UNIVERSIDADE FEDERAL DO RIO DE JANEIRO
PROGRAMA DE PÓS-GRADUAÇÃO EM INFORMÁTICA

Helio Mendes Salmon

Sistema de Detecção de Intrusão Imuno-inspirado customizado para Redes de Sensores Sem Fio

Rio de Janeiro
2011

Helio Mendes Salmon

Sistema de Detecção de Intrusão Imuno-inspirado customizado para Redes de Sensores Sem Fio

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática, Universidade Federal do Rio de Janeiro, como requisito parcial à obtenção do título de Mestre em Informática

Orientadoras: Luci Pirmez
Silvana Rossetto

Rio de Janeiro
2011

S171 Salmon, Helio Mendes
Sistema de detecção de intrusão imuno-inspirado customizado para rede de sensores sem fio / Helio Mendes Salmon. – 2011.
93 f.: il.

Dissertação (Mestrado em Informática) – Universidade Federal do Rio de Janeiro, Instituto de Matemática, Núcleo de Computação Eletrônica, 2011.

Orientadoras: Luci Pirmez ; Silvana Rossetto

1. Rede de Sensores Sem Fio (Teses). – 2. Sistema de Detecção de Intrusão (Teses). – 3. Sistema Imunológico Artificial (Teses). – I. Luci Pirmez (Orient.). II. Silvana Rossetto (Orient.). III. Universidade Federal do Rio de Janeiro, Instituto de Matemática, Núcleo de Computação Eletrônica. IV. Título.

CDD.

Helio Mendes Salmon

Sistema de Detecção de Intrusão Imuno-inspirado customizado para Redes de Sensores Sem Fio

Dissertação de Mestrado apresentada ao Programa de Pós-Graduação em Informática, Universidade Federal do Rio de Janeiro, como requisito parcial à obtenção do título de Mestre em Informática

Aprovada em: Rio de Janeiro ___ de _____ de _____.

Prof[a]. Luci Pirmez – Orientadora
D.Sc., COPPE/UFRJ, Brasil
UFRJ / PPGI

Prof[a]. Silvana Rossetto – Orientadora
D.Sc. PUC/RJ, Brasil
UFRJ / PPGI

Prof. Célio Vinicius Neves de Albuquerque
Ph. D. University of California, EUA
UFF / IC

Prof. Luis Henrique Maciel Kosmalski Costa
Dr. Université Pierre et Marie Curie, França
UFRJ / COPPE

Prof. Paulo Henrique de Aguiar Rodrigues
Ph. D. University of California, EUA
UFRJ / PPGI

Rio de Janeiro
2011

Dedico este trabalho a todos da minha família, especialmente minha esposa Andrea e minha filha Isabella, que, como sempre, me ajudaram a superar mais este desafio.

AGRADECIMENTOS

À Deus, acima de tudo, por ter me dado esta oportunidade e iluminar meu caminho.

Aos meus pais, Helio e Olga, pelos ensinamentos, pela educação e pelos exemplos.

À minha esposa Andrea, pelo incentivo, apoio e pela compreensão durante os vários momentos de ausência em prol da dedicação acadêmica.

À minha filha Isabella, luz da minha vida, que me alegra com cada sorriso.

Aos amigos André, Érico, Henrique, Humberto, Igor, Joffre, Juan, Maicon, Rafael, Renato, Sandro, Sérgio e Tiago, que durante todo o período do curso me apoiaram e ajudaram tanto moralmente quanto tecnicamente tornando o Laboratório de Redes um lugar excelente para se trabalhar. Sinto-me honrado em tê-los como amigos, senhores.

Ao grande Claudio, amigo incansável, sempre disposto a ajudar e incentivar a mim e a todos no Laboratório. Agradeço muito por tudo o que fez pelo andamento do trabalho, pelas revisões e pela ajuda na programação. Sua ajuda foi fundamental para o sucesso desta Dissertação.

À amiga Paula, pela sua dedicação, determinação e disposição na ajuda com todas aquelas simulações, revisões e programação. Fiquei muito feliz em conhecê-la e por ter tido a sorte de tê-la ao meu lado me ajudando de maneira tão competente e eficiente. Continue assim, pois seu futuro é brilhante!

Aos professores do PPGI pelas orientações e ensinamentos e, em especial, aos professores Paulo Aguiar, Flávia Delicato e Luíz Fernando Rust pelas correções e revisões dos trabalhos. A ajuda dos senhores foi fundamental para o meu sucesso.

Em especial às minhas orientadoras, professora Luci Pirmez e professora Silvana Rossetto, que durante todo o período do curso mantiveram-se sempre dispostas a orientar e ajudar, guiando-me durante todo o caminho. Professoras, o sucesso deste trabalho é um reflexo de toda a sua dedicação. Sinto-me orgulhoso de ter sido orientado pelas senhoras. Muito obrigado.

Ao Comandante Vianna (Diretoria de Comunicações e Tecnologia da Informação da Marinha), pelo seu apoio e pelas suas orientações durante o curso. O senhor me serve como um exemplo a seguir.

À Marinha do Brasil, pela oportunidade de aperfeiçoamento profissional e pessoal.

“Less is More.”
Ludwig Mies van der Rohe

“It makes all the difference whether one sees darkness through the light,
or brightness through the shadows.”
David Lindsay

RESUMO

SALMON, Helio Mendes. Sistema de Detecção de Intrusão Imuno-inspirado customizado para Redes de Sensores Sem Fio. Rio de Janeiro, 2011. Dissertação (Mestrado em Informática) - Programa de Pós-Graduação em Informática, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2011.

Redes de Sensores Sem Fio (RSSF) representam um novo paradigma de monitoração ambiental com muitas aplicações em potencial e riscos inerentes a sua forma de organização e tipo de comunicação. Como em qualquer outra rede, as informações transmitidas podem ser alvo de diversos tipos de ataques. Mecanismos que previnam estes ataques, tais como um Sistema de Detecção de Intrusão (SDI), podem ser empregados para proteger as RSSFs. Este trabalho propõe uma arquitetura genérica para um SDI bio-inspirado no Sistema Imunológico Humano, usando a Teoria do Perigo e o Algoritmo das Células Dendríticas como os mecanismos para a sua implementação. Esta arquitetura foi customizada para o contexto das RSSFs. Diversos experimentos foram realizados, tanto por simulação quanto em plataformas de sensores reais, a fim de se calibrar o SDI proposto e compará-lo com outro trabalho, que utiliza outra metodologia bio-inspirada de detecção. Também foi realizada uma comparação entre um cenário real e seu equivalente simulado, onde foi verificada a proximidade dos resultados obtidos entre ambos. Os resultados obtidos apresentaram-se melhores tanto com relação à eficiência da detecção de intrusos quanto com relação à quantidade de energia consumida pelo SDI proposto ao serem comparados com outro trabalho.

Palavras-chave: Rede de Sensores Sem Fio. Sistema de Detecção de Intrusão. Sistema Imunológico Artificial.

ABSTRACT

SALMON, Helio Mendes. Sistema de Detecção de Intrusão Imuno-inspirado customizado para Redes de Sensores Sem Fio. Rio de Janeiro, 2011. Dissertação (Mestrado em Informática) - Programa de Pós-Graduação em Informática, Universidade Federal do Rio de Janeiro, Rio de Janeiro, 2011.

Wireless sensor networks (WSN) represent a new paradigm for environmental monitoring supporting many potential applications, however offering risks inherent to its form of organization and type of communication. Like any other network, the information provided may be subject to various types of attacks. Attack prevention mechanisms, such as an Intrusion Detection System (IDS), can be employed to protect WSNs. This work proposes a generic architecture for an IDS Bio-inspired by the Human Immune System, using the Danger Theory and the Dendritic Cells Algorithm as the mechanisms for its implementation. The architecture was customized to the WSNs context, and several experiments were carried out by both simulation and real sensor platforms, for calibration of the proposed IDS and comparison with related work, which uses a different bio-inspired methodology for detection. We also compared a real scenario with its equivalent simulation, and demonstrated close results between the two. Results were better according to the efficiency of intrusion detection and according to the amount of energy consumed by the proposed IDS comparing to another work.

Keywords: Wireless Sensor Networks. Intrusion Detection System. Artificial Immune System.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

ACD	Algoritmo das células dendríticas
CD	Célula dendrítica
CIDF	*Common Intrusion Detection Framework*
CSM	*Costimulatory Molecules*
EB	Estação base
FN	Falsos Negativos
FP	Falsos Positivos
LN	Linfonodo
MCAV	*Mature Context Antigen Value*
NodeId	Número de identificação de um nó sensor na rede
PAMP	*Pathogenic Associated Molecular Patters*
PDR	*Packet delivery ratio*
RSSF	Rede de sensores sem fio
RSSI	*Received signal strength information*
SDI	Sistema de detecção de intrusão
Sensor-cd	Sensor executando o papel de célula dendrítica
Sensor-linfo	Sensor executando o papel de linfonodo
SIA	Sistema imunológico artificial
SIH	Sistema imunológico humano
VN	Verdadeiros Negativos
VP	Verdadeiros Positivos
FN	Falsos Negativos
FP	Falsos Positivos

Sumário

1 Introdução

Os recentes avanços na tecnologia de sistemas micro-eletro-mecânicos, nas comunicações sem fio e na eletrônica digital possibilitaram a construção de sensores de baixo custo que possuem tamanho reduzido, os quais, em grande número, permitem monitorar variáveis físicas e ambientais como temperatura, umidade, níveis de ruído e movimento de objetos com elevado grau de precisão. Aplicações de monitoramento de estruturas, mapeamento de recursos naturais, rastreamento e coordenação em diferentes contextos, são exemplos de uso deste tipo de rede sem fio (AKYILDIZ *et al.* 2002; DA SILVA *et al.* 2006).

No meio militar, em particular, as redes de sensores sem fio (RSSF) podem integrar parte dos sistemas militares de Comando, Controle, Comunicações, Computação, Inteligência, Vigilância, Reconhecimento e Mira (C^4ISRT em inglês). Quando equipadas com os sensores adequados, estas redes podem realizar, por exemplo: monitoramento de forças amigas, equipamento e munição; vigilância em campo de batalha; reconhecimento de forças inimigas e terreno; sistemas de mira; avaliação de danos em batalha; detecção e localização de atiradores escondidos (*snipers*); e detecção e reconhecimento de ataques nucleares, biológicos ou químicos (HUSSAIN *et al.* 2009; WINKLER 2008; SIMON *et al.* 2004; LOUREIRO *et al.* 2002; XUE e HASSANEIN 2006). Sistemas deste tipo já são comercializados por empresas do ramo (BAE SYSTEMS 2010).

Se por um lado as RSSFs trazem novas e amplas perspectivas para diversas aplicações, por outro lado trazem também uma série de desafios que ainda devem ser superados, propiciando um vasto campo para pesquisa.

Para produzir sensores pequenos e baratos, tornando-os economicamente viáveis para uma distribuição em larga escala e para a instalação em locais restritos, eles são desenvolvidos com capacidade de processamento e de memória limitados. Com uma fonte de energia também limitada e, geralmente, não substituível, torna-se um grande desafio mantê-los funcionando por alguns meses ou até mesmo por alguns dias (RAYMOND 2009). Aliado à limitação de recursos, as RSSFs estão sujeitas a vulnerabilidades associadas à comunicação sem fio e à organização *ad hoc* dessas redes e, em alguns cenários de aplicação, ao fato dos sensores serem depositados em áreas abertas, desprotegidas e muitas vezes hostis. Essas características tornam as RSSFs alvo de diferentes tipos de ataques, podendo comprometer a confiabilidade, a integridade e a disponibilidade de seus dados, assim como a vida útil da rede (DA SILVA *et al.* 2006; YICK *et al.* 2008; DATEMA 2005).

Em redes com tantas restrições como as RSSFs, a implementação de mecanismos de segurança é uma necessidade. Porém, a utilização de um mecanismo de segurança deve causar o mínimo de impacto nos recursos disponíveis da rede, devendo este impacto ser também alvo de estudos além da própria verificação da eficiência do mecanismo adotado. Uma das formas de lidar com as vulnerabilidades associadas às RSSFs é através de um Sistema de Detecção de Intrusão (SDI). Esses sistemas têm como finalidade detectar ataques de nós maliciosos e aplicar as contramedidas adequadas a fim de manter a operacionalidade da rede. Por razões tecnológicas, as necessidades de um SDI no âmbito das RSSFs são diferentes das redes cabeadas. A arquitetura de um SDI em RSSFs deve ser simples e altamente especializada, sendo capaz de analisar os protocolos específicos de uma RSSF, e empregar algoritmos de execução rápida que ocupem pouco espaço de memória e consumam pouca energia. Os alertas gerados pelo SDI devem chegar o mais rápido possível à estação base (EB), ou acionar mecanismos de defesa codificados nos próprios sensores. Finalmente, os sensores de uma RSSF devem ter a capacidade de trocar informações entre si de forma a alcançar um melhor desempenho na detecção de um intruso (ROMAN *et al.* 2005; ROMAN *et al.* 2006).

Entretanto, um problema associado aos SDIs são as falhas na detecção e os alarmes falsos que podem ocorrer com certa frequência, comprometendo a sua adoção. Uma forma de solucionar esse problema é empregando métodos de Inteligência Computacional de forma a tornar o SDI mais eficaz (SILVA 2009). Neste trabalho, foi adotada uma das frentes de pesquisa exploradas atualmente pela Inteligência Computacional: o Sistema Imunológico Artificial (SIA), que é uma solução computacional inspirada no Sistema Imunológico Humano (SIH). A semelhança entre os problemas existentes na área de segurança de computadores e o SIH constitui uma rica fonte de inspiração para o desenvolvimento de novos sistemas de detecção de intrusos biologicamente inspirados (HONG e YANG 2009; AICKELIN e CAYZER 2002).

O Sistema Imunológico Humano vem sendo estudado por teorias como a Seleção Negativa, a Seleção Clonal e a Rede Imunológica, chamadas de teorias clássicas, as quais se baseiam em características estruturais dos patógenos (os elementos invasores) que os distinguem das células do hospedeiro (o corpo humano). Esta diferenciação entre o que é do hospedeiro (próprio) e o que é alheio a este (não-próprio) permite ao hospedeiro a eliminação do patógeno. Porém, esta abordagem vem sendo questionada desde 1994 (MATZINGER 1994), quando foi introduzida a Teoria do Perigo, e continua sendo estudada pelos imunologistas (MATZINGER 2002; GREENSMITH *et al.* 2008). Em Matzinger

(MATZINGER 1994) são discutidos os mecanismos e as interações biológicas que fundamentam a Teoria do Perigo. Nesta teoria, as características próprias ou não-próprias dos antígenos não são consideradas como determinantes em sua classificação como invasores. A Teoria do Perigo leva em conta se os antígenos são perigosos ou não conforme a emissão de sinais produzidos pelas células danificadas dos tecidos. Uma das estratégias para implementar o processamento destes sinais de forma a indicarem a presença de anomalia no sistema é por meio do Algoritmo das Células Dendríticas (ACD), conforme introduzido por Greensmith *et al.* (GREENSMITH *et al.* 2005) e experimentado em redes de computadores.

A abordagem apresentada neste trabalho tem como diferencial a inspiração biológica no SIH, baseado no funcionamento da Teoria do Perigo e no ACD, para a construção de um SDI imuno-inspirado genérico para RSSFs. A técnica imuno-inspirada é considerada de próxima geração para os SDIs, objetivando capacitá-los com características do SIH, entre as quais se destacam: auto-organização, pois consegue combinar a funcionalidade de vários elementos distintos na detecção e eliminação de um invasor de forma automática; autonomia, pois é um sistema que funciona independentemente de ser comandado por outros sistemas; adaptabilidade, pois melhora sua eficiência com o tempo, identificando e combatendo doenças antes não conhecidas; robustez, pois possuem vários pontos de detecção de anomalias, suportando falhas eventuais em alguns pontos do sistema; e tolerantes, pois consegue identificar substâncias do próprio organismo impedindo uma reação contra ele mesmo.

1.1 Objetivo

Nesta dissertação é proposta uma arquitetura genérica para um SDI aplicado à RSSFs, usando a Teoria do Perigo e uma customização do ACD para estas redes. O SDI proposto foi simulado e implementado em uma RSSF real para a realização dos experimentos. Foi utilizada a linguagem de programação nesC e o sistema operacional TinyOS (LEVIS e GAY 2009) como ferramentas desta implementação.

Foi avaliada a qualidade da detecção de intrusos ao serem comparados os números de falsos positivos e de falsos negativos, comprovados por medições do desempenho deste SDI quanto à sua sensibilidade (taxa de acerto na detecção) e sua especificidade (taxa de acerto em relação a uma situação normal). O SDI proposto pode ser empregado em qualquer aplicação para RSSF, independente do tipo de aplicação ou de protocolos específicos de comunicação, clusterização ou controle de topologia. Além disso, o mesmo pode ser “ligado” ou “desligado” pela aplicação, caso necessário, quando, por exemplo, for mais importante

realizar o sensoriamento do que ter uma maior segurança, ou quando as reservas de energia estiverem baixas.

Quanto às contribuições deste trabalho buscou-se: (i) verificar a adequabilidade do ACD para RSSFs; (ii) customizar o ACD para melhor satisfazer às restrições de recursos das RSSFs, tornando-o genérico no sentido de ser independente do tipo de ataque; e (iii) implementar o SDI proposto segundo uma arquitetura descentralizada, onde os componentes biológicos são distribuídos entre vários sensores, cooperando entre si para a identificação de um ataque.

1.2 Organização do trabalho

O restante deste trabalho está organizado da seguinte forma. No Capítulo 2 discutem-se os conceitos básicos que descrevem em linhas gerais a base de conhecimento para este trabalho. No Capítulo 3 são destacados os trabalhos relacionados. No Capítulo 4 é apresentada a arquitetura computacional do Sistema de Detecção de Intrusão proposto. No Capítulo 5 são apresentados os experimentos realizados e a análise dos resultados obtidos. Finalmente no Capítulo 6 são tecidas as conclusões finais e as propostas de trabalhos futuros.

2 Conceitos Básicos

Neste capítulo são descritos os conceitos básicos que sustentam os temas abordados e são necessários para o entendimento do trabalho proposto.

2.1 Redes de Sensores Sem Fio

As redes de sensores sem fio (RSSFs) constituem uma nova área de pesquisa no domínio da computação distribuída e vêm sendo pesquisadas com grande interesse nos últimos anos. Uma RSSF é um tipo de rede *ad hoc* com características e requisitos específicos. O objetivo principal de uma RSSF é sensoriar, de forma distribuída, informações ambientais tais como, temperatura, umidade, luminosidade, vibrações, dentre outras, e fornecê-las a aplicações clientes. Os dados são coletados por diversos sensores distribuídos pelo ambiente e encaminhados a nós de saída da RSSF, chamados de sorvedouros ou estações base (EBs). As EBs estão, normalmente, conectadas a equipamentos com maior capacidade de processamento e armazenamento, dedicados ao processamento dos dados recebidos dos sensores (DELICATO 2005; YICK *et al.* 2008).

As RSSFs podem utilizar diversos modelos de entrega de dados, classificando-as como: (i) contínua, quando os dados possuem uma periodicidade de leitura e encaminhamento para a EB; (ii) dirigidas a eventos, quando o sensoriamento seguido de encaminhamento de dados para o sorvedouro é acionado apenas quando uma grandeza ultrapassa um limiar, como por exemplo, o controle de temperatura em um alarme de incêndio; (iii) iniciadas pelo observador, quando o usuário daquela RSSF comanda uma leitura pelos sensores; ou (iv) híbridas, quando apresentam mais do que uma das característica anteriores.

Uma vez definido o modelo de entrega de dados a ser utilizado pela rede, deve ser escolhido o protocolo mais adequado que encaminhe os dados sensoriados dos sensores até a EB, diretamente ou via múltiplos saltos. Os protocolos podem ser classificados em: (i) baseados em comunicação direta; (ii) planos; e (iii) hierárquicos. Nos protocolos baseados em comunicação direta, como o próprio nome sugere, os sensores enviam seus dados diretamente para a EB. Nos protocolos planos, todos os nós da RSSF têm a mesma probabilidade de participar de um roteamento de dados para a EB quando sensores mais distantes precisarem informar seus dados sensoriados. Nos protocolos hierárquicos, os sensores são classificados com papéis diferentes: sensores líderes de um *cluster* e sensores membros do *cluster*. A

clusterização fornece à RSSF mais escalabilidade e maior tolerância a falhas (DELICATO 2005).

Adicionalmente, as RSSF podem ser classificadas quanto a: (i) composição, que define uma rede como sendo homogênea, quanto todos os sensores são iguais entre si, ou heterogênea, caso contrário; (ii) organização, que define uma rede como plana, quando todos os sensores desempenham o mesmo papel, ou hierárquica, quando existem agrupamentos (*clusters*) de um ou mais níveis com sensores desempenhando papéis diferentes; (iii) mobilidade, quando os nós sensores são móveis ou estacionários; e (iv) densidade, sendo a rede irregular quando os nós estão espalhados desordenadamente em uma região de interesse, ou balanceada caso contrário. Podendo ser classificada também como densa, caso a concentração de nós por unidade de área seja alta ou, esparsa, caso contrário (DELICATO 2005).

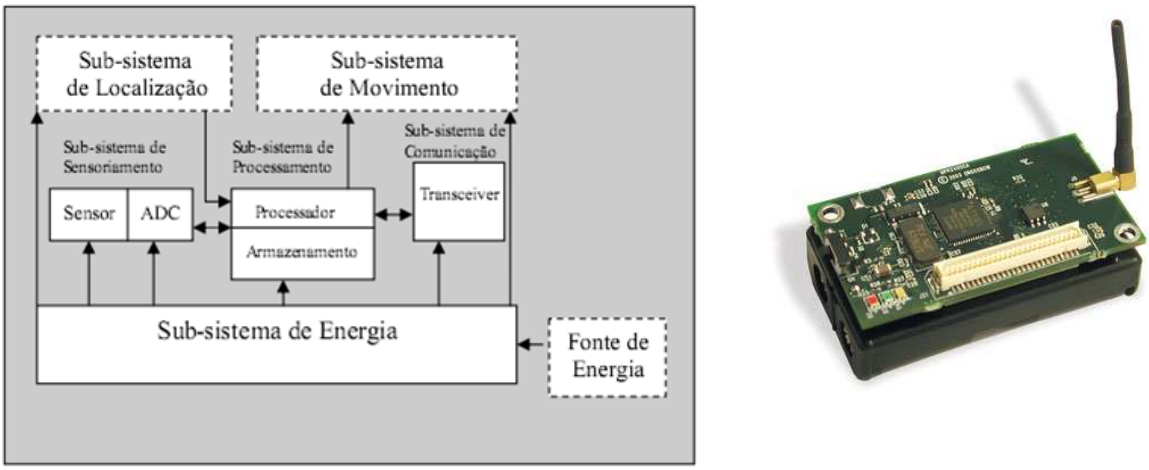
Um ponto a ser destacado é quanto à economia de energia dos sensores, um requisito fundamental em RSSFs, o qual demanda o uso de técnicas particulares visando o menor consumo de energia. Um exemplo é a técnica de programar os sensores de forma a revezar intervalos de atividade e de inatividade dos componentes de rádio e de processamento. Os sensores e a EB podem transmitir e receber dados pelo meio sem fio. Alguns autores constataram que a energia consumida pela transmissão de um bit de informação foi aproximadamente igual à quantidade de energia consumida por até 1000 instruções de processamento (DIETRICH e DRESSLER 2009).

Um sensor típico é composto por quatro subsistemas principais: (i) sensoriamento; (ii) processamento; (iii) comunicação; e (iv) fonte de energia, conforme mostra a Figura 1(a) (DELICATO 2005). Os subsistemas de Localização e de Movimento são opcionais, estando presentes em alguns tipos de sensores. A Figura 1(b) mostra o sensor modelo MICAz da Crossbow (CROSSBOW 2010), utilizado neste trabalho.

