josang, A., Ismail, R., and Boyd, C. (2007). A survey of trust and reputation systems for online service provision. *Decis. Support Syst.*, 43(2):618–644.

[23] Jøsang, A. and Pope, S. (2005). Semantic constraints for trust transitivity. In *Proceedings of the Second Asia-Pacific Conference on Conceptual Modelling (APCCM)*.

[24] Kyasanur, P. and Vaidya, N. (2003). Detection and handling of mac layer misbehavior in wireless networks. *Dependable Systems and Networks, 2003. Proceedings. 2003 International Conference on*, pages 173–182.

[25] Li, M. (2007). Cross-layer resource control to improve tcp performance over wireless network. *Computer and Information Science, 2007. ICIS 2007. 6th IEEE/ACIS International Conference on*, pages 706–711.

[26] M. Conti and E. Gregori and G. Maselli (2004). Cooperation issues in mobile ad hoc networks. *Distributed Computing Systems Workshops, 2004. Proceedings. 24th International Conference on*, pages 803–808.

[27] Marti, S., Giuli, T. J., Lai, K., and Baker, M. (2000). Mitigating routing misbehavior in mobile ad hoc networks. In *Proceedings ACM/IEEE International Conference on Mobile Computing and Networking (MobiCOM)*, volume 2, pages 255–265.

[28] Michiardi, P. and R.Molva (2001). CORE: A collaborative reputation mechanism to enforce node cooperation in mobile ad hoc networks. Technical Report EURECOM+816, Institut Eurecom, France.

[29] Murthy, C. S. R. and Manoj, B. S. (2004). *Ad Hoc Wireless Networks: Architecture and Protocols*. Prentice Hall, Upper Saddle River, N.J.

[30] Papadimitratos, P. and Haas, Z. (2002). Secure routing for mobile ad hoc networks. In *SCS Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS 2002).*

[31] Raya, M., Hubaux, J., and Aad, I. (2004). DOMINO: a system to detect greedy behavior in ieee 802.11 hotspots. In *MobiSys '04: Proceedings of the 2nd international conference on Mobile systems, applications, and services*, pages 84–97, New York, NY, USA. ACM.

[32] Sanzgiri, K., Dahill, B., Levine, B., Shields, C., and Belding-Royer, E. (2002). A secure routing protocol for ad hoc networks. *Network Protocols, 2002. Proceedings. 10th IEEE International Conference on*, pages 78–87.

[33] Srivastava, V. and Motani, M. (2005). Cross-layer design: a survey and the road ahead. *Communications Magazine, IEEE*, 43(12):112–119.

[34] Tanenbaum, A. S. (2003). *Redes de Computadores*. Elsevier Editora Ltda., Rio de Janeiro, 4 edition. Tradução Vandenberg D. de Souza.

[35] Theodorakopoulos, G. and Baras, J. S. (2004). Trust evaluation in ad hoc networks. In *Proceedings of 2004 ACM Workshop on Wireless Security*, pages 1–10. ACM Press.

[36] Xiao, Y. (2004). IEEE 802.11e: QoS provisioning at the MAC layer. *Wireless Communications, IEEE [see also IEEE Personal Communications]*, 11(3):72–79.

[37] Yau, P. and Mitchell, C. J. (2003). Reputation methods for routing security for mobile ad hoc networks. In *Proceedings of SympoTIC '03, Joint IST Workshop on Mobile Future and Symposium on Trends in Communications*, pages 130–137.

[38] Zhong, S., Yang, Y., and Chen, J. (2003). Sprite: A simple, cheat-proof, credit-based system for mobile ad hoc networks. In *In Proceedings of IEEE INFOCOM'03*, volume 3, pages 1987–1997, San Francisco, CA.

[39] Zimmermann., H. (1980). OSI Reference Model–The ISO Model of Architecture for Open Systems Interconnection. *Communications, IEEE Transactions on [legacy, pre - 1988]*, 28(4):425–432.