O subsistema de sensoriamento é composto pelos dispositivos de sensoriamento e pelo conversor de sinais analógico para digital (ADC). Um sensor pode possuir mais de um tipo de dispositivo de sensoriamento. O subsistema de processamento está, em geral, associado à unidade de armazenamento local, sendo responsável pela execução dos protocolos de comunicação, pelo processamento de dados, pelo controle dos sensores e pela gerência colaborativa entre os nós sensores. O subsistema de comunicação é composto por algum dispositivo de rádio, sendo responsável pela transmissão e recepção de dados entre os sensores e entre estes e o sorvedouro (DELICATO 2005). A fonte de energia dos sensores é

composta por uma bateria de tamanho reduzido. O subsistema de Localização é implementado por uma funcionalidade de GPS e o subsistema de Movimentação é implementado por um dispositivo que permite ao sensor movimentar-se no meio.

Além dos componentes de hardware acima descritos, as plataformas de sensores podem incluir um sistema operacional rudimentar. Este sistema é responsável por gerenciar de maneira eficiente a operação do sensor. Um exemplo deste tipo de sistema operacional é o TinyOS (LEVIS e GAY 2009).



(a) (b)

Figura 1. (a) Arquitetura de um sensor e (b) sensor MICAz da Crossbow.

2.2 Sistemas de Detecção de Intrusão

Os Sistemas de Detecção de Intrusão (SDI) são responsáveis por identificar, relatar e combater atividades maliciosas provenientes tanto de elementos externos quanto de elementos internos ao sistema (BARBOSA 2000; TIMOFTE 2008).

O *Common Intrusion Detection Framework* (CIDF) sugere uma padronização de SDI. Segundo o CIDF (DEBAR *et al.* 1999; BARBOSA 2000; GARCÍA-TEODORO *et al.* 2008), o modelo sugerido para projetos de um SDI deveria possuir os seguintes componentes: (i) gerador de eventos (E-BOX); (ii) analisador de eventos (A-BOX); (iii) banco de dados (B-BOX); e (iv) contramedidas (C-BOX), conforme ilustrado na Figura 2.

O componente Gerador de Eventos captura os pacotes da rede e entrega ao componente Analisador de Eventos, para o processamento das informações, e para o componente Banco de Dados, para armazenar estas informações e manter um histórico. O componente Analisador de Eventos é o núcleo do SDI, sendo responsável por identificar se as informações que chegaram

do componente Gerador de Eventos são ou não um ataque. Este componente também pode armazenar as informações de ataques identificados no componente Banco de Dados. O componente Contramedidas é responsável por receber uma informação do componente Analisador de Eventos avisando que a rede está sob ataque. Este componente pode tomar diversas ações, como comunicar-se com outros SDI, acionar alarmes e avisar ao administrador do sistema (BARBOSA 2000).

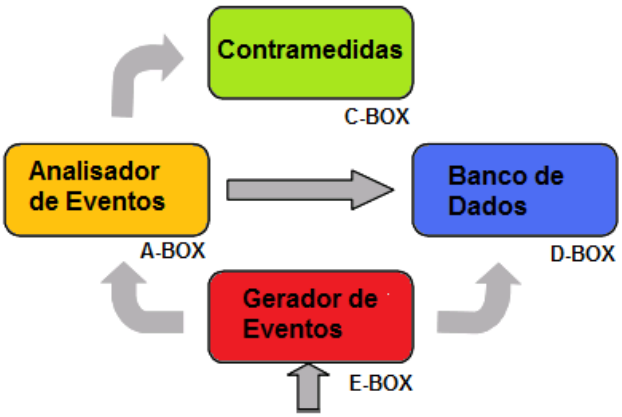


Figura 2. Interação dos componentes do modelo CIDF (BARBOSA 2000).

Um SDI deveria ser capaz de identificar possíveis incidentes de segurança, guardar informações sobre estes incidentes, tentar realizar uma ação de forma a parar o incidente e reportá-los ao administrador da rede (TIMOFTE 2008). Esta definição encaixa-se perfeitamente nas RSSFs, onde a mera identificação do atacante não resulta na economia de recursos da rede. É preciso que o SDI também seja capaz de realizar algum tipo de medida (contramedida) que anule ou reduza o efeito do ataque, protegendo os recursos da rede, como por exemplo a ativação de criptografia ou o desligamento do rádio por um período de tempo.

2.2.1 Classificação dos SDI

Um SDI pode ser classificado segundo as seguintes características (DA SILVA 2005; KAUR e SINGH 2010): (i) método de detecção; (ii) comportamento na detecção; (iii) tempo de detecção; (iv) processamento; (v) fonte de auditoria; (vi) local de processamento de dados; e (vii) local da coleta de dados. A Figura 3 ilustra essas características.

Método de detecção: tradicionalmente, é dividido em duas abordagens: a detecção por anomalias (*anomaly detection*), onde o SDI identifica intrusões como um comportamento não-usual que difere do comportamento normal esperado; e a detecção por mau uso (*misuse detection*), onde o SDI observa eventos que combinem com padrões pré-definidos de ataques

conhecidos, os quais são armazenados em uma base de dados (DA SILVA 2005). O SDI proposto neste trabalho utiliza a abordagem de detecção por anomalias, pois em RSSF o custo de se manter uma base de dados em um sensor é maior do que as informações que definem um comportamento normal.

Comportamento na detecção: indica como o SDI reagirá ao detectar um intruso. Se o SDI apenas gerar alertas, ele será um SDI passivo. Caso ele também possua a capacidade de reagir, ele será considerado um SDI ativo (DA SILVA 2005). Neste trabalho, o SDI proposto é ativo, pois a RSSF deve ser defendida do ataque a fim de preservar seus recursos.

Tempo de detecção: indica se a análise dos dados é feita em tempo real, próximo de tempo real, ou com algum atraso. No caso das RSSF é interessante que a detecção seja feita em tempo real, ou próxima do tempo real (DA SILVA 2005). O SDI proposto neste trabalho gera alarmes próximos do tempo real.

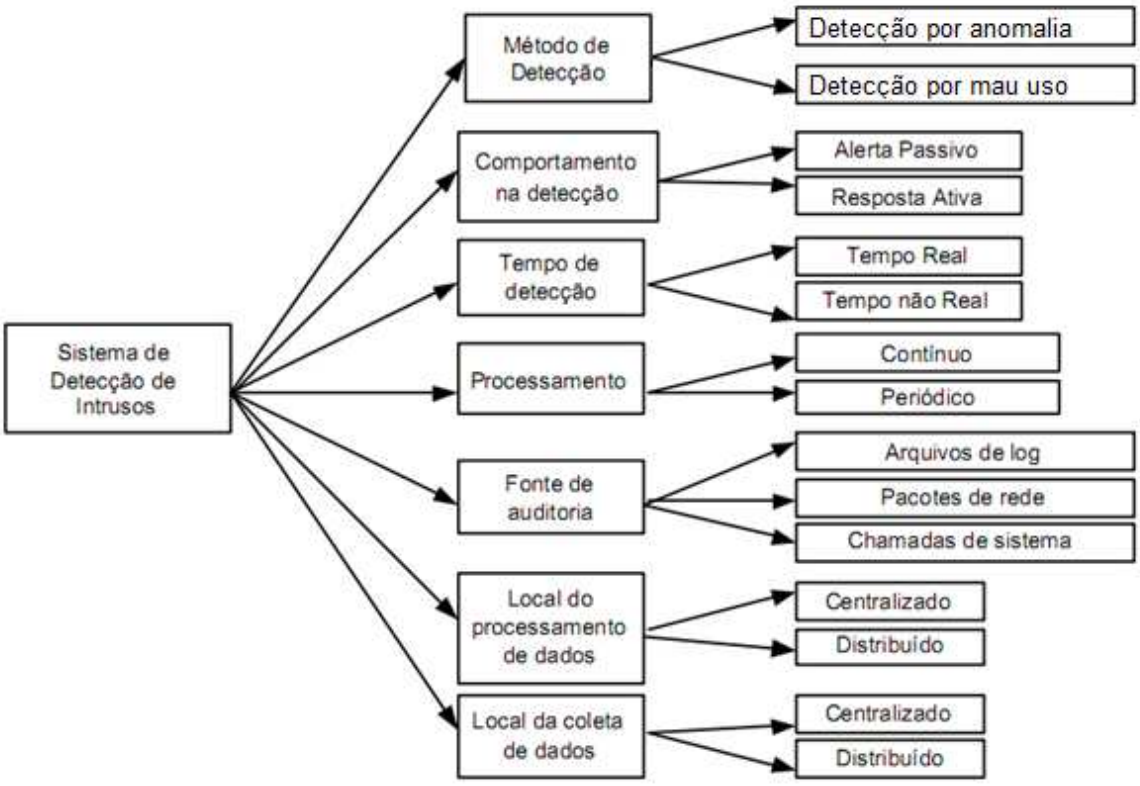


Figura 3. Classificação dos SDI (DA SILVA 2005).

Processamento: os dados podem ser analisados de forma contínua ou por blocos de dados em um intervalo regular. Este conceito afeta diretamente o tempo de detecção do intruso. É interessante que em uma RSSF a detecção seja feita em tempo real, pois quanto

antes um ataque for identificado e combatido menor será o prejuízo da RSSF. No SDI proposto, o processamento é contínuo.

Fonte de auditoria: indica o tipo de dados que o SDI analisa. As fontes podem ser arquivos de *log*, pacotes da rede ou chamadas de sistema. O SDI proposto analisa pacotes da rede, pois não existe espaço suficiente de armazenamento nos sensores para *logs*.

Local de processamento: a detecção pode ser realizada em um ponto central de coleta de dados ou de forma distribuída. No SDI proposto, o processamento é distribuído entre papéis diferentes implementados nos sensores da rede, o que contribui para a redução do processamento em um único sensor.

Coleta de dados: os dados podem ser coletados por um ponto central ou de forma distribuída. No SDI proposto, a coleta é distribuída uma vez que o SDI é distribuído.

2.3 Segurança em Redes de Sensores Sem Fio

Uma rede ou um sistema de comunicação pode ser considerado seguro se atender aos seguintes requisitos de segurança: (i) **confidencialidade**, que é a garantia de proteção de uma informação armazenada ou transmitida quanto a divulgação a uma entidade não autorizada; (ii) **autenticação**, que é o método de comprovação da identidade de um parceiro de comunicação ou autor de mensagem; (iii) **integridade**, que é a garantia de proteção de informações armazenadas ou em trânsito contra a modificação por uma entidade não autorizada; (iv) **irretratabilidade**, que é a garantia de que determinada entidade que tenha transmitido ou recebido uma mensagem não alegue que não a transmitiu ou não a recebeu; (v) **disponibilidade**, que representa a garantia de que determinado recurso ou o próprio sistema esteja sempre disponível para as entidades autorizadas; (vi) **controle de acesso**, que representa a garantia de que somente as entidades autorizadas tenham acesso a um determinado recurso e que essas autorizações não sejam modificadas indevidamente; e (vii) **auditoria**, que é a garantia do armazenamento de informações sobre a utilização de recursos do sistema (VIANNA 2006, SU *et al.* 2007).

Corroborando com os requisitos anteriores, para que um sistema seja considerado seguro, a segurança deve estar integrada em todos os aspectos do projeto de um sistema (PERRIG *et al.* 2004), o que se torna ainda mais evidente no contexto de uma RSSF. Nas RSSFs, a garantia da segurança pode ser dificultada por vários fatores, tais como: (i) restrições de recursos (energia, memória e processamento), afetando o requisito de disponibilidade; (ii) possibilidade de captura ou destruição dos sensores, o que pode afetar

todos os requisitos mencionados anteriormente; (iii) utilização de comunicação sem fio, afetando os requisitos de confidencialidade e controle de acesso; e (iv) necessidade de garantir a escalabilidade em redes com uma grande quantidade de sensores, o que afeta diretamente o requisito de disponibilidade (SHI *et al.* 2004). Estas restrições inerentes às RSSFs tornam inviável a aplicação direta das soluções de segurança propostas para redes sem fio de computadores tradicionais (DA SILVA 2005).

Em RSSFs surge um novo requisito de segurança além dos requisitos acima relacionados: o fornecimento de **dados atualizados** pelos sensores (SU *et al.* 2007). Este requisito representa a importância da atualização dos dados lidos pelos sensores e a entrega desses dados à EB dentro de um tempo que não invalide a medição. A demora no encaminhamento de um dado pode resultar em inconsistências do sistema, pois as informações ficarão desatualizadas.

2.3.1 Ataques em Redes de Sensores Sem Fio

RSSFs são vulneráveis a diversos tipos de ataques que buscam desde a captura de informações trafegando na rede até a desativação da mesma. Dentre os ataques mais comuns, pode ser citado o *Denial-of-Sleep*, que busca acabar com os recursos de energia dos sensores e, conseqüentemente, com a disponibilidade da rede por meio de consumo excessivo de energia pela recepção de mensagens inúteis pelos sensores e pelas sucessivas tentativas de retransmissão devido à interferências. Uma maneira de evitar este tipo de ataque seria utilizando espalhamento espectral para a codificação dos sinais. Porém, os rádios com suporte a codificação por espalhamento espectral são mais complexos, mais caros e consomem mais energia, inviabilizando seu uso em RSSF (MARGI *et al.* 2009).

Outra vulnerabilidade, presente em alguns tipos de RSSFs, é oriunda do fato dos sensores ficarem em locais sem segurança física ou não monitorados, permitindo formas de *node tampering*, ou seja, um intruso poderia danificar um sensor, de modo que este não efetuasse suas funções de coleta e/ou roteamento de dados, prejudicando o funcionamento da aplicação em execução na RSSF. Um inimigo poderia ainda substituir um sensor por outro malicioso para gerar ataques ou obter informações da rede. Uma terceira possibilidade seria a extração de informações armazenadas em um sensor capturado, permitindo a um atacante obter chaves de criptografia ou autenticação. Para evitar a vulnerabilidade *node tampering* são necessários circuitos ou mecanismos para proteção dos dados, capas de proteção ou selos (MARGI *et al.* 2009).

Considerando que em uma RSSF todos os sensores são também roteadores de mensagens, surgem ataques que buscam alterar ou impedir este roteamento. Dentre os mais comuns pode ser citado o Buraco Negro (*Black Hole*), onde um sensor malicioso é introduzido na rede e passa a informar que a melhor rota a vários destinos é passando por ele. Todas as mensagens podem então ser descartadas ou alteradas, afetando a integridade da rede. Outro ataque similar ao Buraco Negro é o Encaminhamento Seletivo (*Selective Forwarding*), onde algumas mensagens são re-encaminhadas e outras não, podendo esta seleção ser aleatória ou baseada em algum critério previamente estabelecido (KARLOF e WAGNER 2003).

Outro ataque possível é o de Inundação da rede (*Flooding*), onde um sensor malicioso inunda a rede com mensagens falsas, congestionando e gerando consumo excessivo de energia pela rede. Além disso, o envio de falsas mensagens de roteamento poderia criar rotas erradas, causando descarte de pacotes e mais desperdício de energia (MARGI *et al.* 2009).

Com o ataque de Buraco de Verme (*Wormhole*) dois nós maliciosos localizados em diferentes locais da rede criam um túnel entre si, utilizando frequência de rádio diferente da utilizada na RSSF. Por meio deste túnel, pacotes de um lado da rede são enviados para o outro lado, fazendo com que nós em diferentes localidades acreditem serem vizinhos, e, conseqüentemente, criando problemas no roteamento (MARGI *et al.* 2009).

Finalmente, outro ataque comum é o chamado Nós irmãos (*Sybil Nodes*), onde um sensor malicioso assume múltiplas identidades (identificador do nó na rede). Este ataque permite que os “nós irmãos” sejam usados para atacar protocolos de armazenamento distribuído, agregação de dados, eleições e algoritmos de detecção de mau-comportamento (KARLOF e WAGNER 2003).

2.3.2 Sistemas de Detecção de Intrusão Aplicados a Redes de Sensores Sem Fio

Em uma RSSF o SDI poderia ser localizado na EB, sendo, portanto centralizado. Porém, esta abordagem resultaria em um drástico aumento na quantidade de mensagens transmitidas pelos sensores para a EB. Com um SDI distribuído, além de possibilitar uma verificação mais abrangente da rede, é possível diminuir a quantidade de mensagens transmitidas, pois estas passam a ser previamente processadas pelos nós (WANG e TSENG 2006; KAUR e SINGH 2010). Neste contexto, a cooperação entre os sensores torna-se essencial para a economia de energia e para uma detecção eficiente.

Em Tiwari *et al.* (TIWARI *et al.* 2009) e Ioannis *et al.* (IOANNIS *et al.* 2007), são levantados os seguintes requisitos para um SDI em RSSFs: (i) realizar auditoria localizada, devido à falta de um ponto centralizador responsável pela coleta de dados de auditoria; (ii) utilizar recursos mínimos, uma vez que RSSFs possuem recursos limitados de comunicação, armazenamento e energia; e (iii) ser distribuído por natureza, ou seja, a coleta e análise dos dados de auditoria e a execução do algoritmo de detecção devem ser feitos em vários pontos na RSSF (KAUR e SINGH 2010) .

Em RSSFs, não apenas a detecção de um nó malicioso é importante, mas também a aplicação de contramedidas efetivas de forma a combater ou minimizar as ações realizadas pelos ataques, garantindo a melhor utilização dos recursos escassos deste tipo de rede (BROWNFIELD *et al.* 2005; CHEN *et al.* 2009).

2.4 Sistema Imunológico Humano

Atualmente, vem surgindo um grande interesse pelo estudo do Sistema Imunológico Humano (SIH). Imunologistas e profissionais da área médica vêm tentando compreender melhor este sistema para poder combater mais eficientemente doenças infecciosas, doenças auto-imunes e outros problemas de saúde. Biólogos e imunologistas vêm tentando simular e observar tais fenômenos em laboratórios por meio de modelos teóricos de forma a conseguir reproduzir os mecanismos de um sistema imunológico. Mais recentemente, pesquisadores das áreas de engenharia e computação começaram a se interessar pela imunologia, tentando simular seus mecanismos para criar sistemas artificiais que solucionem os problemas de suas áreas específicas (DE CASTRO 2001).

O SIH é composto por órgãos, tecidos e células que têm como função proteger o organismo de agentes externos (HOFMEYR e FORREST 2000). Os agentes externos (ou patógenos), como as bactérias, vírus e germes podem causar doenças em um determinado indivíduo (o hospedeiro). Os patógenos são identificados pelo organismo por meio dos antígenos, que são pequenas partes destes agentes (GREENSMITH 2007).

O SIH é extremamente complexo, sofisticado e vem se desenvolvendo há milhares de anos com a evolução das espécies. Ele pode reconhecer uma grande quantidade de patógenos e, dependendo do tipo de agente invasor, é capaz de produzir secreções e células que identificam e eliminam os patógenos do organismo ou neutralizam as suas ações. Um patógeno pode atacar o organismo em diferentes lugares e o sistema imune, sendo um sistema distribuído, consegue identificar o ataque e iniciar as medidas de defesa.

O mecanismo de defesa do corpo humano é dividido em três níveis (Figura 4): o primeiro é o nível físico, representado pela pele. A pele é responsável por impedir a entrada de diversas substâncias e agentes nocivos ao organismo. O segundo nível são as barreiras bioquímicas como o suor, a saliva e as secreções produzidas por alguns órgãos, os quais tornam impróprio o ambiente para a manutenção da vida de agentes patógenos. Por último, temos o SIH que pode ser dividido em duas camadas: o Sistema Imunológico Inato e o Sistema Imunológico Adaptativo (SILVA 2009).

O Sistema Imunológico Inato destaca-se por sua natureza congênita que, apesar de possuir uma capacidade limitada de identificação de patógenos, age de forma rápida e efetiva contra invasores. Esta camada do sistema imunológico é composta pelas células apresentadoras de antígenos, entre elas, os macrófagos e as células dendríticas, tipos de células brancas do sangue que são as responsáveis pela ativação do sistema imune e pelas respostas às invasões. Os antígenos são apresentados por estas células apresentadoras em órgãos chamados linfonodos.

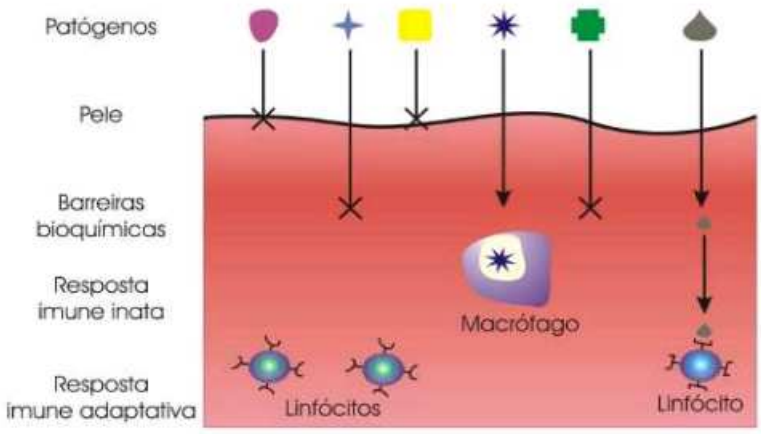


Figura 4. Camadas do Sistema Imunológico Humano (SILVA 2009).

Um linfonodo é um órgão pertencente ao SIH, mais especificamente um gânglio, atuando como uma região de convergência de um extenso sistema de vasos que coletam fluidos dos tecidos e para onde as células dendríticas migram. Existem diversos linfonodos espalhados pelo corpo humano. Um linfonodo recebe os antígenos das células dendríticas e os apresenta às células B e T do SIH. Existem vários destes órgãos espalhados por todo o corpo humano e é neste órgão que ocorre a ativação da resposta imune adaptativa.

O Sistema Imunológico Adaptativo é capaz de identificar patógenos desconhecidos, guardando uma “memória” daquele patógeno para que seja identificado caso apareça uma

próxima vez. Apesar de suas respostas serem especializadas por patógenos e possuírem um efeito mais duradouro, sua atuação é mais lenta se comparada ao sistema inato. O sistema adaptativo é composto pelas células B e pelas células T, as quais são as responsáveis por produzir os anticorpos. Os anticorpos irão de fato combater os patógenos.

O SIH vem sendo estudado por teorias como a Seleção Negativa, a Seleção Clonal e a Rede Imunológica, chamadas de teorias clássicas, as quais se baseiam em características estruturais dos patógenos que os distinguem das células do hospedeiro. Esta diferenciação entre o que é do hospedeiro (próprio) e o que é alheio a este (não-próprio) permite ao hospedeiro a eliminação do patógeno. Porém, esta abordagem vem sendo questionada desde 1994 (MATZINGER 1994), quando foi introduzida a Teoria do Perigo, e continua sendo estudada até hoje pelos imunologistas (MATZINGER 2002; GREENSMITH *et al.* 2008). Nesta teoria, as características próprias ou não-próprias dos antígenos não são consideradas como determinantes em sua classificação como invasores. A Teoria do Perigo leva em conta se os antígenos são perigosos ou não conforme a emissão de sinais de perigo pelas células danificadas dos tecidos.

2.5 Teoria do Perigo

A Teoria do Perigo popularizou-se na área de Computação com os trabalhos de Aickelin e Cayzer em 2002 e 2003 (AICKELIN e CAYZER 2002; AICKELIN *et al.* 2003), os quais contestaram as teorias clássicas de antígenos próprios e não-próprios para SDI baseado em SIA, substituindo-as pelas idéias apresentadas na Teoria do Perigo. Também apresentaram as possíveis aplicações de SDI baseados na Teoria do Perigo na detecção de anomalias. SDIs que utilizam esta abordagem são considerados de próxima geração (AICKELIN *et al.* 2003).

Em Aickelin *et al.* (AICKELIN *et al.* 2003), os autores passam a discriminar os sinais de perigo em duas classes diferentes: aqueles produzidos pelo organismo (endógenos) e aqueles produzidos por entidades externas (exógenos). As duas classes de sinais compartilham a mesma característica: ativar as células apresentadoras de antígenos conduzindo a uma resposta imune. Este trabalho serviu como base para outros trabalhos que utilizaram o ACD, pois permitiu fazer a ligação entre a Teoria do Perigo, que utiliza um “sinal de perigo genérico” (sinal único), e o ACD, que utiliza os sinais perigosos internos (necrose), os sinais regulatórios (apoptose) e os sinais de perigo externo para gerar um “sinal de perigo composto”.

Neste trabalho, a escolha da Teoria do Perigo em detrimento das teorias clássicas do SIH se deve a três razões principais. Primeiro devido ao fato das teorias clássicas classificarem os antígenos como próprios e não-próprios. No SIH alguns antígenos classificados como não-próprios não devem ser combatidos, como, por exemplo, a comida ingerida e as bactérias da flora intestinal, enquanto que outros antígenos classificados como próprios, mas que causam mal ao organismo, deveriam ser eliminados do sistema, como as células cancerígenas, por exemplo. O fato da Teoria do Perigo considerar um sinal de perigo ao invés das características próprias e não-próprias na identificação de um elemento invasor torna-a mais interessante do que as teorias clássicas, pois possibilita que esses sistemas sejam capazes de detectar novos tipos de ataques (HONG e YANG 2009), relacionando o sinal de perigo ao elemento (antígeno) causador deste sinal. A segunda razão diz respeito às mudanças que um organismo sofre durante seu ciclo de vida. Com o passar do tempo, o que é definido como próprio pode mudar, como é o caso das mutações, da puberdade e do envelhecimento. Esta característica de adaptação está presente na Teoria do Perigo tornando-a uma técnica capaz de lidar com o dinamismo e a natureza complexa de um sistema computacional. Finalmente, a terceira razão está ligada ao fato de que nas teorias clássicas as comparações realizadas entre seqüência de *strings*, as quais representam as características próprias e não-próprias dos antígenos (AICKELIN 2003), fazem uso de uma base de dados que aumenta com o passar do tempo, podendo ir de encontro com restrições de recursos de memória e processamento das RSSF, somando-se a estes a necessidade constante de se manter a base de dados atualizada.

Conforme Matzinger (MATZINGER 1994), o modelo proposto leva em conta dois sinais para a ativação do Sistema Imunológico: a presença de antígenos (pequenas partes dos patógenos); e a presença de um sinal que representa perigo às células do organismo. A ocorrência de ambos indicaria a existência de uma invasão. Em 2005, com o trabalho de Greensmith *et al.* (GREENSMITH *et al.* 2005), foi introduzido uma nova visão para o sinal de perigo. Este sinal passou a ser uma composição de quatro tipos de sinais que indicariam a presença de perigo: (i) sinais de perigo, quando as células sofrem necrose (morte celular não programada); (ii) sinais seguros, quando as células sofrem apoptose (morte celular programada); (iii) sinais de PAMP (*Pathogenic Associated Molecular Patters*), que são substâncias produzidas especificamente por microorganismos e não pelo organismo hospedeiro; e (iv) inflamação, que é um sinal com a capacidade de amplificar os efeitos dos outros sinais. A presença de moléculas de PAMP é um indicador da presença de uma entidade

extra-organismo e a inflamação representa o aumento do fluxo de sangue e o aumento da temperatura em uma área afetada por uma invasão, influenciando todos os sinais (GREENSMITH *et al.* 2008).

O principal elemento da Teoria do Perigo, responsável pela captura de antígenos e sua futura classificação por meio de um "sinal de perigo", são as células apresentadoras de antígenos como as células dendríticas.

2.5.1 Células Dendríticas

As células dendríticas (CDs) funcionam como "investigadores da cena do crime" cuja função é coletar antígenos de patógenos e de células dos tecidos. Elas são consideradas um dos maiores controladores do SIH, influenciando e orquestrando a resposta deste sistema e servindo como interface entre o sistema imunológico inato e o adaptativo.

As CDs coletam os antígenos nos tecidos e em caso de invasão, ou seja, na presença de células danificadas, tal área é denominada de Região de Perigo. Nesta região, as células dendríticas são ativadas pelos sinais emitidos pelas células dos tecidos que estão sofrendo danos causados por patógenos.

As CDs apresentam-se em três possíveis estados de maturação: imatura, semi-madura ou madura (MATZINGER 1994). O estado imaturo representa o estado inicial destas células, quando ainda não foram expostas a antígenos e sinais de perigo. As células dendríticas imaturas residem nos tecidos. Durante seu tempo de vida, uma CD fica coletando antígenos e sinais e, ao atingir um determinado limiar de migração, a CD migra para o linfonodo. No linfonodo, a CD muda seu estado de imaturo para maduro, caso os sinais coletados indiquem uma anomalia, ou de imaturo para semi-maduro, indicando uma situação normal. Além disso, as CDs apresentam os antígenos coletados durante seu estado imaturo às células B e às células T. Para uma CD migrar para o estado maduro ela deverá ter sido exposta mais a sinais de perigo e PAMP do que a sinais seguros. O oposto é observado para o estado semi-maduro. Várias células passam por este processo, e os antígenos apresentados por elas são classificados como seguros, quando apresentados por células semi-maduras, ou perigosos, quando apresentados por células maduras (GREENSMITH *et al.* 2005).

No SIH, as CDs não realizam sua função isoladamente. Existe uma população de CDs que reside nos tecidos e cada uma delas captura antígenos e sinais de entrada, migrando em tempos diferentes para o linfonodo. A multiplicidade de CDs é fundamental, pois são necessárias várias CDs apresentando análises sobre um mesmo tipo de antígeno para causar

uma resposta do SIH. Assim, o SIH torna-se tolerante a erros e robusto, pois a classificação errônea feita por uma CD não é suficiente para estimular uma reação indevida do SIH. Uma vez que as CDs informam ao linfonodo sobre a presença de um invasor, as células B e as células T são ativadas, ficando responsáveis pela produção de anticorpos específicos para aquele patógeno. Na Figura 5, ilustra-se o modelo de uma CD, seus possíveis estados de maturação e os sinais que influenciam a mudança entre os estados.

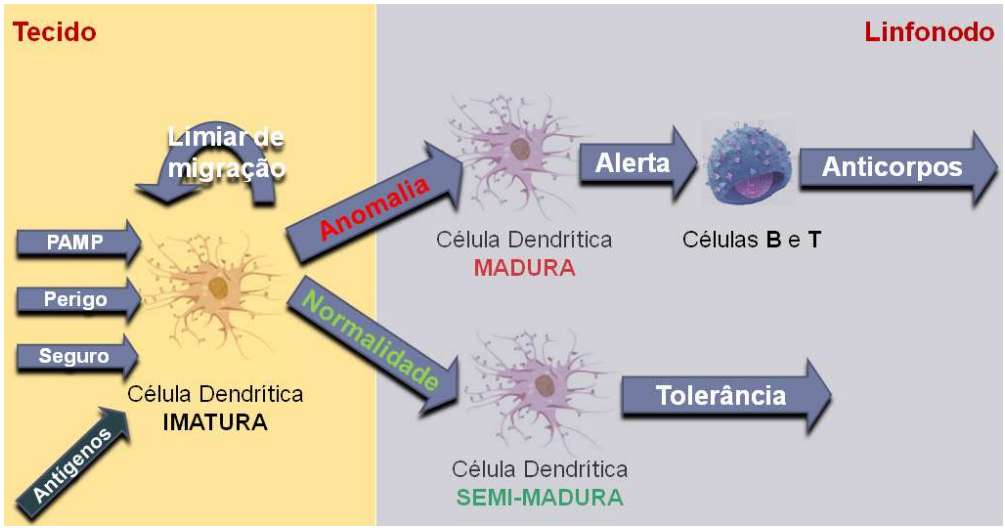


Figura 5. Modelo abstrato da diferenciação das células dendríticas (baseado em GREENSMITH *et al.* 2008).

2.6 Sistemas Imunológicos Artificiais

Os Sistemas Imunológicos Artificiais (SIA) são sistemas imuno-inspirados e representam uma das áreas de pesquisa da Inteligência Computacional (Figura 6).

A modelagem de um sistema bio-inspirado deve seguir os seguintes passos: (i) observar o sistema biológico que se quer modelar; (ii) entender os detalhes desse sistema; (iii) criar um algoritmo que reflete o sistema observado; e (iv) implementar o algoritmo em alguma linguagem de programação (GREENSMITH *et al.* 2008).

Os SIAs vêm sendo especialmente estudados devido ao fato de poderem ser empregados em diversas áreas da Computação e da Engenharia. Dentre as linhas de pesquisas encontradas na literatura destacam-se a aplicação de SIA no reconhecimento de padrões, em sistemas para detecção de falhas e anomalias, em métodos de buscas e otimização, em sistemas autônomos descentralizados, em abordagens para vida artificial, em segurança de sistemas de informação, no aprendizado de máquina e na mineração de dados (*data mining*) (DE CASTRO 2001; BACHMAYER 2008; HART e TIMMIS 2008; KIM *et al.* 2007).

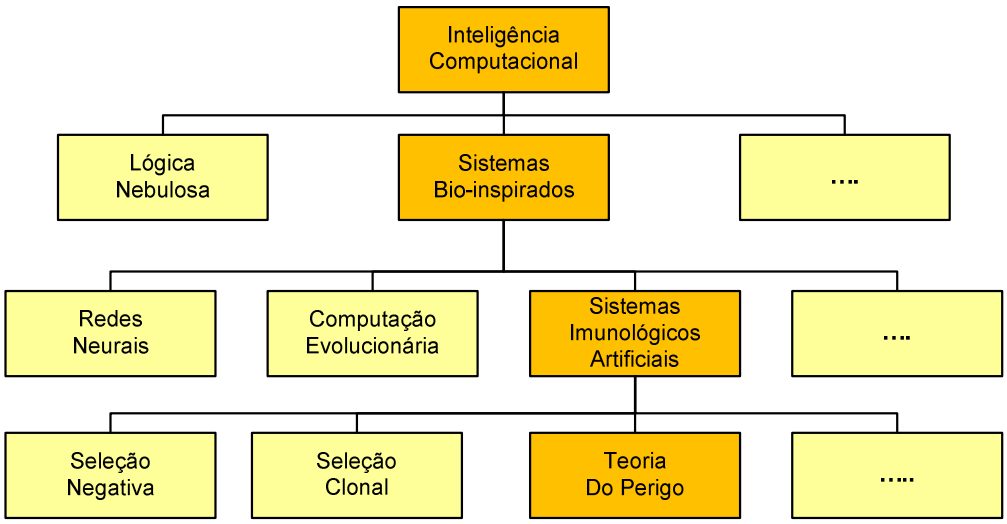


Figura 6. Inteligência Computacional e suas subdivisões (DASGUPTA 2006).

Dentre todas estas áreas de pesquisa a que vem se destacando mais é a de segurança computacional e, especialmente para as RSSFs, os SIAs parecem ser um paradigma promissor para a concepção de SDIs devido a sua baixa carga computacional (MAZHAR e FAROOQ 2008), paralelismo, distribuição e adaptabilidade (ZAMANI *et al.* 2009).

2.6.1 Algoritmo das Células Dendríticas

Em Greensmith *et al.* (GREENSMITH *et al.* 2005) foram apresentadas as investigações iniciais no uso de CDs em SIA aplicados em SDIs por anomalias, tendo sido introduzido pela primeira vez o Algoritmo das Células Dendríticas (ACD).

Segundo Greensmith *et al.* (GREENSMITH *et al.* 2008), o ACD executa os passos:

1. Inicialização
2. Coleta de antígenos
3. Coleta de sinais de entrada
4. Cálculo dos sinais de saída
5. Migração das células dendríticas e mudança de estado
6. Cálculo do limiar de anomalia (MCAV)

O algoritmo é dividido em três fases: (i) inicialização (passo 1); (ii) atualização (passos 2 a 5); e (iii) agregação (passo 6). Na fase de inicialização são configurados e inicializados os parâmetros do algoritmo, tais como os limites de normalidade e as regras que definem um tipo de ataque. A fase de atualização é dividida em duas etapas: atualização de dados e amostragem de dados. Na atualização de dados (passos 2 e 3) ocorre um processo contínuo de atualização dos sinais de entrada e antígenos. Na etapa de amostragem de dados (passos 4 e 5), os sinais de entrada e os antígenos são amostrados e acessados pelas células dendríticas.

As células dendríticas são atualizadas com os novos valores dos sinais de entrada e dos antígenos e os sinais de saída são gerados. Neste ponto estas células migram para o linfonodo caso tenham atingido um determinado limiar, chamado de limiar de migração. Finalmente, a fase de agregação (passo 6) ocorre no linfonodo. Nesta fase os antígenos apresentados pelas CDs maduras ou semi-maduras são analisados e o limiar de anomalia dos antígenos, conhecido pela sigla MCAV (*Mature Context Antigen Value*), é calculado. O MCAV foi introduzido por Greensmith *et al.* (GREENSMITH *et al.* 2006) e varia entre zero (ou 0%) e um (ou 100%) e representa o quão anômalo é um determinado antígeno, sendo calculado pela seguinte fórmula: MCAV = (M)/(SM+M). Onde "M" representa a quantidade de um determinado antígeno em células maduras e "SM" a quantidade do mesmo antígeno em células semi-maduras. Caso o índice esteja acima de um valor pré-determinado (limiar de anomalia), os anticorpos são acionados, iniciando o combate aos invasores. O processo de avaliação de antígenos é repetido um determinado número de ciclos ou até que todos os antígenos tenham sido avaliados (GREENSMITH *et al.* 2006). A Figura 7 ilustra o diagrama de atividades de uma CD.

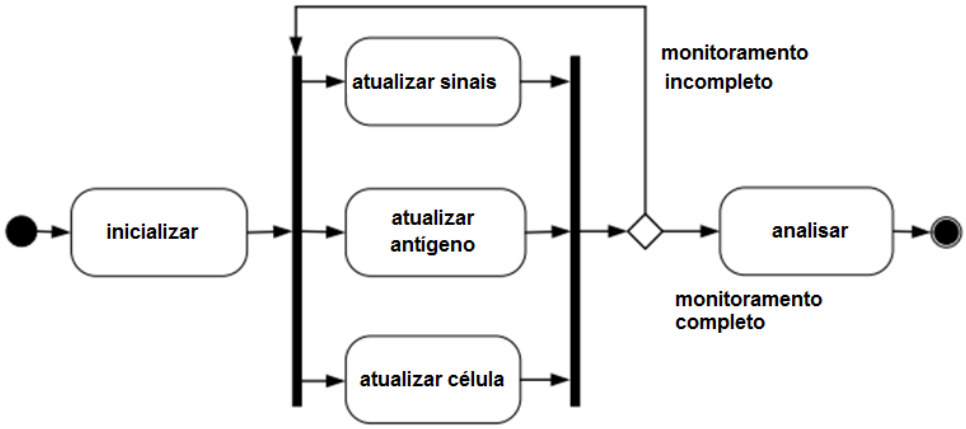


Figura 7. Diagrama de atividades de uma célula dendrítica (GREENSMITH *et al.* 2008).

As CDs processam os sinais de entrada de forma a gerar os sinais de saída de acordo com a equação 1 onde os pesos "W" dependem do tipo de sinal de saída que se quer obter. A equação é executada uma vez para cada um dos sinais de saída: (i) sinal de migração (*Costimulatory Molecules* - CSM); (ii) semi-maduro; e (iii) maduro. Cada CD permanece contabilizando os sinais de saída (por isso o somatório) enquanto o sinal de migração não atingir um limite pré-determinado (limiar de migração). Quando o sinal de migração atingir tal limite, a CD compara os valores armazenados para os sinais semi-maduro e maduro. O sinal de maior valor define o estado de maturação daquela CD. Em seguida, esta CD migra

para o linfonodo. Na equação 1, "P_i" representa o sinal de PAMP, "D_i" o sinal de perigo, "S_i" o sinal seguro e "IC" o sinal de inflamação. O somatório de cada um desses sinais, de 0 até uma quantidade de vezes "I", é multiplicado pelos seus respectivos pesos "W_p", "W_d" e "W_s" (GREENSMITH 2007).

$$Saída_{\left[\begin{array}{l} csm \\ semi-madura \\ madura \end{array}\right]} = \left(W_P \sum_{i=0}^{I} P_i + W_D \sum_{i=0}^{I} D_i + W_S \sum_{i=0}^{I} S_i \right) * (1 + IC) \qquad \text{Equação 1}$$

Os pesos "W" assumem valores diferentes para cada um dos três tipos de saída da equação. Os valores foram obtidos a partir de experimentos conduzidos por pesquisadores da área de Biologia, pela observação das interações e reações das células componentes do SIH (GREENSMITH 2007). Os resultados são apresentados na Tabela 1. Nesta Tabela, observa-se que o sinal de PAMP influencia mais a migração das células DC do que o sinal de perigo. Da mesma forma, o sinal seguro influencia negativamente o sinal de maturação. Já o sinal de semi-maturação é influenciado diretamente pelo sinal seguro.

Tabela 1. Pesos dos sinais para cálculo da saída (SILVA 2009).

Sinais de Saída	Sinais de Entrada		
	PAMP	Perigo	Seguro
Limiar de Migração	2	1	3
Semi-madura	0	0	1
Madura	2	1	-3

3 Trabalhos Relacionados

Em RSSFs, existem vários trabalhos propondo SDIs para RSSFs utilizando técnicas estatísticas, dentre os quais se destacam Da Silva *et al.* (DA SILVA *et al.* 2005), Onat e Miri (ONAT e MIRI 2005) e Martynov *et al.* (MARTYNOV *et al.* 2007). Dentre os trabalhos pesquisados, somente Drozda *et al.* (DROZDA *et al.* 2007), Liu e Yu (LIU e YU 2008), Kim *et al.* (KIM *et al.* 2006), Wallenta *et al.* (WALLENTA *et al.* 2010) e Zamani *et al.* (ZAMANI *et al.* 2009) apresentaram trabalhos utilizando um SDI imuno-inspirado para RSSFs.

Em Da Silva (DA SILVA 2005), foi proposto um SDI detector de anomalias dividido em três fases e baseado em monitoramento promíscuo da RSSF. Durante a primeira fase é feita a aquisição de dados: um nó monitor escuta a rede em modo promíscuo e guarda as informações utilizando a memória disponível no sensor. Os autores definiram um conjunto de regras que são aplicadas na segunda fase e, quando uma mensagem falha na verificação de uma destas regras, aumentam um contador daquela regra. Finalmente, na terceira fase, os contadores são comparados com valores de limiar. Se o número de falhas for maior do que o limiar pré-estabelecido um alarme é ativado. Para a operação do SDI proposto por Da Silva (DA SILVA 2005), torna-se necessário incluir um campo novo na estrutura da mensagem da aplicação para a utilização das regras propostas pelo SDI, de forma a permitir a coleta das informações necessárias pelos sensores. Esta alteração torna o SDI específico para a aplicação. Diferentemente de Da Silva (DA SILVA 2005), neste trabalho foi criado um SDI genérico e independente da aplicação em execução em uma RSSF.

Em Onat e Miri (ONAT e MIRI 2005), foi proposto um SDI baseado em anomalia para RSSFs que utiliza um algoritmo estatístico. Este algoritmo explora a estabilidade de uma rede estática em larga escala. No SDI proposto por Onat e Miri, os sensores têm a capacidade de armazenar estatísticas simples sobre o comportamento de seus nós vizinhos, tais como a taxa de transmissão de mensagens e a potência do sinal de transmissão, permitindo a identificação de anomalias nos mesmos. Na proposta de Onat e Miri, o SDI foi instalado em todos os sensores da rede mas não foi implementado nenhum mecanismo de colaboração entre os nós para a identificação de anomalias na RSSF. Diferentemente, neste trabalho, o SDI não necessita ser instalado em todos os sensores e há colaboração entre sensores executando diferentes papéis na rede.

Martynov *et al.* (MARTYNOV *et al.* 2007) propuseram e implementaram em uma plataforma real de sensores um SDI baseado em agentes e que utiliza a abordagem de detecção de anomalias. Este SDI era capaz de identificar um ataque de *Denial-of-Service* em

RSSFs. Os agentes foram chamados de *Status Nodes* e *Send and Receive Nodes*. O *Status Node* possui a funcionalidade exclusiva de alertar sobre a ocorrência ou não de um ataque não tendo participação na identificação do ataque. O *Send and Receive Node* possui a funcionalidade de enviar e receber mensagens em diferentes taxas, simulando diferentes taxas de envio das mensagens da aplicação. Estes agentes foram distribuídos por diferentes sensores da rede e, os *Send and Receive Nodes*, ao comparar o tráfego da rede com um *baseline* pré-estabelecido, conseguiram identificar se o ataque estava em andamento. Caso o ataque fosse identificado era transmitido a todos os nós da rede uma mensagem informando sobre o ataque. Apesar dos agentes distribuídos, os nós trabalhavam independentemente não havendo nenhum tipo de cooperação entre eles para a identificação do ataque.

Drozda *et al.* (DROZDA *et al.* 2007) propôs um SDI imuno-inspirado por detecção de mau uso, baseado na teoria da seleção negativa para RSSFs. Os autores verificam que a escolha do que representa um antígeno possui uma grande influência no desempenho do SIA. Na teoria da seleção negativa, caso os antígenos não sejam todos mapeados, surgirão "buracos" de detecção, os quais desencadearão a ocorrência de falsos positivos e falsos negativos. Além deste fato, os autores também informam que o desempenho do SDI depende do tamanho dos antígenos, onde, quanto maior o tamanho dos antígenos, maior a quantidade de detectores necessária para a correta identificação dos ataques. Isso vai de encontro com uma das características básicas dos sensores, a escassez de recursos, no caso a memória. Diferentemente, em nosso trabalho é utilizada a Teoria do Perigo e o algoritmo das células dendríticas que possuem uma abordagem diferente quanto a utilização dos antígenos na identificação de um ataque, além de termos usado um SDI baseado em anomalias e não em mau uso.

Em Liu e Yu (LIU e YU 2008), foram aplicadas as técnicas de seleção negativa e seleção clonal na criação de um SDI imuno-inspirado para uma RSSF. No SDI proposto pelos autores, as seguintes condições foram estabelecidas: (i) os nós são estáticos e nenhum nó novo é adicionado à rede; (ii) pacotes de dados são encaminhados para a EB e a rede usa uma estrutura baseada em árvore para o roteamento; (iii) nós adulterados funcionam normalmente, exceto quando realizam um ataque; (iv) houve treinamento suficiente antes do ataque iniciar; e (v) todos os nós da RSSF são equipados com o módulo extra de detecção de anomalias. Para monitorar o comportamento dos nós vizinhos, um determinado nó escuta as mensagens de seus vizinhos pelo seu módulo de detecção. Este módulo é dividido em quatro fases. A primeira fase é a "aquisição de próprio", onde, durante um período de treinamento, o nó

escuta as transmissões/recepções de seus vizinhos, extraindo informações dos pacotes trafegados e armazenando-as na memória do sensor. Ao adotar um período de aprendizado, os autores fazem com que o sistema não aceite mudanças na RSSF. À medida que os sensores vão tendo sua bateria esgotada, as características “próprias” da RSSF vão mudando. Isso gera a necessidade de novo treinamento, caso contrário podem ser emitidos falsos positivos ou falsos negativos uma vez que o cenário terá mudado. A segunda fase, chamada de “gerador de detectores”, ocorre após o treinamento do sistema. Nesta fase são gerados os detectores que identificarão os ataques, sendo os mesmos armazenados na memória do sensor. A terceira fase, chamada “detecção”, ocorre quando o treinamento termina e o sistema inicia a detecção. Nesta fase os pacotes enviados pelos nós vizinhos são ouvidos por um nó e os parâmetros a serem analisados (chamados antígenos) são extraídos. Se o detector atingir seu tempo limite de vida e o número de antígenos que combinaram for menor que o limite, o detector morrerá e um novo detector será gerado. Se o número de antígenos for maior do que o limite estipulado, o detector irá ativar um alarme de intrusão. Quando um detector for ativado, ele passa para a próxima fase: seleção clonal. Na quarta e última fase, chamada de “seleção clonal”, detectores ativos evoluem e vão para a memória passando a ter um tempo de vida maior e limites menores. Esta técnica permite que detectores guardados na memória possam ser ativados rapidamente quando ataques similares acontecem. A fim de se reduzir os falsos positivos, um mecanismo de co-estimulação foi proposto. Porém, neste mecanismo, um operador deve marcar uma *string* como própria a fim de corrigir um falso positivo, tornando o SDI dependente de intervenção humana e, portanto, violando um dos princípios básicos do SIH: a autonomia. Nas simulações, os autores verificaram que, com um conjunto de antígenos próprios mapeados e um grande conjunto de detectores, o SDI conseguiu uma taxa de 100% de detecção para todos os ataques simulados (verdadeiros positivos). Porém, foram verificados 92,3% de falsos positivos durante um ataque de *jamming*. Os falsos positivos emitidos pelo SDI fazem parecer com que a rede ainda está sendo atacada. Diferentemente, neste trabalho foi utilizada a Teoria do Perigo e o ACD customizado para realizar a detecção de ataques a RSSF. Os sensores podem ter implementado ou não os módulos do TinyOS que executam o SDI, permitindo a inclusão de novos sensores na RSSF sem nenhum tipo de alteração da aplicação. O SDI foi distribuído entre os sensores, os quais executaram papéis diferentes e complementares, não sendo necessário que todos os nós da rede os possuíssem instalados. Além disso, o SDI proposto pode ser utilizado em conjunto com qualquer tipo de aplicação, independente do tipo de protocolo de roteamento utilizado, pois a execução do SDI funciona independentemente do tipo de protocolo adotado. Em Liu e Yu (LIU e YU 2008)

não foi realizada uma avaliação da energia gasta pelo SDI, o que foi feito neste trabalho. No capítulo 5 encontra-se uma comparação entre o trabalho proposto e o trabalho realizado por Liu e Yu.

Wallenta *et al.* (WALLENTA *et al.* 2010) estenderam o trabalho de Kim *et al.* (KIM *et al.* 2006), no qual foi realizada a primeira implementação de um ACD aplicado a RSSF para a detecção de uma intrusão. Wallenta *et al.* (WALLENTA *et al.* 2010) ilustraram o quão próximo um SIA baseado na Teoria do Perigo e utilizando o ACD atendeu aos requisitos de uma RSSF. Nesse trabalho foi tratado um ataque específico ao protocolo de difusão direcionada em RSSF, chamado de "*Interest Cache Poisoning Attack*". Esse ataque pode desorganizar os caminhos dos dados em uma RSSF, demonstrando a vulnerabilidade das abordagens centradas em dados, muitas vezes utilizadas nessas redes. A arquitetura do SIA proposto em Wallenta *et al.* (WALLENTA *et al.* 2010) foi composta por dois componentes principais: (i) o componente de difusão direcionada; e (ii) o componente do ACD. Ambos os componentes estavam presentes no mesmo sensor e todos os sensores da rede possuíam a mesma configuração. Um sensor empregando difusão direcionada mantinha duas tabelas: a tabela de interesse de *cache* e a tabela de dados; e trabalhava com dois tipos de pacotes: os de interesse e os de dados. Os pacotes de interesse foram mapeados como os antígenos a serem classificados como normais ou perigosos. Dentro do componente de difusão direcionada foram programados dois sub-módulos, chamados de Extrator de Antígenos e Gerador de Sinais, os quais passaram a integrar o SIA no protocolo de difusão direcionada. Quando um pacote de interesse ou de dado chega a um nó, o protocolo atualiza os *caches* de interesse e de dados, respectivamente, extraindo as informações dos sinais e dos antígenos dos pacotes ou dos *caches*. Os sinais foram categorizados da seguinte forma: (i) sinal de perigo 1, gerado pela taxa de inserção no *cache* de interesse; (ii) sinal de perigo 2, gerado pela expiração das entradas do *cache* de interesse; (iii) sinal seguro, gerado pela chegada de pacotes de dados; (iv) PAMP, gerado pela falha na entrega de dados ao nó sorvedouro; (v) inflamação 1, gerado pelas alterações na direção dos gradientes; (vi) inflamação 2, gerado pela não combinação de dados com as entradas de *cache* de interesse. Nesse trabalho o SDI proposto pelos autores atendeu exclusivamente a uma RSSF que utilizava o protocolo de Difusão Direcionada e um ataque específico para esse protocolo, sendo de 30% a menor taxa de falsos negativos observada nos experimentos realizados. Já no presente trabalho, o SDI proposto é genérico, adaptando-se a qualquer tipo de protocolo e ataque.

Zamani *et al.* (ZAMANI *et al.* 2010) propuseram uma arquitetura genérica utilizando agentes móveis para um SDI imuno-inspirado na Teoria do Perigo. Esta arquitetura foi aplicada a uma RSSF a fim de identificar o ataque de *Distributed Denial-of-Service* (DDoS) ao se utilizar o protocolo de Difusão Direcionada. A arquitetura foi dividida entre agentes estáticos, os quais permanecem fixos em determinados sensores e simulam os tecidos do SIH; e em agentes móveis, que são transmitidos entre os sensores, simulando o comportamento das células do SIH. Os agentes estáticos simularam as características dos órgãos do SIH: timo, medula, linfonodo e tecidos. Os agentes móveis simularam as características das células B, células T e das células dendríticas. Os autores em Zamani et al. (ZAMANI et al. 2010) informam que os agentes estáticos trocam mensagens, chamadas de *Address Notification Messages* (ANM), para a troca de informações entre os agentes estáticos. Com estas trocas de informações os agentes conseguem montar uma tabela de mapeamento a fim de localizarem-se na rede. Porém, os autores não informaram a estrutura computacional e o tamanho destas mensagens. Os autores utilizam uma equação similar a equação 1 da seção 2 (GREENSMITH *et al.* 2007), porém, não especificam qual parâmetro seria considerado como o sinal seguro. Já o sinal de PAMP foi alterado para refletir um grau de afinidade entre as células, não identificando padrões de irregularidade do sistema, conforme estabelecido em Greensmith (GREENSMITH *et al.* 2007). Também não informaram os valores dos pesos utilizados para os três sinais de entrada. Os autores não informam, caso ocorra a falha de um dos agentes estáticos, como o sistema se recuperaria. Também não informaram caso um agente estático esteja distante de um sensor que necessite enviar uma mensagem contendo um agente móvel, se ocorreria uma transmissão direta ou multihop. Não avaliaram energia, não implementaram em sensores reais, apenas no simulador NS-2 e não informaram como os sensores foram dispostos na topologia (aleatoriamente ou não). Pelos resultados apresentados por Zamani *et al.* (ZAMANI *et al.* 2010), para uma quantidade de 100 nós, os autores obtiveram uma taxa de FN igual a 40.0% e uma taxa de FP igual a 8.23%. Diferentemente, no trabalho proposto o mapeamento biológico restringiu-se ao mapeamento dos componentes biológicos célula dendrítica, linfonodo, células B, células T e tecido onde, em nosso mapeamento considerou-se o tecido como toda a rede de sensores, não necessitando de nenhum tipo de alteração nos sensores. As funcionalidades da medula (produção de células B) e do timo (produção de células T) foi implementada no próprio sensor que continha o linfonodo.

4 Sistema de Detecção de Intrusos Imuno-inspirado

Neste capítulo são descritos: (i) a arquitetura lógica do SDI proposto para RSSFs, onde serão apresentados os seus elementos constituintes; (ii) o mapeamento dos elementos computacionais nos elementos imuno-inspirados; (iii) a descrição das fases que definem o fluxo de funcionamento do SDI proposto; (iv) o ACD customizado para RSSFs; e (iv) uma descrição do funcionamento do SDI proposto.

4.1 Arquitetura Lógica do SDI

A arquitetura lógica do SDI para RSSF, mostrada na Figura 8, segue a arquitetura proposta pelo *Common Intrusion Detection Framework* (CIDF) (DEBAR *et al.* 1999; BARBOSA 2000; GARCÍA-TEODORO *et al.* 2008) e consiste dos seguintes componentes: Monitoramento, Gestor de Detecção de Intrusão, Gestor de Contexto, Gestor de Decisão, Base de Parâmetros, Base de Regras e Contramedidas.

Os componentes da arquitetura proposta foram agrupados em quatro subsistemas: (i) Ambiente Monitorado (E-BOX); (ii) Detector de Intrusos (A-BOX); (iii) Armazenador (D-BOX); e (iv) Contramedidas (C-BOX).

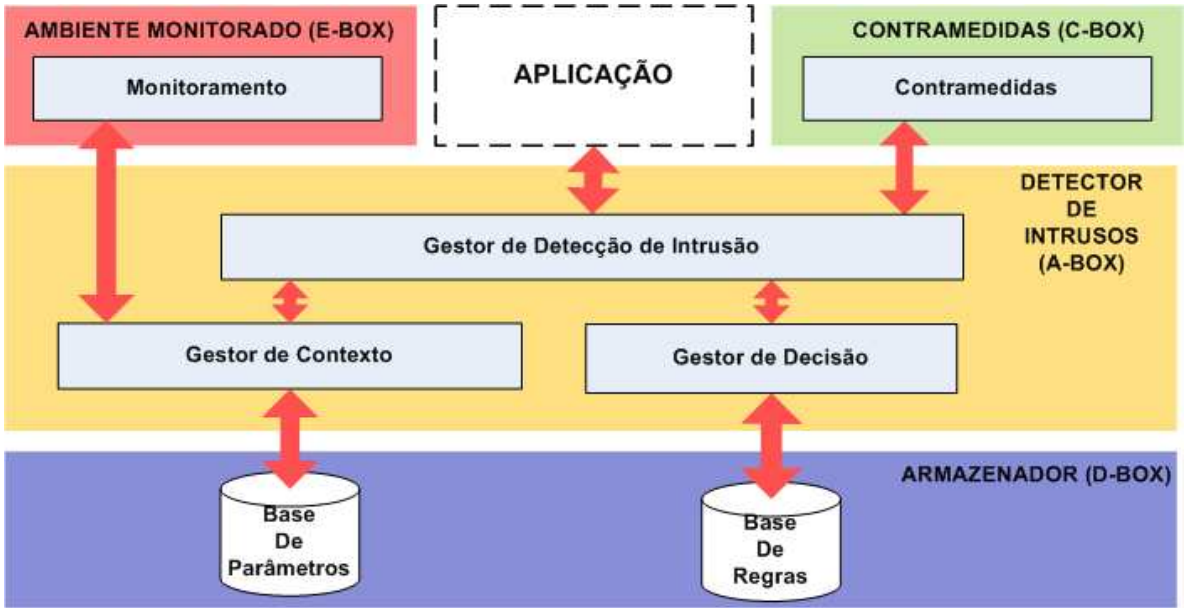


Figura 8. Arquitetura Lógica do SDI.

O subsistema **Ambiente Monitorado**, composto pelo componente Monitoramento, é responsável pela captura dos valores dos parâmetros definidos pelo Gestor de Contexto, tais como a quantidade de mensagens enviadas e recebidas e a intensidade do sinal recebido, que representam as entradas do SDI proposto. Estes parâmetros são utilizados para determinar uma possível invasão.

No subsistema **Detector de Intrusos** são feitas as análises das informações coletadas para que seja tomada uma decisão referente à presença ou não de um intruso no ambiente onde se localiza o nó. Este subsistema é composto pelos componentes: Gestor de Detecção de Intrusão, Gestor de Contexto e Gestor de Decisão.

O **Gestor de Detecção de Intrusão**, componente central na arquitetura, é o responsável por organizar as tarefas e coordenar as ações e respostas dos outros gestores. Durante a instanciação do sistema, o Gestor de Detecção de Intrusão informa ao Gestor de Contexto o tipo de ataque que o SDI está pronto para monitorar. O Gestor de Contexto, ao tomar conhecimento do tipo de ataque, consulta a Base de Parâmetros de forma a descobrir quais os parâmetros que devem ser monitorados pelo componente de Monitoramento para aquele tipo de ataque. O componente de Monitoramento fornece ao Gestor de Contexto as medições mais recentes desses parâmetros. Os valores dos parâmetros recebidos pelo Gestor de Contexto são repassados pelo Gestor de Detecção de Intrusão para o Gestor de Decisão, a fim de se identificar a existência ou não de ataque na área monitorada e, em caso de ataque, o tipo de ataque e o grau de anomalia do mesmo. Estas informações relativas à existência ou não de ataque, o tipo de ataque e o grau de anomalia são retornados para o Gestor de Detecção de Intrusão que por sua vez as encaminha ao componente de Contramedidas, o qual verifica a ação (contramedida) a ser executada.

O **Gestor de Contexto** é responsável por duas funcionalidades: (i) gerenciar o Monitoramento; e (ii) gerenciar a Base de Parâmetros. A funcionalidade de gerenciar Monitoramento é responsável pela solicitação de leitura de parâmetros do meio pelo componente de Monitoramento, enquanto que, a funcionalidade de gerenciar Base de Parâmetros é responsável pelo armazenamento e leitura dos parâmetros recebidos do componente de Monitoramento na Base de Parâmetros. O Gestor de Contexto, ao ser informado pelo Gestor de Detecção de Intrusão sobre qual(is) o(s) ataque(s) que deve(m) ser monitorado(s) pelo SDI, consulta a Base de Parâmetros para descobrir qual(is) o(s) parâmetro(s) que deve(m) ser monitorado(s) para aquele(s) tipo(s) de ataque(s) e, em seguida, aciona o componente de Monitoramento passando o(s) parâmetro(s) relativo(s) ao(s) ataque(s) sendo avaliado(s). O Gestor de Contexto mantém uma estrutura de dados, denominada **Base de Parâmetros,** com os valores mais recentes coletados pelo componente Monitoramento e a relação de tipos de ataques que consegue identificar e os parâmetros relacionados aos mesmos.

O **Gestor de Decisão** é responsável por realizar três funcionalidades: (i) gerenciar a Base de Regras; (ii) executar o algoritmo imuno-inspirado; e (iii) identificar a presença de ataque(s). O Gestor de Decisão necessita consultar no repositório **Base de Regras** as regras específicas estabelecidas para cada tipo de ataque. Uma vez identificado uma possibilidade de ataque, o Gestor de Detecção de Intrusão é avisado. A detecção de diferentes ataques pode ser feita com a aplicação de um conjunto de regras distintas e avaliando o conjunto adequado de antígenos. Além disso, o ajuste do limiar de anomalia (MCAV) também é importante para a detecção do ataque e localiza-se nesta base.

O subsistema **Armazenador** representa a parte do sistema onde serão armazenadas: (i) a **Base de Parâmetros**, que contém: o histórico dos parâmetros coletados, o tipo de ataque, a lista de parâmetros desse ataque e para cada um dos parâmetros os valores limiares; e (ii) a **Base de Regras**, contendo as regras que identificam os tipos de ataques que o sistema é capaz de identificar e o limiar de anomalia (MCAV) estabelecido pelo administrador da rede por cada tipo de ataque. Estas bases são consultadas e comparadas com os dados monitorados.

O subsistema de **Contramedidas** contém o componente chamado Contramedidas, responsável por executar ações de combate aos ataques identificados. As contramedidas podem ser ações diretas, executadas no próprio nó, ou o envio de informações de alerta ao administrador para que o mesmo tome alguma medida cabível.

4.1.1 Diagrama de Sequência

Na Figura 9 e na Figura 10 são apresentados os diagramas de sequência da arquitetura proposta, especificando as interações entre os componentes que representam as células dendríticas e o linfonodo, respectivamente.

No passo (1) o Gestor de Detecção de Intrusão informa ao Gestor de Contexto que um determinado ataque deve ser monitorado chamando o comando *informarTipoDeAtaque*. No passo (2) o Gestor de Contexto consulta a Base de Parâmetros para identificar qual(is) o(s) parâmetro(s) que devem ser monitorados. Em seguida o Gestor de Contexto passa a informação de qual parâmetro deve ser coletado do meio ao Monitoramento chamando o comando (3) *solicitarParametro*. O Monitoramento, então, realiza em (4) a leitura do meio pelo comando *lerParametro*, capturando o valor mais atual do(s) parâmetro(s) solicitado(s). Caso seja necessário algum processamento sobre o valor lido, como por exemplo, a contagem das mensagens enviadas ou recebidas, o Monitoramento executa o comando (5) e, posteriormente repassa o valor lido para o Gestor de Contexto chamando o comando (6). O

Gestor de Contexto, recebendo este novo valor, compara-o com a Base de Parâmetros pelo comando *verificarNormalidade* (7), armazenando-o com o comando *armazenarValorParametro* (8). Os comandos (7) e (8) fazem parte da funcionalidade de gerenciamento da Base de Parâmetros do Gestor de Contexto. O valor é então encaminhado ao Gestor de Detecção de Intrusão pelo comando *enviarParametro* (9), que repassa este valor para o Gestor de Decisão em (10). No Gestor de Decisão, os valores recebidos são verificados por tipo de ataque conforme definido na Base de Regras (11) e, em seguida, dão entrada na funcionalidade do algoritmo imuno-inspirado na funcionalidade de algoritmo das células dendríticas do Gestor de Decisão (12). O acesso à Base de Regras representa a funcionalidade de gerenciamento da Base de Regras do Gestor de Decisão.

Uma vez identificado um ataque, o Gestor de Decisão chama o comando *avisarCelulaDendrítica* no passo (13), informando ao Gestor de Detecção de Intrusão que, finalmente, envia estas informações a um nó atuando como um linfonodo, chamando o comando *migrarCelulaDendritica* (14).

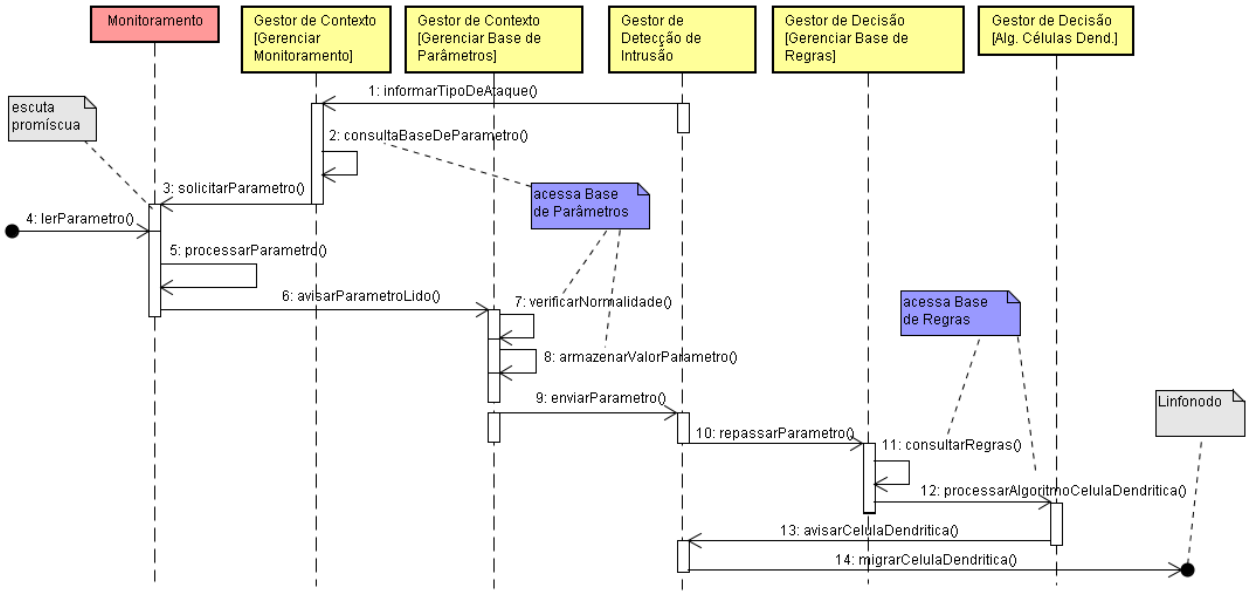


Figura 9. Diagrama de sequência - Nó no papel de CD.

O nó atuando como linfonodo escuta as mensagens enviadas a ele pelos nós atuando na função de CDs e contendo informações que representam as CDs migradas no passo (1). Em seguida o Gestor de Decisão, na funcionalidade de identificar ataques, calcula o limiar de anomalia (MCAV) de cada ataque identificado no sistema no passo (2). O Gestor de Decisão informa este índice ao Gestor de Detecção de Intrusão no passo (3), que o envia ao

componente de Contramedidas no passo (4). No componente de Contramedidas esse ataque é identificado no passo (5) e as medidas para conter o ataque são repassadas ao Gestor de Detecção de Intrusão no passo (6), a fim de que este componente acione contramedidas específicas para aquele tipo de ataque e instrua os outros nós da rede a fazerem o mesmo, conforme ilustra o passo (7). Neste nó, está previsto ainda o passo (8), *informarEstacaoBase*, que é responsável por avisar a EB sobre os ataques que estiverem ocorrendo na rede.

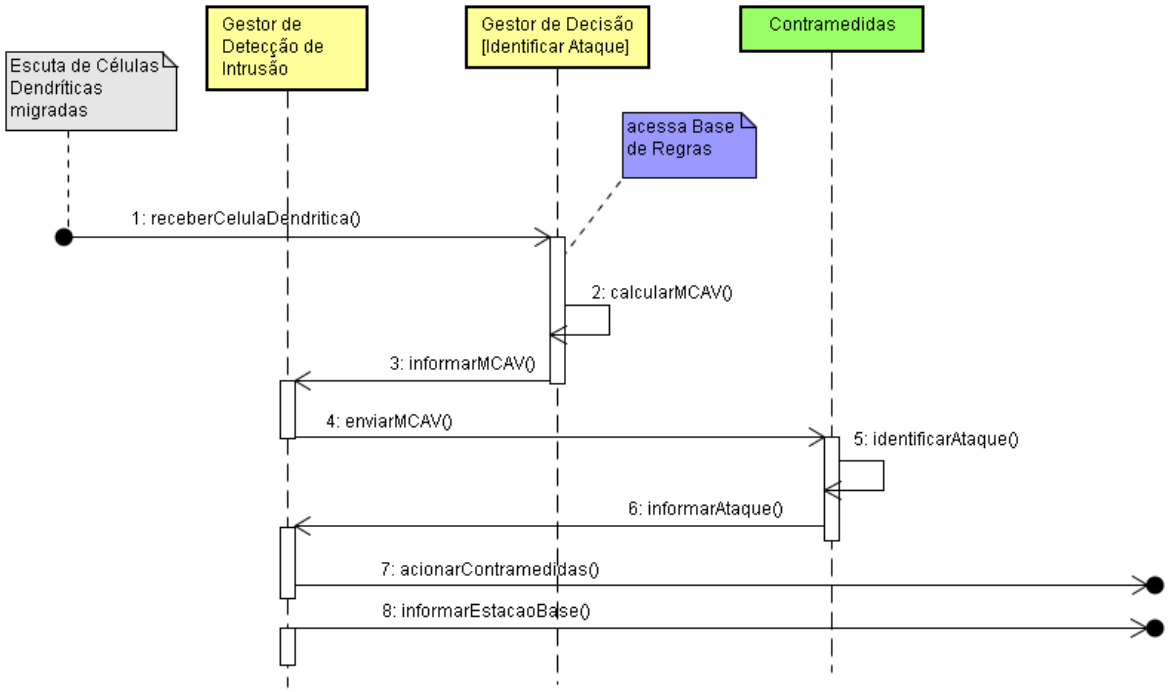


Figura 10. Diagrama de sequência - Nó no papel de linfonodo.

4.2 Mapeamento dos Elementos Computacionais em Imuno-inspirados

Neste trabalho, uma RSSF é composta por diversos nós sensores, os quais podem assumir dois papéis diferentes na rede: célula dendrítica e linfonodo. O sensor que assume o papel de célula dendrítica é chamado de sensor-cd enquanto que o sensor que realiza o papel de linfonodo é chamado de sensor-linfo (SALMON *et al.* 2010).

Nesta seção, as funcionalidades dos elementos computacionais são mapeadas nas funcionalidades dos elementos biológicos, conforme mostra a Tabela 2.

Os **patógenos** são tratados como sendo os ataques.

Os **antígenos** representam uma forma de identificar um ataque que se queira classificar. Esta identificação é específica para cada tipo de ataque, ou seja, cada ataque possui um

identificador próprio e único. Neste trabalho, por exemplo, os identificadores utilizados eram o campo indicando o nó origem da mensagem. Esta informação foi retirada do cabeçalho da mensagem recebida pelo nó. Desta forma, estas mensagens podem ser consideradas próprias ou não-próprias, ou seja, podem ter sido geradas por um nó legítimo ou por um nó intruso, devendo ser avaliadas quanto ao perigo que representam ao sistema.

O **tecido** sendo avaliado quanto à presença de um perigo causado por algum patógeno é representado pelos nós que compõem a rede sem fio.

A **região do perigo** é representada pela área de alcance dos nós, que é controlada pelo componente Monitoramento e pela funcionalidade de gerenciar monitoramento do Gestor de Contexto.

As **células dendríticas** são representadas pelo componente Gestor de Detecção de Intrusão e pela funcionalidade de gerenciar Base de Parâmetros do Gestor de Contexto e possuem os seguintes atributos: (i) Identificador; (ii) Antígeno; (iii) Estado; (iv) Tempo de Migração; (v) Sinal de Perigo; (vi) Sinal de PAMP; (vii) Sinal Seguro e (viii) Sinal de Inflamação. O Identificador associa a CD ao nó onde ela foi criada. O Estado contém o valor do estado atual da CD no sistema (imatura, semi-madura ou madura). O tempo de vida de uma CD representa o tempo máximo que uma CD permanece coletando antígenos e sinais de entrada no tecido até que seu estado de maturação seja alterado, fazendo-a migrar do tecido para o linfonodo.

Tabela 2. Mapeamento biológico / computacional.

Elementos biológicos	Elementos computacionais
Patógenos	Ataques
Antígenos	Informações retiradas das mensagens recebidas ou enviadas
Tecido	Nós que compõem a RSSF
Região de perigo	Área de alcance dos nós sensores-cd
Células Dendríticas	Componente Gestor de Detecção de Intrusão e componente Gestor de Contexto (funcionalidade de gerenciar Base de Parâmetros) dos sensores-cd
Linfonodo	Componente Gestor de Decisão dos sensores-linfo
Células B e Células T	Componente Contramedidas
Anticorpos	Contramedidas acionadas pelos nós

O **linfonodo** é representado pelo mecanismo de decisão contido no componente Gestor de Decisão.

As **células B** e as **células T**, representando o sistema imunológico adaptativo, são representadas pelo componente de Contramedidas. Este componente é o responsável pelo combate aos invasores, e suas ações são consideradas como sendo os **anticorpos**.

Quanto aos sinais de entrada, tem-se que os sinais de perigo, seguro, PAMP e de inflamação são parâmetros variáveis e diferentes para cada tipo de ataque e serão discutidos nas próximas seções.

Os repositórios **Base de Parâmetros** e **Base de Regras** foram modelados para disponibilizar uma funcionalidade computacional de armazenamento de dados, sem buscar inspiração no sistema biológico.

4.3 Fases do Sistema de Detecção de Intrusão Imuno-inspirado

O fluxo do funcionamento do SDI proposto é dividido em quatro fases: (i) Fase de Coleta, (ii) Fase de Análise, (iii) Fase de Decisão e (iv) Fase de Reação.

Os procedimentos relativos à primeira e a segunda fase estão relacionados com os procedimentos do ACD (Figura 11). A terceira fase está relacionada com os procedimentos de decisão do linfonodo e a quarta fase representa o sistema imunológico adaptativo e as reações contra os invasores (Figura 12).

Conforme ilustra a Figura 11, o SDI inicia sua operação com o componente Inicializar. Este componente é responsável pela inicialização das variáveis do sistema. Na primeira fase os antígenos e os sinais de entrada são capturados, conforme ilustrado nos componentes Atualizar Sinais de Entrada, Atualizar Antígenos e Atualizar Células. Na segunda fase, realizada no componente Análise da Figura 11, os sinais de entrada e os antígenos são analisados de forma a gerar os sinais de saída. Esses sinais de saída informam o estado de maturação das células dendríticas. A informação sobre o estado da CD é enviado para o linfonodo pelo componente Migração. Na terceira fase as CDs apresentam os antígenos próprios e os não-próprios (componente Migração) ao linfonodo. O linfonodo, então, recebe as informações das CDs (componente Receber CDs) e calcula o limiar de anomalia (MCAV) apresentado pelos antígenos (componente Calcular MCAV). Na quarta e última fase as células B e T iniciam a produção de anticorpos os quais irão combater aquele invasor específico (componente Contramedidas). A Tabela 3 ilustra a localização e a funcionalidade dos

componentes computacionais de acordo com as fases do SDI. As descrições detalhadas das fases são apresentadas em seguida.

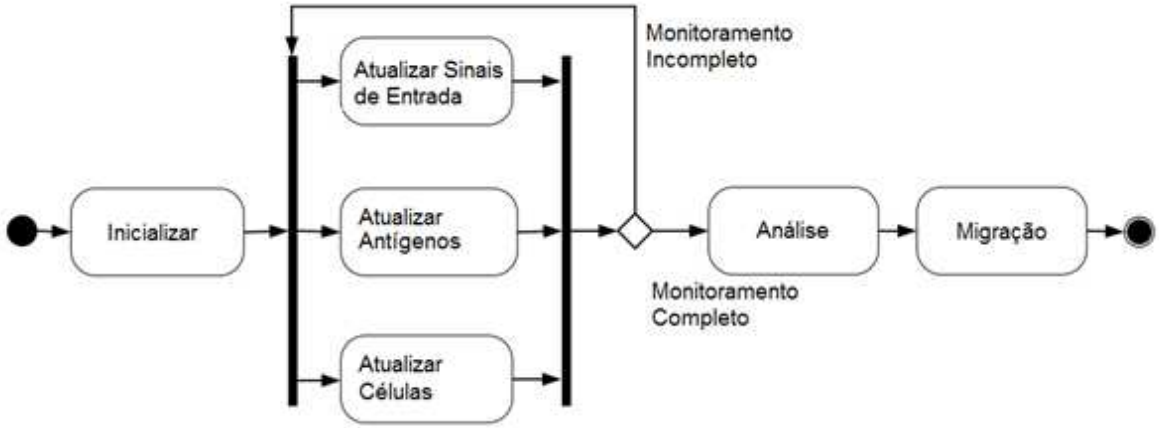


Figura 11. Diagrama de atividades da primeira e da segunda fase da Arquitetura.

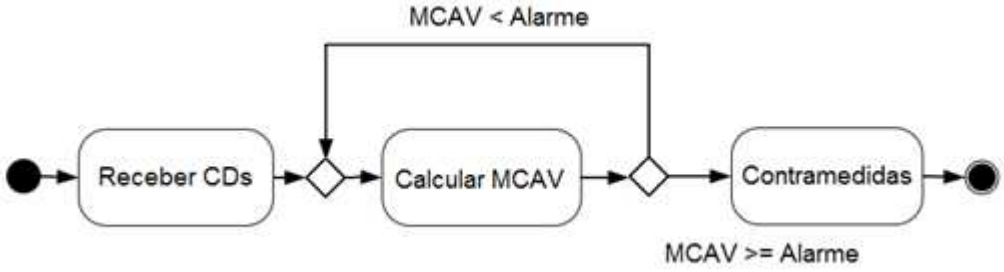


Figura 12. Diagrama de atividades da terceira e da quarta fase da Arquitetura.

4.3.1 Fase de Coleta

É na fase de Coleta que o ACD inicia sua execução. Os sinais a serem usados para a detecção da intrusão são coletados pelo componente Monitoramento dos sensores-cd, que podem operar em modo promíscuo, capturando todas as informações trafegadas na rede, ou no modo normal, capturando apenas as informações destinadas a ele próprio.

Nesta fase, os sensores-cd coletam os sinais de entrada da rede, os quais são específicos para cada tipo de ataque. A definição de cada sinal de entrada determina o tipo de ataque que um sensor está monitorando. Estes sinais são representados por um conjunto de parâmetros que devem ser monitorados de forma a identificar um ataque. Após o término da Fase de Coleta de Dados, inicia-se a Fase de Análise do perigo.

4.3.2 Fase de Análise

A análise é baseada na comparação de parâmetros e regras previamente definidos e armazenados na Base de Parâmetros e na Base de Regras. Por exemplo, para o ataque de *Denial-of-sleep*, um parâmetro a ser monitorado poderia ser o sinal de potência de uma mensagem recebida por um sensor, enquanto que uma regra seria como este sinal deveria se comportar, seus limites mínimo e máximo.

O Gestor de Contexto recebe o valor de um parâmetro recebido do Monitoramento e passa esta informação ao Gestor de Detecção de Intrusão, que a repassa para o Gestor de Decisão.

No Gestor de Decisão, os dados coletados servem de entrada para o algoritmo imuno-inspirado e são processados até que a CD migre para o linfonodo. Durante o processamento os dados coletados são acumulados por meio de uma função utilidade, detalhada no Capítulo 2 (Equação 1). A saída desta função será o estado de maturação da CD analisada: madura, para os casos de perigo identificado, ou semi-madura, caso seja verificada a normalidade da rede. Ao atingir o limiar de migração, a CD migra para um dos estados: maduro ou semi-maduro, dando início à terceira fase.

Tabela 3. Fases do SDI imuno-inspirado e seus elementos componentes

Componentes biológicos	Sistema Imunológico Inato			Sistema Imunológico Adaptativo
	Célula dendrítica		Linfonodo	Células B e T
Componentes computacionais	Fase de Coleta	Fase de Análise	Fase de Decisão	Fase de Reação
	Monitoramento; e Gestor de Contexto (gerenciar monitoramento)	Gestor de Contexto (gerenciar base de parâmetros) Gestor de Decisão (gerenciar base de regras e algoritmo imuno-inspirado)	Gestor de Decisão (identificar ataque)	Contramedidas

4.3.3 Fase de Decisão

A fase de Decisão ocorre no linfonodo e é executada dentro do Gestor de Decisão, exercendo a funcionalidade de identificar um ataque. Neste componente as CDs que migraram são contabilizadas e os antígenos apresentados por elas são classificados como normais ou anômalos, gerando o limiar de anomalia (MCAV). Este índice é passado para o Gestor de

Detecção de Intrusão, que o repassa para o componente de Contramedidas, dando início à quarta e última fase.

4.3.4 Fase de Reação

Nesta fase, que representa as reações do Sistema Imunológico Adaptativo, o componente de Contramedidas recebe informações do tipo e intensidade do ataque (limiar de anomalia - MCAV) que está ocorrendo na rede.

O componente Contramedidas é o responsável pelo início das ações de combate aos invasores, lançando anticorpos que irão combater a invasão.

4.4 Algoritmo Personalizado das Células Dendríticas Aplicado à RSSF

A Figura 13 apresenta o pseudocódigo do ACD original, proposto por Greensmith (GREENSMITH et al. 2007). Os índices e as estruturas de dados utilizados no algoritmo original estão ilustrados na Tabela 4 e na Tabela 5, respectivamente.

```
1   Criar DCs;
2   Inicializar parâmetros {I, J, K, L, M, N, O, P, Q};
3   Para l < L faça
4       Atualizar A e S;
5       Para m = 0 até M faça
6           Para n = 0 até Q faça
7               DCm amostra Q antígeno de A;
8           Fim
9           Para i = 0 até I e j = 0 até J faça
10              S(DC)ij = Sij;
11          Fim
12          Para n = 0 até N faça
13              DCm processa a(DC)nm;
14          Fim
15          Para p até P faça
16              Calcula Op;
17              Op(m) = Op(m) + Op;
18          Fim
19          Se Op(m) > Tm então
20              DCm removida da população;
21              DCm migra para linfonodo;
22              DCm limpa vetor de antígenos e sinais;
23          Fim
24      Fim
25      l++;
26  Fim
27  Analisa antígenos e calcula MCAV;
```

Figura 13. Pseudocódigo do ACD original (GREENSMITH *et al.* 2007).

Pelo algoritmo original, nas linhas 1 e 2 são inicializados todos os parâmetros necessários para a execução do ACD. Na linha 3 é executado um loop que controla a quantidade de ciclos de atualização (L) de antígenos e sinais. Na linha 4 as estruturas de dados

que contém os antígenos (A) e os sinais de entrada (S), representando o tecido sendo avaliado, são atualizadas. Na linha 5 inicia-se um loop que percorrerá todas as CDs da população (M), fazendo-as coletar e avaliar os antígenos e os sinais de entrada. No loop das linhas 6 a 8 cada CD (DC_m) preenche sua estrutura de dados de antígenos e no loop das linhas 9, 10 e 11 são coletados seus sinais de entrada (seguro, perigo e PAMP). Nas linhas 12, 13 e 14 cada CD processa o vetor de antígenos ($a(DC)_{nm}$). O loop das linhas 15 a 18 calcula os três sinais de saída (O_p) para aquela CD naquele ciclo. O teste condicional nas linhas 19 a 23 remove a CD da população, migrando-a para o linfonodo e limpando seu conteúdo de antígenos e sinais, recolocando-a na população. A linha 25 incrementa o ciclo de execução do algoritmo. Finalmente, na linha 27 é feito o cálculo do MCAV para os antígenos coletados.

Tabela 4. Índices do ACD original.

Índice	Variação	Descrição
i	De 0 até I	número de sinais de entrada por categoria
j	De 0 até J	número de categorias de sinais de entrada
k	De 0 até K	número de antígenos no vetor de antígenos do tecido
l	De 0 até L	número de ciclos da CD
m	De 0 até M	número de CDs na população
n	De 0 até N	tamanho do vetor de antígenos da CD
p	De 0 até P	número de sinais de saída por CD
q	De 0 até Q	número de antígenos amostrados por CD, por ciclo
T_{max}	T_m	tamanho do vetor de antígenos do tecido

Tabela 5. Estruturas de dados do ACD original.

Estrutura	Descrição
T = {S, A}	o tecido
S	matriz de sinais do tecido
S_{ij}	sinal do tipo i, categoria j na matriz de sinais S
A	vetor de antígenos do tecido
a_k	antígeno k no vetor de antígenos do tecido
$DC_m = \{s(m), a(m), o_p(m), t_m\}$	uma CD da população
s(m)	matriz de sinais da DC_m
a(m)	vetor de antígenos da DC_m
$o_p(m)$	sinal de saída p da DC_m
t_m	limiar de migração da DC_m
w_{ijp}	peso dos sinais de entrada Sij

No presente trabalho, o ACD original foi adaptado de forma a melhor explorar a característica de densidade das RSSFs e reduzir o processamento e as estruturas de dados necessárias em cada nó sensor (Figura 14 e Figura 15). Desta forma, os procedimentos do ACD original foram divididos entre o sensor-cd e o sensor-linfo. Cada sensor-cd ficou

responsável pelo procedimento de uma CD apenas, tornando desnecessário qualquer tipo de processamento específico para a criação de uma CD, conforme ocorre na linha 1 do algoritmo original. O *loop* da linha 5 foi excluído dos sensores-cd. As funcionalidades dos *loops* das linhas 6 a 8, 9 a 11, 12 a 14 e 15 a 18 permaneceram inalteradas. O teste condicional da linha 19 até a linha 23 foi deslocado para fora do *loop* da linha 5 e permaneceu nos sensores-cd. Ao término de um ciclo de execução, ou seja, após o valor do limiar de migração atingir o valor limite, o sensor-cd envia uma mensagem de controle para o sensor-linfo, indicando seu estado final e quais antígenos foram processados, e reinicia o ciclo de execução (Figura 14). A linha 27, referente ao cálculo do limiar de anomalia (MCAV), passou a ser executada apenas pelo nó sensor-linfo, que usa como informação de entrada as mensagens recebidas dos nós sensores-cd, enviando para a EB uma mensagem informando o MCAV obtido (Figura 15).

É importante mencionar que no ACD original o aumento de confiabilidade quanto à decisão de existir ou não um ataque era obtido pela existência de um conjunto de CDs em um único dispositivo. Em nossa proposta, essa confiabilidade é obtida fazendo com que existam vários nós com a funcionalidade de sensor-cd e um sensor-linfo (para cada grupo de sensores sensor-cd) que reúne as avaliações sobre a existência ou não de ataque provenientes desses sensores-cd. Essa decisão visa explorar o fato das RSSFs serem constituídas por vários pequenos nós dispostos próximos uns dos outros, o que permite ter diferentes ângulos de visão sobre um mesmo ataque.

```
 1  Inicializar parâmetros;
 2  Atualizar A e S;
 3  Para n = 0 até Q faça
 4      DCm amostra Q antígeno de A;
 5  Fim
 6  Para i = 0 até I e j = 0 até J faça
 7      S(DC)ij = Sij;
 8  Fim
 9  Para n = 0 até N faça
10      DCm processa a(DC)nm;
11  Fim
12  Para p até P faça
13      Calcula Op;
14      Op(m) = Op(m) + Op;
15  Fim
16  Se Op(m) > Tm então
17      DCm removida da população;
18      DCm migra para linfonodo;
19      DCm limpa vetor de antígenos e sinais;
20  Fim
```

Figura 14. Pseudocódigo do ACD customizado para RSSF (papel sensor-cd).

```
1    Analisa antígenos;
2    Calcula MCAV;
3    Envio de alerta para a EB;
```

Figura 15. Pseudocódigo do ACD customizado para RSSF (papel sensor-linfo).

Nesse trabalho, observou-se que o cálculo do limiar de migração poderia ser customizado para diferentes tipos de ataques, não sendo necessário o cálculo realizado pela equação 1 do capítulo 2, economizando processamento pelos sensores. Desta forma, o *loop* das linhas 15 a 18 da Figura 13 (seção 4.4) apesar de mantido, pode ter seu processamento reduzido a duas repetições. Este procedimento será detalhado no próximo capítulo.

4.5 Descrição da Operação do SDI

Os sensores-cd (CD na Figura 16) devem coletar os valores dos parâmetros solicitados pelo Gestor de Detecção de Intrusão, e, ao executar o ACD customizado, enviar mensagens representando o estado das CDs ao sensor-linfo (LN na Figura 16). Em um ambiente onde não esteja ocorrendo um ataque, os sensores-cd enviam mensagens ao sensor-linfo indicando uma migração de uma CD semi-madura. Já em um ambiente com a presença de um atacante, os sensores-cd devem enviar mensagens indicando uma migração de uma CD madura.

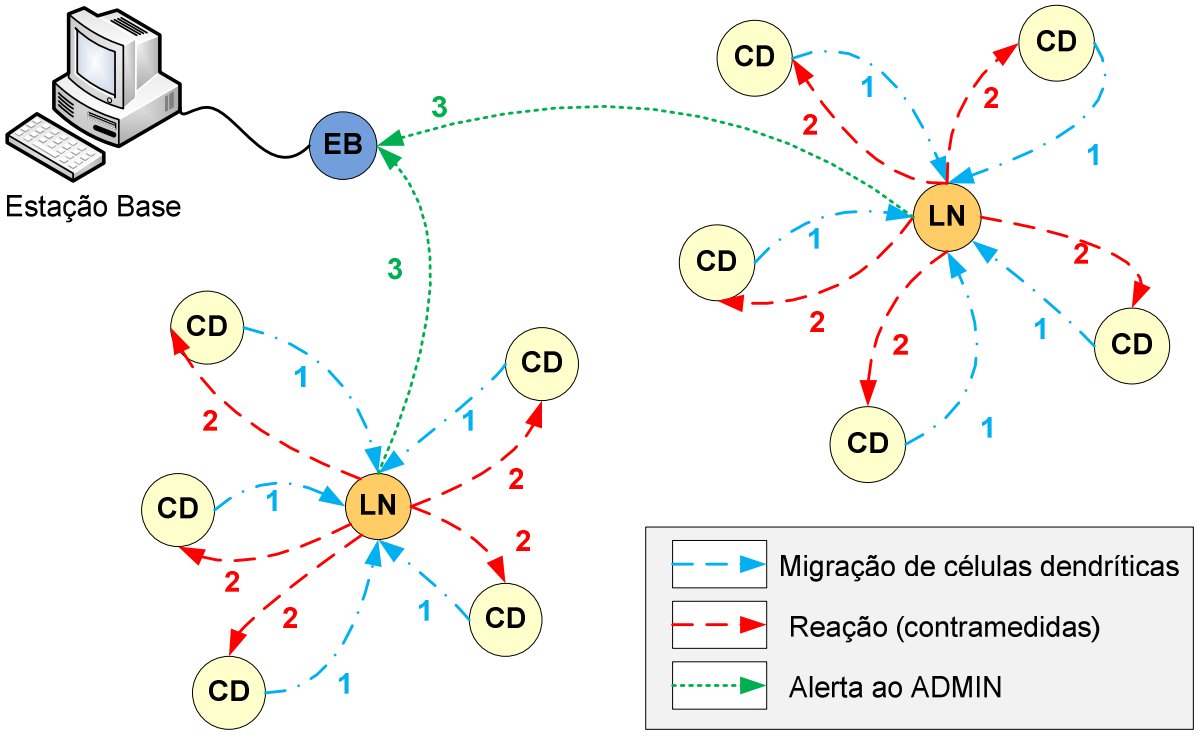


Figura 16. Interações entre sensores-cd e sensor-linfo.

Cada sensor-linfo é capaz de identificar as informações contidas nas mensagens que representam as CDs que migraram. Com estas informações o sensor-linfo consegue identificar e calcular o MCAV por tipo de ataque. Ao ser verificado que um determinado tipo de ataque está ocorrendo, o sensor-linfo emite um alerta por meio de uma mensagem contendo informações sobre aquele ataque aos outros nós, fazendo-os ativarem contramedidas específicas conforme o tipo de ataque identificado. Estas contramedidas são mensagens que contém instruções aos nós que estão sofrendo aquele tipo de ataque ordenando-os a acionar medidas de forma a eliminar a ameaça. Estas ações podem ser, por exemplo, ativar criptografia, iniciar a utilização de autenticação, excluir algum nó da rede, ou mesmo desligar-se por algum tempo específico. Estas ações dependem do tipo de ataque que a rede esteja sofrendo.

Na Figura 16 podem ser observados dois conjuntos de cinco sensores-cd comunicando-se, cada um, com seu sensor-linfo correspondente. As setas indicam a ordem dos eventos: as setas indicativas “1” indicam o envio de uma CD madura ou semi-madura de um sensor-cd para o sensor-linfo. As setas indicativas “2” indicam que o sensor-linfo identificou a presença de um ataque na RSSF e enviou uma mensagem indicando que todos os sensores-cd ativassem suas contramedidas. As setas indicativas “3” representam uma mensagem que pode ser enviada à EB pelos sensores-linfo de forma a avisar ao administrador sobre o ataque que esteja ocorrendo.

5 Experimentos com o SDI Imuno-Inspirado para RSSFs

No SDI proposto, a abordagem de detecção por anomalias foi utilizada, pois em RSSF o custo de se manter uma base de dados em um sensor é maior do que as informações que definem um comportamento normal. Além disso, o SDI proposto é ativo, pois a RSSF deve ser defendida do ataque a fim de preservar seus recursos, gerando alarmes próximos do tempo real. O processamento é contínuo, pois o tempo de detecção e reação a um ataque é importante em uma RSSF, e distribuído entre os sensores da rede, o que contribui para a redução do processamento em um único sensor. No SDI proposto, a coleta é distribuída uma vez que o SDI é distribuído e é realizada uma análise de pacotes da rede, pois não existe espaço suficiente de armazenamento nos sensores para *logs*.

Foram realizadas inicialmente diversas simulações para calibrar o SDI proposto e analisar sua eficiência. Em seguida foram realizadas simulações para avaliar o emprego de diferentes perfis, um priorizando a segurança e outro priorizando a economia de energia, ambos utilizados neste trabalho. Além disso, foi realizada uma comparação com outro trabalho, o qual utiliza uma abordagem diferente para o SDI, no caso a Teoria da Seleção Negativa, permitindo medir e comparar a eficiência das duas abordagens em termos de detecção e consumo de energia. Finalmente, um experimento em uma plataforma real de sensores sem fio foi realizado. Em tal experimento, foi possível avaliar a eficiência do algoritmo customizado das células dendríticas em sensores reais. Este experimento foi repetido no simulador TOSSIM, permitindo comparar os resultados de simulação com os resultados obtidos em uma RSSF real.

Na dissertação foi considerado apenas o ataque *Denial-of-Sleep,* que se caracteriza pela presença de um atacante, chamado de *Jammer*, o qual causa um ruído no meio de comunicação sem fio. Tal ruído atrapalha a comunicação entre os nós da rede, impedindo-os de entrar em modo de “hibernação” devido à inundação do meio com mensagens. O ataque de *Denial-of-Sleep* é considerado como uma das maiores ameaças às RSSFs (MARTYNOV *et al.* 2007). Uma customização realizada refere-se ao cálculo do limiar de migração para a identificação do ataque de *Denial-of-Sleep*. Este cálculo não foi feito utilizando-se a equação 1 do capítulo 2, mas contando-se diretamente o número de mensagens recebidas pelo nó, o que permite respostas mais rápidas para o caso específico de ataques de *jamming* (taxa alta de mensagens na rede). Esta customização possibilitou também uma economia de energia pelos sensores passando a ser realizado pela contagem de mensagens recebidas pelos sensores-cd. Como o *Jammer* envia uma grande quantidade de mensagens por segundo, o limiar é atingido

rapidamente. Caso fosse utilizado um limiar por tempo, por exemplo, o sensor-cd ficaria gastando energia desnecessária apenas para cumprir o tempo de escuta antes da migração. Ou seja, ao invés de possuir um gasto constante de processamento, mesmo que não esteja ocorrendo um ataque na rede, os sensores-cd passam a contabilizar o número de mensagens de aplicação e, ao se atingir um determinado número, enviam uma mensagem para o sensor-linfo contendo o resultado do processamento das informações coletadas. Foi deixada como trabalho futuro a análise da detecção de outros tipos de ataques em RSSFs.

Como contramedida, foi programado o desligamento do rádio dos sensores por um período de tempo programável a fim de evitar que os sensores sendo afetados pelo atacante permanecessem recebendo as mensagens emitidas e, conseqüentemente, gastando energia. Decorrido o período programado, os sensores retomavam sua atividade normal na RSSF.

5.1 Ambiente do experimento

Para todos os experimentos realizados, utilizando sensores reais ou simulados, a RSSF projetada foi composta de sensores MICAz, fabricado pela Crossbow Technology (CROSSBOW 2010). Os sensores foram programados com o ambiente de desenvolvimento TinyOS, versão 2.1.1, usando nesC (LEVIS e GAY 2009), uma extensão da linguagem C, que implementa um modelo de programação orientado a eventos. TinyOS é um arcabouço baseado em componentes, projetado especificamente para o desenvolvimento de soluções para RSSFs. Os experimentos reais foram realizados em um ambiente fechado (laboratório). Os cenários simulados foram realizados com o simulador TOSSIM, que é próprio do TinyOS (LEVIS e GAY 2009). TinyOS oferece vários componentes de software, incluindo componentes que implementam a pilha de protocolo de comunicação. Cada componente TinyOS possui uma interface bem definida, implementada por funções que são caracterizadas como manipuladores de eventos ou controles.

É importante ressaltar que foram utilizados apenas os protocolos padrões de roteamento do TinyOS, os quais são fornecidos pelo ambiente de programação. Não foi utilizada nenhuma placa de sensoriamento, uma vez que o objetivo dos experimentos foi avaliar o SDI proposto.

A aplicação utilizada ao longo de todos os experimentos foi a *BlinkToRadio*, disponível no repositório do TinyOS, a qual foi instalada em todos os sensores. Esta aplicação utiliza uma programação simples para a utilização do rádio dos sensores, suficiente para realizar todos os experimentos propostos. A aplicação *BlinkToRadio* faz com que um sensor

permaneça enviando uma mensagem para outro(s) sensor(es) na rede constantemente e ao receber uma mensagem de outro sensor da rede este acende um *led*. Todos os sensores foram configurados de forma a realizar envios periódicos de mensagens a cada 1 segundo.

5.1.1 Ambiente real

O SDI proposto foi implementado definindo-se dois novos componentes para o TinyOS: o *SDICelulaDendriticaC*, com a funcionalidade de CD, implementado nos nós sensores-cd; e o SDILinfonodoC, com a funcionalidade de linfonodo, implementado nos nós sensores-linfo.

O componente *SDICelulaDendriticaC* (Figura 17) foi projetado de modo a ser usado pelas aplicações no lugar do componente padrão *AMReceiverC*. Este último faz o recebimento de mensagens do TinyOS. Desta forma, todas as mensagens que chegam ao nó sensor são avaliadas pelo componente *SDICelulaDendriticaC* e repassadas de forma transparente para a aplicação em execução no sensor. Este componente possui as seguintes funcionalidades: (i) realizar o monitoramento de parâmetros que identifiquem um ataque; (ii) ativar e desativar o rádio do sensor; (iii) executar o ACD (fases de coleta e análise).

O componente *SDILinfonodoC* (Figura 18) necessita apenas ser conectado à aplicação a ser executada no sensor-linfo. Este é um componente adicional que não interfere no recebimento das mensagens da aplicação. Sua função consiste em receber e tratar as mensagens especiais enviadas pelos sensores-cd. Este componente possui as seguintes funcionalidades: (i) ativar ou desativar o rádio do sensor; (ii) receber dos sensores com o componente *SDICelulaDendriticaC* as mensagens contendo as informações das CDs, contabilizando a quantidade de vezes que os estados maduro e semi-maduro foram recebidos (fase de decisão); (iii) controlar, segundo uma periodicidade determinada pelo administrador da rede, quando o linfonodo deverá calcular o limiar de anomalia (MCAV); (iv) ativar os elementos responsáveis pelas contramedidas (fase de reação).

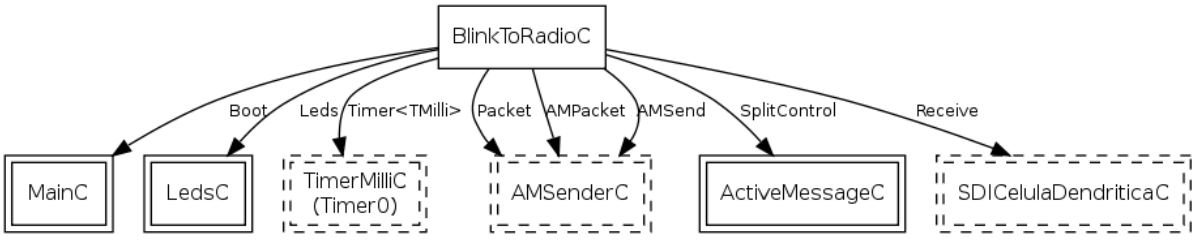


Figura 17. Aplicação utilizando componente SDICelulaDendriticaC.

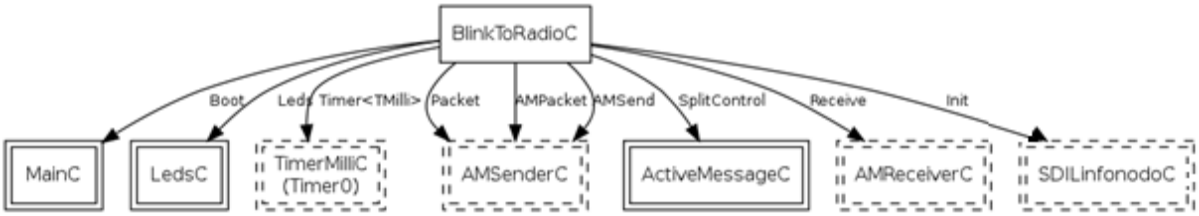


Figura 18. Aplicação utilizando componente SDILinfonodoC.

O nó malicioso (*Jammer*) foi implementado como uma aplicação que utiliza os componentes de comunicação padrão do TinyOS, gerando mensagens na rede a uma taxa pré-definida (dez vezes maior do que o padrão da RSSF), intercalando períodos de ativação e de desativação. Mais informações sobre o ataque escolhido e outros tipos de ataques em RSSF podem ser encontrados em Karlof e Wagner (KARLOF e WAGNER 2003) e Margi *et al.* (MARGI *et al.* 2009).

5.1.2 Ambiente simulado

O simulador TOSSIM possui uma restrição: trabalha apenas com uma imagem de implementação. Ou seja, todos os sensores simulados têm, obrigatoriamente, o mesmo código. Assim, para atender a este pré-requisito e possibilitar os experimentos simulados, foi criado para o ambiente simulado um único código contendo todas as implementações anteriores. Ao iniciar a simulação, por meio de estruturas de decisão programadas no código, os sensores assumiam os papéis de CD, linfonodo ou *Jammer*.

Assim, os códigos utilizados no ambiente real puderam ser reutilizados nas simulações permitindo comparações e testes.

Todos os experimentos simulados tiveram a duração de 100 segundos cada.

5.1.3 Avaliação da quantidade de memória utilizada pelo SDI

O SDI proposto foi implementado na plataforma de sensores reais MICAz (4Kbytes de RAM e 128Kbytes de ROM). No papel de sensor-cd, foram consumidos 116 bytes (2,8%) de memória RAM e 1054 bytes (0,8%) de memória ROM; e no papel de sensor-linfo, foram consumidos 99 bytes (2,4%) de memória RAM e 2956 bytes (2,3%) de memória ROM.

A memória *flash* disponível na plataforma ficou completamente disponível para o sistema. Portanto, a maior parte dos recursos de armazenamento ficou disponível para o Sistema Operacional e para as aplicações.

Existem dois tipos de mensagem de controle: as mensagens transmitidas do sensor-cd para o sensor-linfo, com 3 bytes de tamanho, e as mensagens transmitidas pelo sensor-linfo para os sensores-cd, com 2 bytes de tamanho para que os sensores-cd ativem as contramedidas.

As mensagens de dados da aplicação *BlinkToRadio* possuem 2 bytes de tamanho e não foram alteradas, uma vez que o SDI, a fim de se manter genérico, não deve causar nenhum tipo de alteração nos componentes e mensagens das aplicações.

5.1.4 Modelo de Energia Utilizado

Para avaliar o consumo de energia do IDS foi utilizado um modelo de energia simples, baseado no número de mensagens enviadas e recebidas pelos sensores. Nos sensores de uma RSSF o rádio não fica ligado por todo o tempo, ele permanece ligado apenas por pequenos intervalos de tempo, realizando a recepção e/ou transmissão de mensagens, e em um estado de "hibernação" no restante do tempo. Esta forma de operação garante um tempo de vida maior para o sensor.

A razão pela escolha desta definição provém de uma vertente encontrada na literatura onde a maioria dos autores admitem que os sensores gastam a maior parte de sua energia com comunicação (DIETRICH e DRESSLER 2009). A consequência lógica é que vários autores focam apenas na energia gasta com comunicação quando avaliam a eficiência de algoritmos para RSSF. Desta forma, nos experimentos realizados, foram consideradas as seguintes situações de consumo de energia: transmissão de mensagem e recebimento de mensagem (DA SILVA 2005).

O custo de energia foi modelado como: $Q = Q_{TX} + Q_{RX}$, onde Q_{TX} é a energia consumida na transmissão e Q_{RX} é a energia consumida na recepção. Foi utilizado um modelo de energia baseado nos dados definidos no *datasheet* do sensor MICAz (CROSSBOW 2010). Consideramos a taxa de transmissão do nó de 4 µs/bit, sendo a corrente elétrica que flui pelo nó ao receber um pacote é de 18,8 mA e ao transmitir um pacote é de 17,4 mA.

Na nossa proposta existem 2 tipos de mensagens: mensagens de dados da aplicação (16 bits de tamanho) e de controle do SDI. As mensagens de controle do SDI são divididas em dois tipos: aquelas transmitidas pelo sensor-cd (24 bits de tamanho) e aquelas transmitidas pelo sensor-linfo (16 bits de tamanho). Para cada um dos tipos de mensagem foi calculada a energia dissipada de transmissão (Q_{TX}) e de recepção (Q_{RX}), como mostrado abaixo.

Custo de energia das mensagens de dados da aplicação geradas nos sensores:

$Q_{TX} = 3\ (V) * 17{,}4\ (mA) * 4\ (\mu s/bit) * 16\ (bits) = 3{,}3408\ mJ/mensagem$ (1)

$Q_{RX} = 3\ (V) * 18{,}8\ (mA) * 4\ (\mu s/bit) * 16\ (bits) = 3{,}6096\ mJ/mensagem$ (2)

Custo de energia das mensagens de controle do SDI geradas por um sensor-cd:

$Q_{TX} = 3\ (V) * 17{,}4\ (mA) * 4\ (\mu s/bit) * 24\ (bits) = 5{,}0112\ mJ/mensagem$ (3)

$Q_{RX} = 3\ (V) * 18{,}8\ (mA) * 4\ (\mu s/bit) * 24\ (bits) = 5{,}4144\ mJ/mensagem$ (4)

Custo de energia das mensagens de controle do SDI geradas por um sensor-linfo:

$Q_{TX} = 3\ (V) * 17{,}4\ (mA) * 4\ (\mu s/bit) * 16\ (bits) = 3{,}3408\ mJ/mensagem$ (5)

$Q_{RX} = 3\ (V) * 18{,}8\ (mA) * 4\ (\mu s/bit) * 16\ (bits) = 3{,}6096\ mJ/mensagem$ (6)

Onde: Energia Dissipada (Q) = Voltagem (V) x Corrente Elétrica (mA) x Tempo (s), sendo Tempo = Taxa de Transmissão x Tamanho da Mensagem.

No Anexo 1 encontram-se comparações entre as ferramentas PowerTOSSIMZ e AvroraZ, específicas para a medição do consumo de energia em RSSF compostas por sensores modelo MICAz, e uma justificativa para a sua não utilização neste trabalho.

5.1.5 Descrição das métricas

As métricas utilizadas nos experimentos foram falsos positivos (FP), falsos negativos (FN), verdadeiros positivos (VP), verdadeiros negativos (VN), sensibilidade e especificidade, definidas a seguir.

Os falsos positivos (FP) indicam a quantidade de alarmes gerados pelo SDI quando não está ocorrendo um ataque e os falsos negativos (FN) indicam uma condição de normalidade quando na verdade está ocorrendo um ataque. Os verdadeiros positivos (VP) indicam uma condição de anomalia quando está ocorrendo um ataque e os verdadeiros negativos (VN) indicam uma condição de normalidade quando não está ocorrendo nenhum ataque.

A sensibilidade representa a taxa de acerto do SDI e é calculada pela razão entre a quantidade de VP e a soma de VP e FN, ou seja, Sensibilidade=VP/(VP+FN).

A especificidade representa a taxa de alarmes falsos e é calculada pela razão entre a quantidade de VN e a soma dos VN e FP, ou seja, Especificidade=VN/(VN+FP). Estas duas métricas são apenas para ambientes simulados.

Estas métricas foram geradas pelo SDI durante sua execução para um limiar de anomalia (MCAV) igual a 50%, configurado no sensor-linfo, conforme definido na literatura

(GREENSMITH *et al.* 2007). Ou seja, para todos os MCAV emitidos pelo sensor-linfo maiores ou iguais a 50%, o SDI proposto indicou a presença de intruso.

5.2 Descrição do cenário

Para os experimentos realizados, adotou-se uma rede de topologia plana e com nós fixos tanto para a implementação real quanto para as simulações.

A RSSF foi composta por: (i) nós sensores-cd, com o componente *SDICelulaDendriticaC* instalado; (ii) nós sensores-linfo, com o componente *SDILinfonodoC*; e (iii) um nó sensor *Jammer*. Foi utilizado um intervalo de 100 ms para o envio das mensagens do *Jammer*. O *Jammer* foi posicionado a 1 metro do sensor-linfo na calibração (seção 5.3) e suas mensagens possuíam o mesmo cabeçalho das mensagens da aplicação, mas sem conteúdo (*payload*) válido, a fim de causar um recebimento por todos os sensores da rede e, conseqüentemente, um desperdício de energia. Este *Jammer* é conhecido pelo nome de *Deceptive jammer* (XU *et al.* 2005).

Para aplicar o ACD customizado para RSSFs para a detecção do ataque de *Denial-of-sleep*, os antígenos e os sinais de entrada foram definidos e mensurados a partir das mensagens recebidas pelos sensores-cd da seguinte forma: (i) **antígeno**, definido como a identificação do nó emissor da mensagem (campo de origem no cabeçalho da mensagem); (ii) sinal de **PAMP**, definido como sendo o nível do RSSI presente no meio quando o sensor-cd recebe uma mensagem (o *Jammer* ao emitir mensagens, utiliza uma potência mais elevada para conseguir atingir a maior quantidade de sensores dentro de seu alcance); (iii) sinal de **perigo**, obtido calculando a taxa das mensagens recebidas pelo sensor-cd (o *Jammer* ao emitir mensagens, o faz utilizando uma taxa de envio maior do que um sensor normal da rede); e (iv) sinal **seguro**, definido como sendo o inverso da variação da taxa das mensagens recebidas pelo sensor-cd (em uma rede normal a variação da taxa é pequena, portanto seu inverso é alto, enquanto que em uma rede com um *Jammer*, a variação da taxa é alta e, portanto, seu inverso é pequeno). É importante ressaltar que, apesar da arquitetura ser genérica para qualquer tipo de ataque, o mapeamento dos sinais de entrada é específico para cada ataque.

Segundo Silva (SILVA 2009), os sinais, para serem processados no algoritmo requerem normalização, pois representam grandezas de valores diferentes. No entanto, deve ser escolhida uma técnica de normalização adequada para que o algoritmo produza resultados condizentes. Para isso, os dados devem ser adaptados a uma faixa de valores extremos. Isso pode ser feito através da delimitação de uma faixa de valores, onde os valores abaixo da faixa

assumem o valor mínimo e os valores acima assumem o máximo. Com a normalização, os valores intermediários são normalizados entre os valores máximos. Esta abordagem é considerada viável, principalmente considerando aplicações em tempo real ou sistemas cujas variações numéricas são altas.

Segundo Xu *et al.* (XU *et al.* 2005), essas métricas, utilizadas para os sinais de entrada, são as mais indicadas para a identificação de *Jammer*.

Na equação 1 do Capítulo 2, os pesos foram definidos empiricamente a partir de experimentos imunológicos conduzidos por imunologistas do "*The Danger Project*" (GREENSMITH 2007). Assim, para o sinal de saída semi-maduro os pesos W*p*, W*d* e W*s* assumem, respectivamente, os valores 0, 0 e 1. Para o sinal de saída maduro, os pesos W*p*, W*d* e W*s* assumem, respectivamente, os valores 2, 1 e -3. A inflamação não foi considerada neste trabalho a fim de diminuir ao máximo a carga de processamento nos sensores.

5.3 Simulações

Nesta seção são apresentados os experimentos de simulação realizados no trabalho.

5.3.1 Calibração do SDI

Para efeito de calibração, os sensores-cd foram dispostos ao longo de um círculo de 3 metros de diâmetro de forma a permanecerem eqüidistantes do sensor-linfo, o qual ficou no centro deste círculo. Todos os nós foram programados de forma a possuir uma identificação única e um alcance de rádio omnidirecional fixo de 15 metros.

Durante a calibração, o *Jammer* foi posicionado a uma distância de 1 metro do sensor-linfo e comportava-se como uma função degrau, mantendo-se ativo por 10 segundos (enviando mensagens de aplicação sem conteúdo útil) e desativado por outros 10 segundos, e assim sucessivamente. Importante observar que, como o limiar de migração controla o número de mensagens recebidas pelos sensores, o atraso referente ao recebimento destas mensagens deve ser menor do que os 10 segundos que o *Jammer* permanece ativo.

Em uma RSSF os dados estão sujeitos a diversos tipos de erros, que podem ser gerados devido a uma incorreta calibração dos parâmetros do ACD customizado (número de sensores-cd, limiar de migração e intervalo de verificação do MCAV) além das características intrínsecas dos dispositivos de sensoriamento, características topológicas, cobertura, tempo de vida da rede ou por fatores ambientais e que afetam a acurácia fornecida. Assim, a principal meta da calibração é maximizar a relevância do SDI para a aplicação e, ao mesmo tempo,

minimizar a energia consumida pelo SDI, sujeito a restrições de cobertura, conectividade e energia da rede.

5.3.2 Variação dos parâmetros do ACD Personalizado

Três conjuntos de experimentos foram realizados. No primeiro conjunto, foi determinado o número ideal de sensores-cd que um sensor-linfo precisa em termos de sensibilidade e especificidade na detecção de um ataque *Denial-of-sleep*. O **número de sensores-cd por sensor-linfo** variou de 1 até 10 sensores-cd por sensor-linfo, de 1 em 1 sensor-cd, mantendo-se a densidade de sensores por metro quadrado. O segundo conjunto determinou o valor mais apropriado para o limiar de migração em termos de número de mensagens recebidas por um sensor-cd. O **limiar de migração das CDs** foi variado de 1 até 10 mensagens, de 1 em 1 mensagem. O terceiro conjunto determinou qual o melhor intervalo de verificação do MCAV que um sensor-linfo deveria utilizar de forma a obter o resultado mais eficiente na detecção de um atacante. O **intervalo de verificação do MCAV** foi variado de 1 até 10 segundos, de 1 em 1 segundo.

A - Variação do número de sensores-cd por sensor-linfo

Neste experimento foi variado o número de sensores-cd por sensor-linfo de 1, que é a menor quantidade de sensor-cd, até 10 sensores-cd, de 1 em 1 sensor-cd. Acima de 10 sensores-cd foi verificado que os FN e FP permaneceram zerados, portanto este valor foi o máximo utilizado. A Tabela 6 mostra os valores de FP e FN resultantes em percentuais, fixando o limiar de migração em 1 mensagem e o intervalo de verificação do MCAV igual a 5 segundos.

Observa-se na Tabela 6 que o aumento do número de sensores-cd por sensor-linfo implica na diminuição dos valores FN e FP, tornando o sistema mais eficiente.

Tabela 6. FN e FP para limiar de migração igual a 1 e intervalo de verificação de MCAV igual a 5 segundos.

Métricas	Sensores-cd									
	1CD	2CD	3CD	4CD	5CD	6CD	7CD	8CD	9CD	10CD
FN	31%	13%	3%	1%	0%	0%	0%	0%	0%	0%
FP	5%	0%	0%	0%	0%	0%	0%	0%	0%	0%

Observa-se também na Tabela 6 que os experimentos com 1, 2 e 3 sensores-cd apresentaram valores percentuais elevados de FN ou FP e, portanto, não devem ser considerados. Os resultados para 1, 2 e 3 sensores-cd por sensor-linfo foram descartados por não representaram números significativos de CDs a fim de compor uma quantidade mínima confiável para uma população de CDs (GREENSMITH *et al.* 2006).

B - Variação do limiar de migração das CDs

Nesta seção foram realizados sete experimentos, um experimento para cada quantidade de sensores-cd por sensor-linfo. Assim, o primeiro experimento foi realizado para a quantidade de sensores-cd igual 4, um segundo experimento para 5 sensores-cd e assim por diante até a quantidade de 10 sensores-cd por sensor-linfo. Para cada um desses experimentos foram realizadas variações para o limiar de migração de 1 a 10 mensagens, de 1 em 1 mensagem.

O objetivo desses experimentos foi determinar, em função dos FN e FP, a quantidade de mensagens que os sensores-cd deveriam analisar antes de definir o estado de maturação da CD para em seguida enviar essa informação para o sensor-linfo.

Na Tabela 7, são mostrados os resultados para os FN e FP emitidos pelo sensor-linfo para avaliações realizadas em intervalos de 5 segundos.

Nos experimentos realizados, considerou-se que, para a aplicação adotada nesse trabalho, 1% de ocorrência de FN ou FP era um valor aceitável para tais métricas. Observa-se na Tabela 7 que, para o experimento 1, foram obtidos os melhores valores tanto para FP quanto FN para limiares de migração com valores iguais a 1 ou 2 e para os experimentos 2, 3 e 4, foram obtidos os melhores valores tanto para FP quanto FN para limiares de migração no intervalo de 1 a 4. Os experimentos 5 e 6 mostram os melhores valores de FN e FP para limiares de migração entre 1 e 9 e 1 e 10 mensagens, respectivamente. O experimento 7 apresenta os melhores valores de FN e FP (ambos iguais a zero) para todos os limiares de migração utilizados. Em cada experimento, observa-se que com o aumento do limiar de migração ocorre um aumento nos valores percentuais de FN. Isto ocorre porque os sensores-cd passam a ter a oportunidade de coletar uma quantidade maior de antígenos e de sinais de entrada que pertencem tanto ao atacante quanto aos que pertencem aos sensores próprios da RSSF. Além disso, um limiar de migração maior faz com que um sensor-cd demore mais tempo para enviar uma informação ao sensor-linfo prejudicando sua decisão. Por outro lado,

ao aumentar o número de sensores-cd, o ACD conta com o resultado emitido por diversos outros sensores-cd para que uma resposta correta seja emitida em caso de que algum sensor-cd classifique um antígeno erroneamente. Para quantidades menores de sensores-cd, o peso desta informação errada é maior gerando valores maiores de FN. Desta forma percebe-se a importância de uma população de CDs e a robustez do ACD.

Tabela 7. FN e FP para variação do limiar de migração com intervalo de verificação do MCAV igual a 5 segundos.

Experimento	Sensor-cd	Métricas	Limiar de Migração									
			1	2	3	4	5	6	7	8	9	10
1	4CD	FN	1%	1%	3%	3%	6%	7%	7%	7%	7%	7%
		FP	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
2	5CD	FN	0%	0%	1%	1%	2%	2%	2%	2%	2%	2%
		FP	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
3	6CD	FN	0%	1%	1%	1%	3%	4%	6%	6%	7%	8%
		FP	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
4	7CD	FN	0%	0%	1%	1%	2%	3%	5%	5%	5%	5%
		FP	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
5	8CD	FN	0%	0%	0%	0%	0%	1%	1%	1%	1%	2%
		FP	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
6	9CD	FN	0%	0%	0%	0%	0%	0%	0%	0%	0%	1%
		FP	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
7	10CD	FN	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%
		FP	0%	0%	0%	0%	0%	0%	0%	0%	0%	0%

É importante ressaltar que a atribuição de um valor alto para o limiar de migração ocasiona uma detecção tardia de um evento de anomalia, justificando o aparecimento de FN e FP. Em contrapartida, a atribuição de um valor baixo implica na geração de um número maior de mensagens de controle enviadas dos sensores-cd para o sensor-linfo, gerando um maior gasto de energia. Caso o atraso na detecção não seja um requisito da aplicação, poderia ser utilizado um limiar de migração igual a 10. Caso contrário, o limiar de migração igual a 1 mensagem deveria ser utilizado para uma detecção mais rápida.

C - Variação do intervalo de verificação do MCAV

Nesta seção foram realizados sete experimentos, um experimento para cada quantidade de sensores-CD por sensor-linfo. Assim, o primeiro experimento foi realizado para 4 sensores-cd, um segundo experimento para 5 sensores-cd e assim por diante até a quantidade de 10 sensores-cd. Para cada um desses experimentos, foram realizadas variações do intervalo

de verificação do MCAV no sensor-linfo, de 1 a 9 segundos, de 1 em 1 segundo. O objetivo desses experimentos foi determinar, em função dos FN e FP, o intervalo de verificação do MCAV que melhor identificasse um ataque. Para cada quantidade de sensor-cd, a Tabela 8 mostra os valores de FP e FN em percentuais variando o intervalo de verificação do MCAV e fixando o limiar de migração igual a 1 mensagem.

Vale destacar que a diversidade das CDs de uma população de células é gerada por meio da migração das células maduras e semi-maduras em tempos diferentes, criando o efeito de uma janela de tempo (GREENSMITH *et al*. 2007), ou seja, diferentes CDs coletam informações diferentes em tempos diferentes. No SDI proposto, o efeito de uma janela de tempo foi alcançado com a presença de vários sensores-cd monitorando e percebendo as diferentes condições da RSSF e pela determinação do intervalo de verificação do MCAV no sensor-linfo, aliado ao fato de cada sensor-cd realizar sua própria coleta de antígenos e sinais de entrada.

Observa-se na Tabela 8 que, com o aumento do intervalo de verificação de MCAV, ocorre uma diminuição nos valores percentuais de FN para os valores de 1 a 6 segundos e, a partir de 7 segundos, ocorre o inverso. Valores de intervalo de verificação do MCAV maiores do que 6 segundos indicam que não houve migração de uma quantidade suficiente de CD maduras (anômalas) para o sensor-linfo, implicando em uma avaliação incorreta por parte do SDI sobre a presença de um atacante. O comportamento dos FP é similar ao comportamento dos FN, porém, o ponto onde seus valores começam a aumentar é em 5 segundos.

Tabela 8. FN e FP variando o intervalo de verificação do MCAV e fixando o limiar de migração igual a 1 mensagem.

Experimento	Sensor-cd	Métricas	Intervalo de verificação do MCAV (segundos)								
			1	2	3	4	5	6	7	8	9
1	4CD	FN	10%	9%	5%	1%	1%	1%	5%	6%	10%
		FP	1%	1%	1%	0%	0%	32%	49%	48%	63%
2	5CD	FN	7%	4%	2%	0%	0%	0%	5%	7%	9%
		FP	1%	1%	1%	0%	0%	35%	51%	49%	66%
3	6CD	FN	7%	4%	2%	0%	0%	0%	6%	7%	10%
		FP	1%	0%	0%	0%	0%	34%	49%	49%	64%
4	7CD	FN	5%	4%	2%	0%	0%	0%	6%	7%	10%
		FP	0%	0%	0%	0%	0%	36%	50%	50%	65%
5	8CD	FN	4%	1%	1%	0%	0%	0%	6%	7%	11%
		FP	0%	0%	0%	0%	0%	37%	51%	50%	66%
6	9CD	FN	3%	1%	1%	0%	0%	0%	6%	7%	11%
		FP	0%	0%	0%	0%	0%	37%	51%	50%	66%
7	10CD	FN	2%	1%	1%	0%	0%	0%	6%	8%	10%
		FP	0%	0%	0%	0%	0%	38%	52%	50%	67%

Considerado 1% um valor aceitável para os FN e FP, constata-se para os experimentos da Tabela 8 que, para a faixa de 4 a 10 sensores-cd, foram obtidos os melhores valores de FP e FN quando é utilizado um intervalo de verificação de MCAV de 4 ou 5 segundos. Considerando que os resultados são os mesmos para 4 ou 5 segundos, foi escolhido um intervalo de verificação do MCAV de 5 segundos de forma a economizar energia.

5.3.3 Eficiência do SDI

A fim de extrair informações sobre a eficiência do SDI proposto neste trabalho, os resultados obtidos nas simulações foram analisados por meio das curvas ROC (*Receiver Operating Characteristics*). Estas curvas possuem como característica fundamental a distinção entre taxa de acerto (sensibilidade ou percentual de verdadeiro positivo) e taxa de alarme falso (especificidade ou percentual de falsos positivos) como duas medidas de desempenho. Estas curvas são normalmente empregadas para medição de eficiência para detectores de intrusos baseados em anomalias (SILVA 2009). Os valores de VP, VN, FP e FN foram usados na composição das curvas de sensibilidade e especificidade (SILVA 2009).

A partir dos valores de especificidade e sensibilidade observados, os melhores valores para o MCAV foram encontrados. A Figura 19, a Figura 20 e a Figura 21 mostram os resultados para os experimentos realizados com 5, 7 e 10 sensores-cd, respectivamente. Estes experimentos foram realizados com limiar de migração igual a 1 mensagem e intervalo de verificação de MCAV igual a 5 segundos. Observa-se que a especificidade cresce com o aumento do valor do limiar de anomalia, significando que um limiar baixo pode ocasionar FP. Por outro lado, a sensibilidade começa com um valor igual a 1 e decresce com o aumento do MCAV, causando aumento nos FN.

A melhor configuração de parâmetros em termos de detecção de intrusos é aquela que apresenta os valores mais altos, tanto da sensibilidade quanto da especificidade, ou seja, quando ambos os valores são iguais a 1, ocasionando uma interseção entre as curvas. No que concerne à sensibilidade, pode-se observar que, conforme o número de sensores-cd por sensor-linfo aumenta, a faixa de valores de limiar de anomalia que possui valor igual a 1 também aumenta. Por exemplo, para 5 sensores-cd, os valores de MCAV que se encontram no intervalo de 0 a 50% possuem valor igual a 1 para sensibilidade. Já para 10 sensores-cd, os valores de MCAV que possuem valor igual a 1 se encontram no intervalo de 0 a 60%. Quanto à especificidade, as figuras mostram que alcançamos especificidade igual a 1 para valores de

MCAV acima de 30%, para o caso de 5 sensores-cd. Já para o caso de 7 e 10 sensores-cd, o MCAV pode ser maior ou igual a 20% para um valor de especificidade igual a 1. Desta forma, o MCAV de valor 50% foi escolhido para ser utilizado, pois é o valor máximo que atende simultaneamente a todas estas quantidades de sensores-cd.

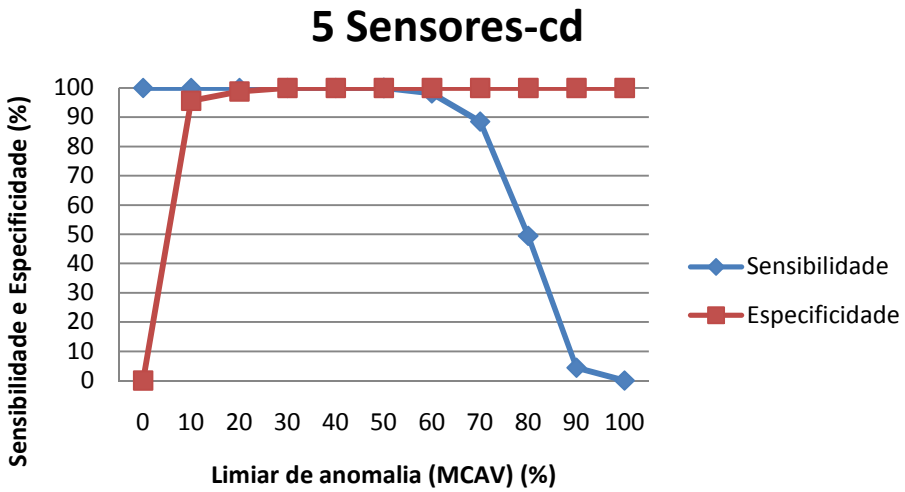


Figura 19. Curvas ROC para 5 sensores-cd.

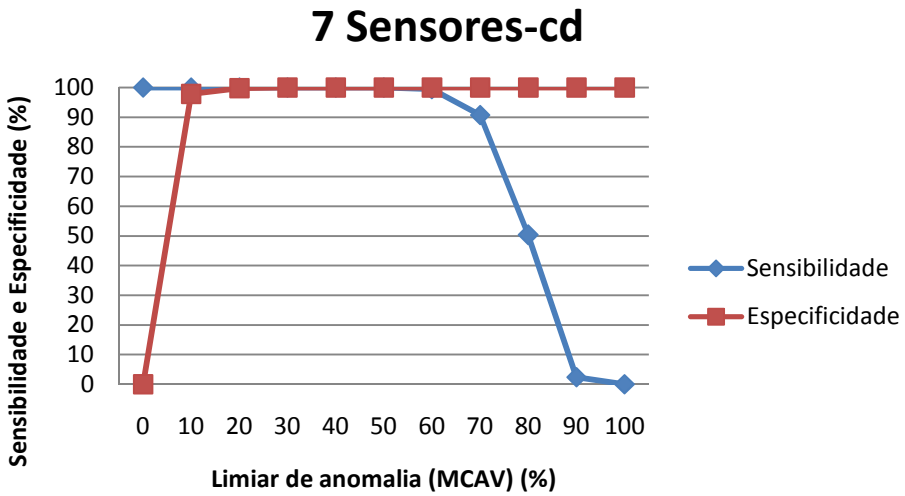


Figura 20. Curvas ROC para 7 sensores-cd.

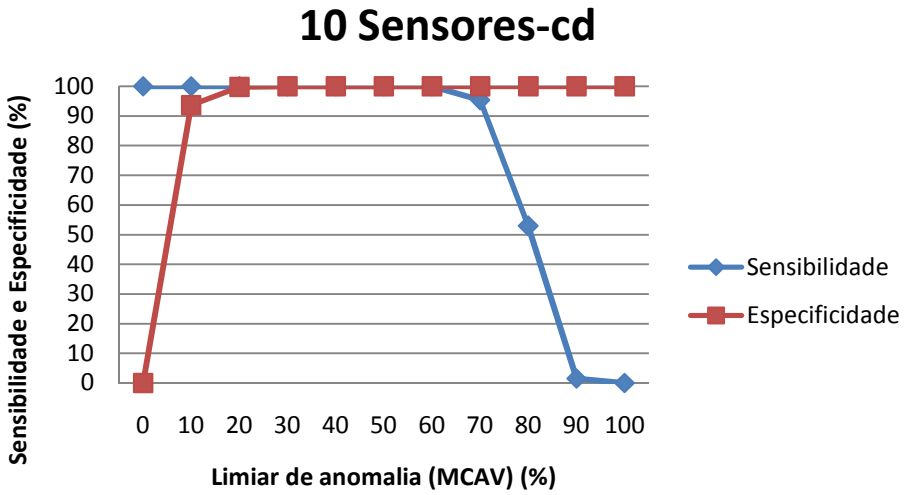


Figura 21. Curvas ROC para 10 sensores-cd.

5.3.4 Experimentos de energia simulados

Para avaliar o custo do SDI proposto em termos de energia, foram realizados 12 experimentos. Os seis primeiros experimentos utilizaram os mesmos valores para limiares de migração (10 mensagens), intervalo de emissão de MCAV (5 segundos) e quantidade de sensores-CD por sensor-linfo (10 sensores). Tais valores garantem uma maior segurança da RSSF uma vez que os valores de FP e FN são iguais a zero, conforme pode ser verificado nos resultados do Experimento 7 da Tabela 7, sem se preocupar com consumo de energia. Os seis últimos experimentos utilizaram os mesmos valores para limiares de migração (10), intervalo de emissão de MCAV (5 segundos) e quantidade de sensores-CD por sensor-linfo (5). Estes valores foram escolhidos de forma a diminuir o consumo de energia admitindo-se um erro de 1% para FN e FP, conforme pode ser verificado nos resultados do Experimento 2 da Tabela 7.

No primeiro, segundo, sétimo e oitavo experimento, o *Jammer* estava desativado. No terceiro, quarto, nono e décimo experimentos o *Jammer* não teve seu funcionamento interrompido, ou seja, considerou-se que o *Jammer* permaneceu ativo do início até o final da simulação. No quinto, sexto, décimo primeiro e décimo segundo experimentos o *Jammer* permaneceu ligado do início até a metade do tempo da simulação, sendo desligado da metade até o final da mesma, ou seja, considerou-se que o *Jammer* foi identificado após um alarme emitido pelo SDI e retirado da rede, permitindo que a RSSF voltasse ao normal.

As diferentes configurações dos 12 experimentos levaram em conta: (i) os parâmetros do ACD; (ii) o período de ativação do *Jammer*; e (iii) SDI ativado ou não.

Os parâmetros relacionados ao ACD foram: o limiar de migração das CDs (LM), o intervalo de verificação do MCAV (LN) e a quantidade de sensores-cd por sensor-linfo (CD). Estas configurações foram agrupadas e denotadas por A1 e A2, onde:

- A1: LM = 10 mensagens; LN = 5 segundos; e CD = 10 sensores-cd.
- A2: LM = 10 mensagens; LN = 5 segundos; e CD = 5 sensores-cd.

As configurações relacionadas ao período de ativação do *Jammer* foram denotadas por B1, B2 e B3, onde:

- B1: *Jammer* desativado.
- B2: *Jammer* ativo 100% do tempo de simulação.
- B3: *Jammer* ativo 50 % do tempo da simulação.

As configurações relacionadas à presença ou não de um *Jammer* em uma RSSF utilizando ou não o SDI proposto foram denotadas por C1, C2 e C3, onde:

- C1: **RSSF sem Ataque e sem SDI -** foi analisado o consumo normal de energia da rede quando está sendo executada a aplicação de *BlinkToRadio*, sem presença de ataque e nenhum nó sensor desempenhando papéis de célula dendrítica ou linfonodo.
- C2: **RSSF sem Ataque e com SDI** - foi analisado o consumo extra de energia gerado pela adição dos papéis de célula dendrítica e linfonodo aos sensores da rede quando está sendo executada a aplicação de *BlinkToRadio*, sem presença de ataques.
- C3: **RSSF com Ataque e sem SDI -** foi analisado o consumo extra de energia gerado pela adição de um sensor no papel de *Jammer* à RSSF. Este sensor executou a aplicação *BlinkToRadio* com uma taxa de emissão 10 vezes maior do que os sensores da rede.
- C4: **RSSF com Ataque e com SDI -** foi analisado o consumo extra de energia gerado pela adição à rede de um sensor no papel de *Jammer*, executando a aplicação *BlinkToRadio* com uma taxa de emissão 10 vezes maior do que os sensores da rede, e pela adição dos papéis de célula dendrítica e linfonodo aos sensores da rede quando está sendo executada a aplicação de *BlinkToRadio*.

Os resultados obtidos foram agrupados segundo a configuração A e gerados dois gráficos, um seguindo a configuração A1 (Figura 22) e o outro seguindo a configuração A2 (Figura 23). A Tabela 9 consolida os resultados obtidos.

A **primeira barra** representa o consumo de energia da RSSF para as configurações B1 e C1 (Sem Ataque, Sem SDI). A **segunda barra** representa o consumo de energia da RSSF para a configuração B1 e C2 (Sem Ataque, Com SDI). A **terceira barra** representa o consumo de energia da RSSF para as configurações B2 e C3 (Com Ataque, sem SDI, período de ativação do *Jammer* 100%), permitindo avaliar o impacto em termos do consumo de energia da aplicação (sem SDI) quando o período de ativação do *Jammer* é de 100%. A **quarta barra** representa o consumo de energia da RSSF para a configuração B2 e C4 (Com Ataque, Com SDI, período de ativação do *Jammer* 100%), permitindo que seja observada a eficiência de um SDI na presença de um *Jammer* quando o período de ativação do mesmo é de 100%. A **quinta barra** representa o consumo de energia da RSSF para a configuração B3 e C3 (Com Ataque, sem SDI, período de ativação do *Jammer* 50%), permitindo avaliar o impacto em termos do consumo de energia da aplicação quando o período de ativação do *Jammer* é de 50%. Finalmente, a **sexta barra** representa o consumo de energia da RSSF para a configuração B3 e C4 (Com Ataque, Com SDI, período de ativação do *Jammer* 50%), na presença de um atacante e com SDI implementado nos sensores, permitindo que seja observada a eficiência de um SDI na presença de um *Jammer* quando o período de ativação do mesmo é de 50%.

A **primeira barra** (configuração **B1** e **C1**) mostra o consumo de energia da RSSF apenas com a execução da aplicação durante todo o tempo da simulação. Essa primeira barra é a menor das seis barras em termos de energia consumida e foi utilizada como *baseline* para as outras barras. Para o primeiro gráfico (A1), ilustrado na Figura 22, a energia consumida quando somente a aplicação está em operação na RSSF é igual a $6{,}37 \times 10^6$ mJ. No segundo gráfico (A2), ilustrado na Figura 23, a energia consumida foi de $3{,}63 \times 10^6$ mJ. Observa-se que, ao analisar a primeira barra de ambos os gráficos, a energia consumida pela rede contendo 5 sensores-cd por sensor-linfo foi 57% menor em relação a energia consumida quando existem 10 sensores-cd por sensor-linfo. A redução da energia consumida da rede já era esperada uma vez que o número de mensagens enviadas para o sensor-linfo é menor.

A **segunda barra** (configuração **B1** e **C2**) mostra qual o consumo de energia que o SDI incutiu à RSSF com o envio, recebimento e escuta das mensagens do SDI. No primeiro gráfico (A1), a energia consumida na configuração **B1** e **C2** foi **9,38%** maior do que a consumida pela configuração **B1** e **C1**. No segundo gráfico (A2), a energia consumida na configuração **B1** e **C2** foi **10%** maior do que a consumida pela configuração **B1** e **C1**. Pela comparação com as barras onde o SDI está desativado e o Jammer está ativado com as barras

onde ambos estão ativados, podemos observar que o SDI proposto economizou uma grande quantidade de energia, conforme explicado nos parágrafos adiante, comprovando a viabilidade de seu uso.

A **terceira barra** (configuração **B2 e C3)** mostra o maior consumo de energia da RSSF uma vez que o SDI não estava implementado nos sensores e, portanto, nenhuma contramedida (desligar sensores) foi efetuada de forma a impedir o ataque de *Denial-of-sleep*, o qual atuou por todo o tempo da simulação. No primeiro gráfico (configuração A1), a energia consumida foi 294% maior do que na configuração **B1** e **C1**. No segundo gráfico (A2), a energia consumida foi 285% maior do que na configuração **B1** e **C1**.

A **quarta barra** (configuração **B2 e C4)** mostra a energia economizada pela rede, se comparada com a configuração **B2** e **C3**, ao serem ativadas pelo SDI as contramedidas após o ataque ter sido identificado. Estas contramedidas consistiram em desligar os rádios dos sensores afetados pelo *Jammer* durante um período de tempo pré-determinado (igual a 20 segundos) e, assim, protegê-los contra o alto consumo de energia resultante do ataque e garantindo um maior tempo de vida da rede. No primeiro gráfico (A1), a energia consumida na configuração **B2** e **C4** foi 75% maior do que na configuração **B1** e **C1**. No segundo gráfico (A2), a energia consumida na configuração **B2** e **C4** foi 80% maior do que na configuração **B1** e **C1**.

A **quinta barra** (configuração **B3 e C3)** ilustra o consumo de energia da RSSF quando o SDI não estava implementado nos sensores e o período de ativação do *Jammer* foi de 50% do tempo da simulação. No primeiro gráfico (A1), a energia consumida na configuração **B3** e **C3** foi 118% maior do que na configuração **B1** e **C1**. No segundo gráfico (A2), a energia consumida na configuração **B3 e C3** foi 114% maior do que na configuração **B1** e **C1**.

A **sexta barra** (configuração **B3 e C4)** ilustra a ação do SDI proposto, ativando as contramedidas após identificar o *Jammer* quando o período de ativação foi de 50% do tempo da simulação. Estas contramedidas consistiram em desligar os rádios dos sensores afetados pelo *Jammer* durante um período de tempo pré-determinado (igual a 20 segundos) e, assim, protegê-los contra o alto consumo de energia resultante do ataque e garantindo um maior tempo de vida da rede. A configuração **B3 e C4** mostra a energia economizada pela rede se comparada com a configuração **B3 e C3** ao serem ativadas pelo SDI as contramedidas. No primeiro gráfico (A1), a energia consumida na configuração **B3 e C4** foi 48% maior do que a da configuração **B1 e C1**. No segundo gráfico (A2), a energia consumida na configuração **B3 e C4**foi 37% maior do que a da configuração **B1 e C1**.

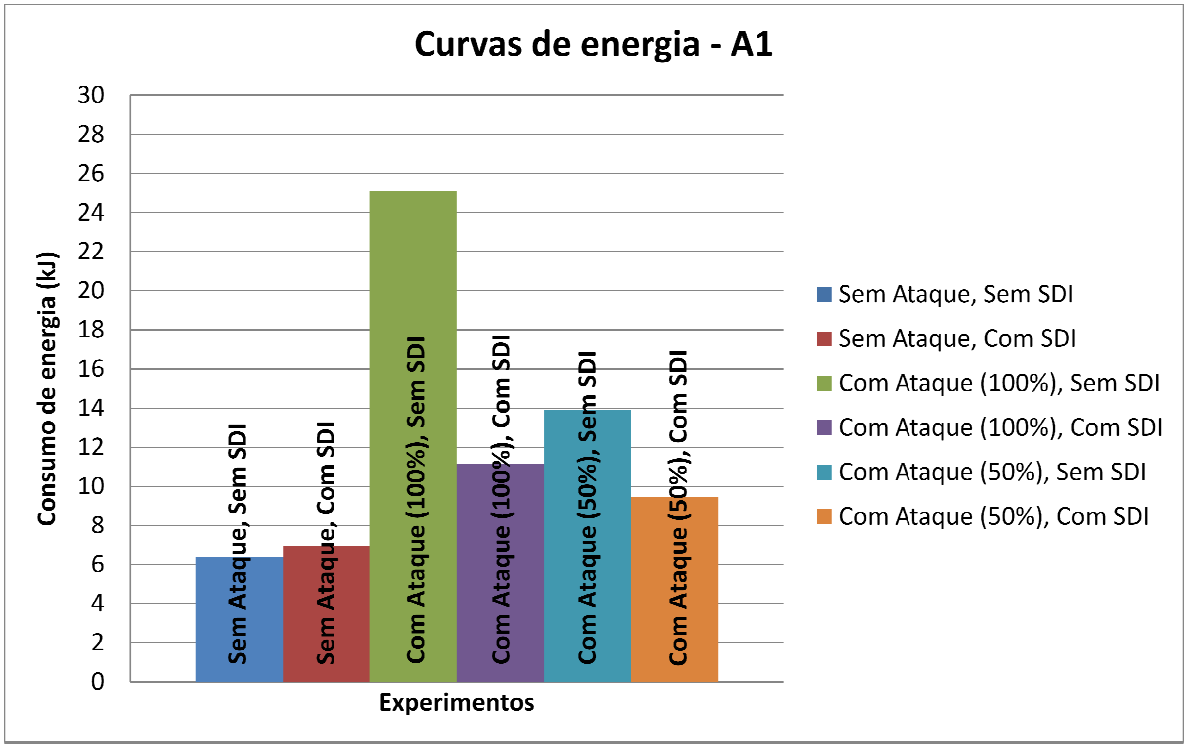


Figura 22. Curvas de energia A1.

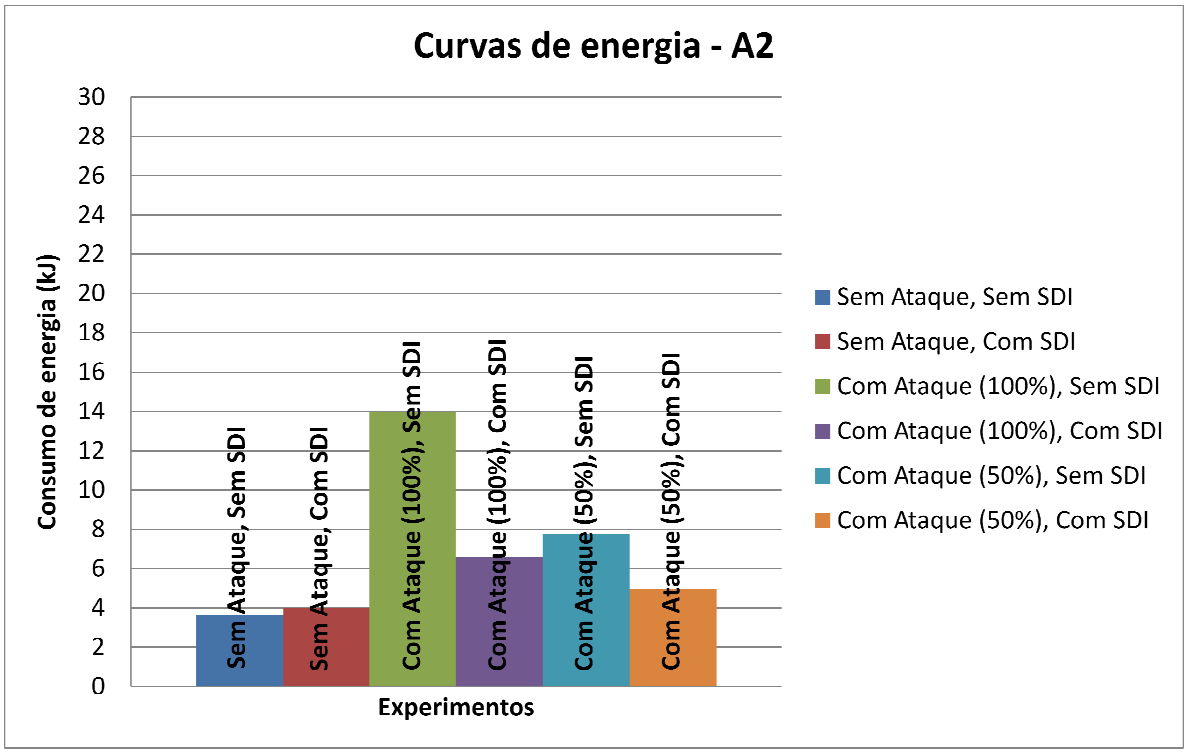


Figura 23. Curvas de energia A2.

Tabela 9. Consumo de energia.

Experimentos	Configurações			Resultados (mJ x 10^6)
1	**A1**	**B1**	**C1**	6,37
2			**C2**	6,97
3		**B2**	**C3**	25,11
4			**C4**	11,17
5		**B3**	**C3**	13,89
6			**C4**	9,46
7	**A2**	**B1**	**C1**	3,63
8			**C2**	3,99
9		**B2**	**C3**	13,98
10			**C4**	6,56
11		**B3**	**C3**	7,77
12			**C4**	4,97

É importante mencionar que, em uma rede densa, nem todos os sensores necessitariam de um SDI instalado, permitindo poupar energia da RSSF.

Pelos resultados obtidos, o experimento onde o SDI utilizou 5 sensores-cd apresentou uma economia de energia de 57% se comparado ao experimentos onde foram utilizados 10 sensores-cd, comprovando a economia com menos sensores-cd.

5.3.5 Atraso

O atraso para a identificação de um ataque está intimamente relacionado com o intervalo de verificação do MCAV realizado por um sensor-linfo e com a quantidade de mensagens recebida dos sensores-cd nesse intervalo. Ou seja, quanto menor este intervalo, mais rápido o sensor-linfo identifica a presença de um ataque, desde que tenha recebido uma quantidade de células dendríticas maduras suficiente para isso. À medida que o intervalo de verificação do MCAV aumenta, maior será o tempo que a rede ficará aguardando por uma decisão de presença de ataque de um sensor-linfo.

Lembrando que o *Jammer* funciona como uma função degrau que permanece alternando entre períodos de atividade e inatividade a cada 10 segundos, e que o intervalo de verificação do MCAV, o qual indica presença de intruso para valores iguais ou maiores do que 50, foi mantido fixo e igual a 5 segundos. Estas configurações tornaram possível para o sensor-linfo verificar os FN e dos FP duas vezes dentro de um intervalo de 10 segundos. Desta forma, passaram a existir duas possibilidades de identificação do intruso: na primeira verificação do MCAV (em 5 segundos) ou na segunda (em 10 segundos). Importante notar que, mesmo que

um sensor-linfo não esteja sincronizado com o *Jammer*, ele será capaz de realizar duas verificações dentro do período de cada degrau do *Jammer*.

Nos experimentos, a identificação do intruso no primeiro intervalo indica um atraso de 5 segundos enquanto que a identificação no segundo intervalo indica 10 segundos de atraso.

Para um intervalo de envio de mensagens da aplicação igual a 1 segundo, apenas 3% dos experimentos gerou um atraso de 10 segundos (intruso identificado apenas na segunda verificação do MCAV). À medida que o intervalo de envio de mensagens da aplicação foi sendo aumentado (2, 3, 4 e 5 segundos) este percentual de atraso na identificação do intruso também aumentou, apresentando os valores de 43% para 2 segundos e 100% de 3 segundos em diante. Na Tabela 10, são ilustrados os resultados dos experimentos.

Analisando o comportamento apresentado, observa-se que, com o aumento do intervalo de envio de mensagens da aplicação, há uma redução da quantidade de mensagens trafegando na rede para ser avaliada, resultando em uma perda da capacidade do SDI de identificar um intruso na primeira verificação do MCAV.

Tabela 10. Atraso na identificação de um ataque.

Intervalo de envio das mensagens da aplicação (segundos)	**1**	**2**	**3**	**4**	**5**
Percentual de atraso (%)	3,3	43	100	100	100

5.3.6 Variação da taxa de envio de mensagens de aplicação pelos sensores

Nesta seção foi avaliado o efeito de se aumentar o intervalo de envio de mensagens da aplicação *BlinkToRadio* pelos sensores da rede a fim de comparar a energia consumida e os valores de FN e FP do SDI proposto neste trabalho. Utilizando-se as configurações estabelecidas na seção 5.3.3, foram realizados 5 experimentos para cada uma das configurações seguintes: (i) **A1-B2-C4**; (ii) **A1-B3-C4**; (iii) **A2-B2-C4**; e (iv) **A2-B3-C4**. Cada um dos experimentos teve o intervalo de envio de mensagens de aplicação fixada de 1 segundo até 5 segundos, de 1 em 1 segundo, respectivamente.

Pelos resultados obtidos e ilustrados nas Figuras 24 a 27 observa-se que, quanto mais mensagens de aplicação estiverem trafegando na rede, mais informações serão coletadas sobre o atacante. À medida que esta taxa diminui, diminuem as chances de identificar o atacante, conforme mostram as curvas de FN.

Do ponto de vista de segurança, a taxa de 1 segundo foi considerada como sendo o melhor intervalo de emissão de mensagens de aplicação experimentada pelos sensores, pois

resultou nos melhores valores de FN. Ao aumentar o intervalo de transmissão de mensagens de aplicação pelos sensores, ou seja, ao diminuir o número de mensagens da aplicação por segundos, as chances de identificar o ataque diminuem, pois além de ter menos mensagens trafegando na rede para análise do SDI, haverá uma maior demora para que o SDI emita uma resposta. Do ponto de vista de consumo de energia, ao aumentar o intervalo de envio de mensagens da aplicação, a energia consumida pela rede como um todo diminui, como era de se esperar, refletindo diretamente a diminuição da quantidade de mensagens trafegadas na rede. Este comportamento pode ser visto nas figuras 24 a 27.

Para as curvas de energia de 10 sensores-cd (A1), a partir de 5 segundos os valores de FN passam a ser maiores do que 99%, inviabilizando a utilização do SDI para este intervalo de emissão de mensagens. Já para as curvas de 5 sensores-cd (A2), este comportamento surge a partir de 4 segundos. Esta diferença ocorre devido à redução do número de sensores-cd, refletindo nas taxas de FN, as quais acabam por aumentar seus valores. Considerando que um FN indica a ausência de um ataque quando na verdade o mesmo está ocorrendo, o sensor-linfo não emite uma informação de contramedida para os sensores-cd (que consiste no desligamento do rádio), acarretando em um maior consumo de energia por parte dos sensores.

Os FP permaneceram zerados para todos os experimentos e não foram mostrados nos gráficos.

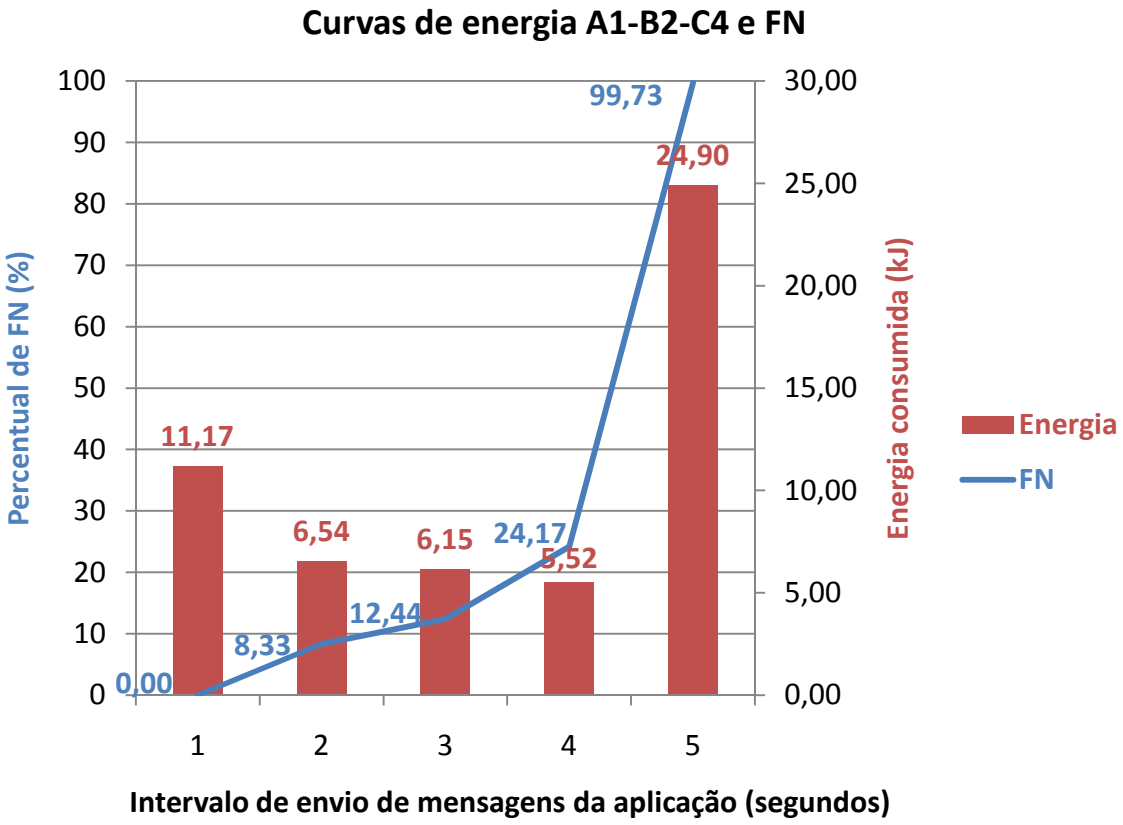


Figura 24. Curvas de energia A1-B2-C4 e FN de 1 a 5 segundos.

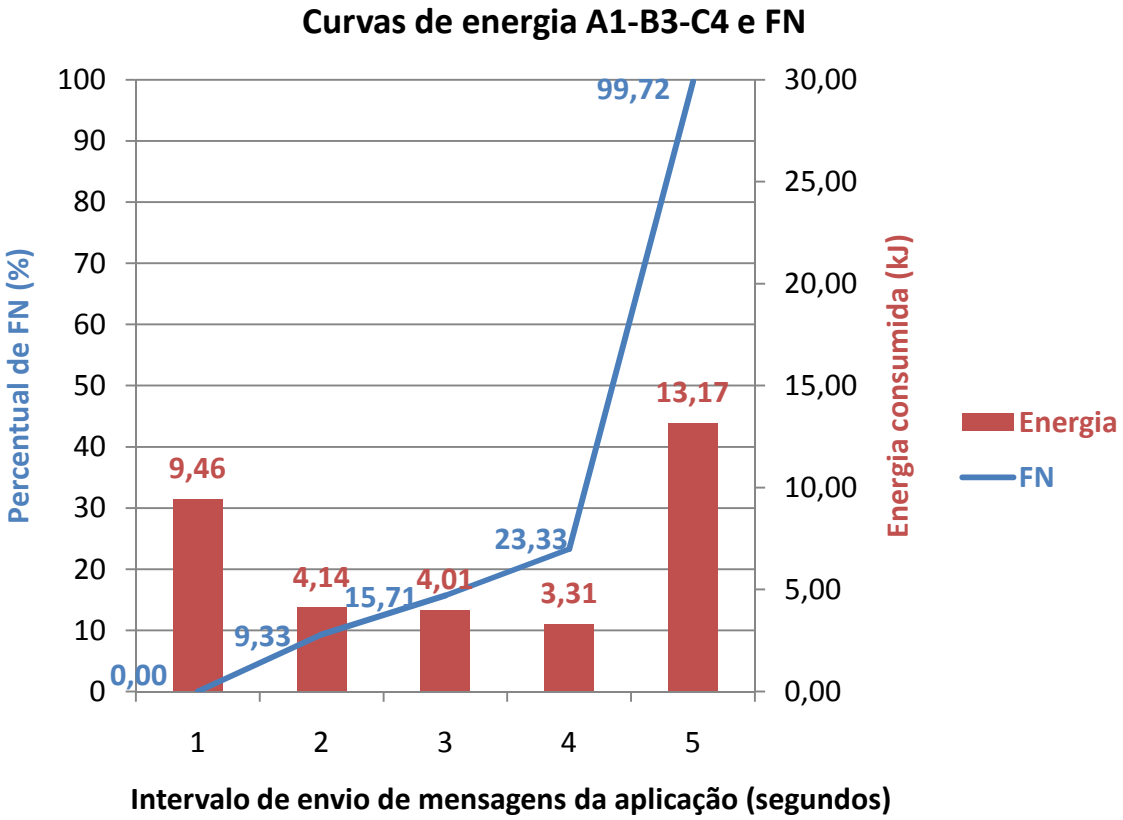


Figura 25. Curvas de energia A1-B3-C4 e FN de 1 a 5 segundos.

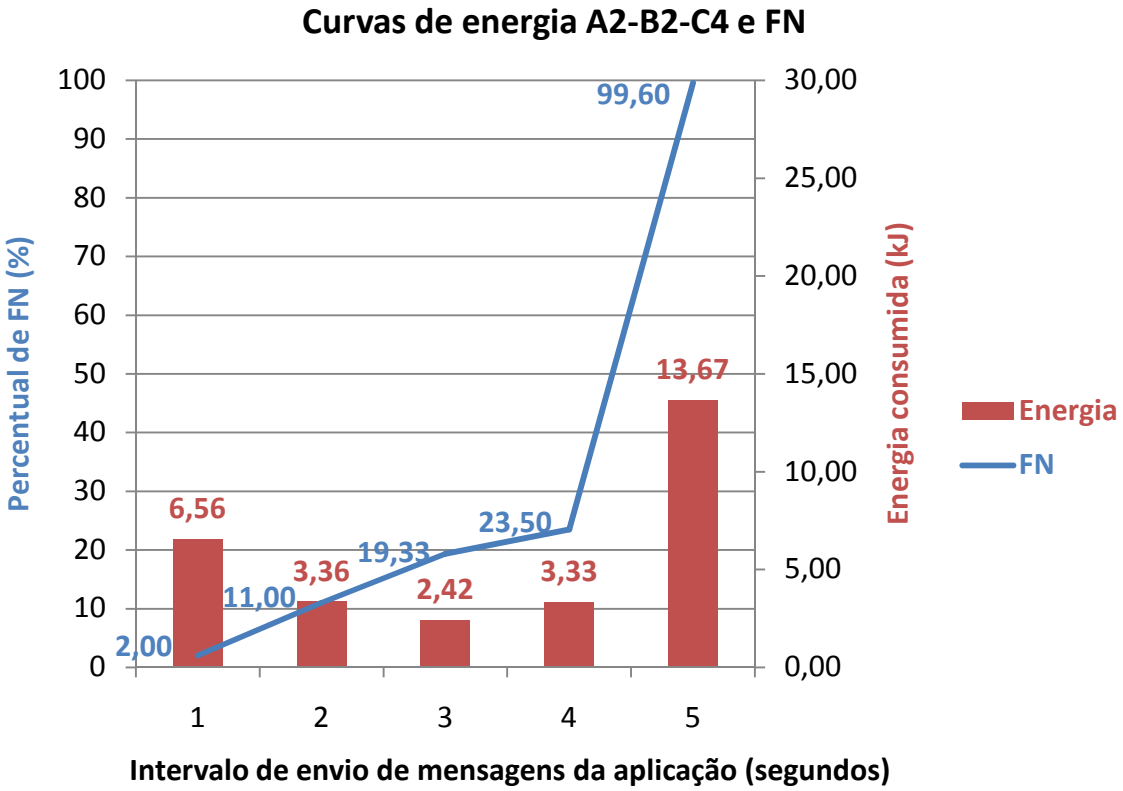


Figura 26. Curvas de energia A2-B2-C4 e FN de 1 a 5 segundos.

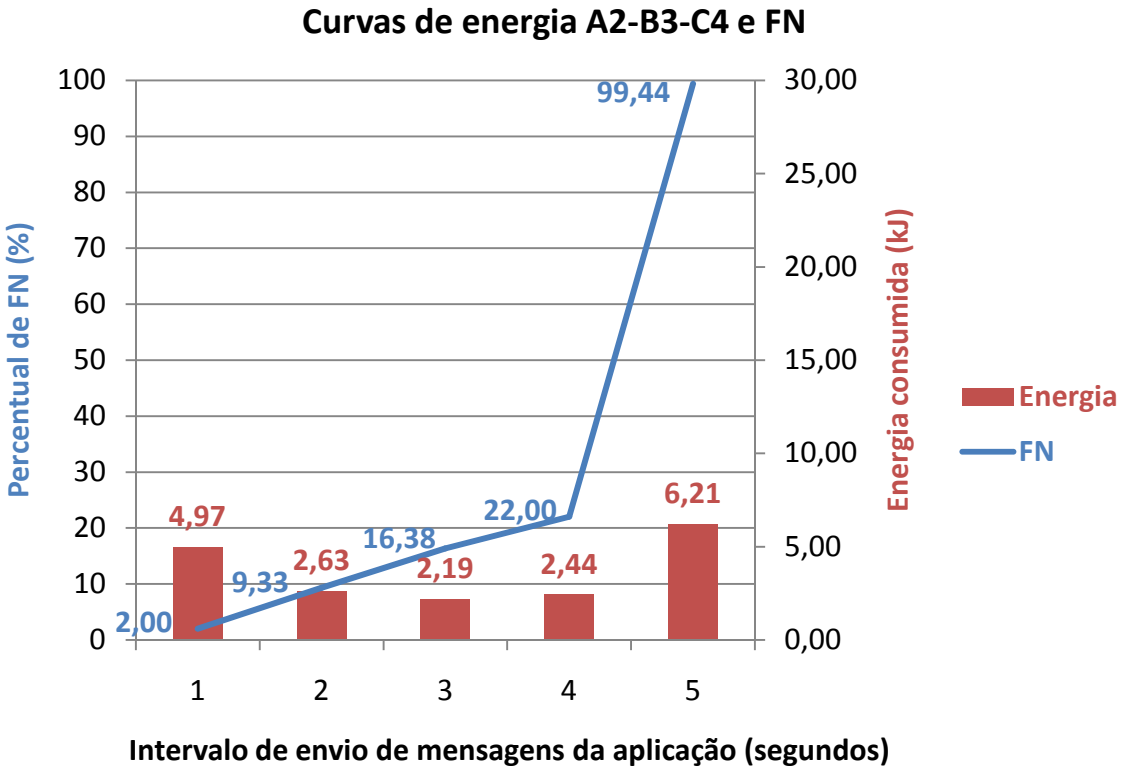


Figura 27. Curvas de energia A2-B3-C4 e FN de 1 a 5 segundos.

5.4 Comparação - Teoria do Perigo versus Teoria da Seleção Negativa

A fim de verificar a eficiência do SDI proposto, foi simulado um experimento onde se buscou comparar os índices de FP e FN encontrados por este trabalho e o trabalho apresentado por Liu e Yu (LIU e YU 2008). O trabalho de Liu e Yu (LIU e YU 2008) foi escolhido, pois utilizava outra técnica de SIH aplicada a RSSF, a Teoria da Seleção Negativa, que é uma concorrente direta da Teoria do Perigo.

Os experimentos realizados nessa seção adotaram o mesmo cenário apresentado por Liu e Yu (LIU e YU 2008) em termos de topologia (os nós foram mantidos estáticos durante toda a simulação), em termos de quantidade de nós presentes na RSSF, em termos de roteamento de pacotes na rede (os quais fluem para a EB por meio de um mecanismo de roteamento baseado em árvore) e em termos do funcionamento do nó atacante, o qual apresenta um funcionamento normal até que seja iniciado o ataque. O cenário apresentado por Liu e Yu (LIU e YU 2008) consistia em: 50 nós sensores distribuídos aleatoriamente por uma área de 100 x 100 metros quadrados (Figura 28). Cada um destes sensores possuía um alcance de 50 metros de raio. Liu e Yu (LIU e YU 2008) utilizaram o simulador TOSSIM para simular sensores modelo MICA2 que possuem um rádio modelo CC1000 (CROSSBOW 2010). No trabalho de Liu e Yu (LIU e YU 2008), todos os sensores possuem o mesmo módulo de SDI instalado e ativo durante todo o período da simulação. A área pintada de vermelho na Figura 28 indica a área de alcance do *Jammer*. Na Tabela 11 encontra-se um resumo contendo os dados da comparação.

Nos experimentos desta seção, também foi utilizado o simulador TOSSIM, porém, para sensores modelo MICAz, que possuem um rádio modelo CC2420 (CROSSBOW 2010). Apesar de Liu e Yu (LIU e YU 2008) não terem informado a aplicação utilizada em seu trabalho, a aplicação utilizada em todos os sensores da RSSF neste trabalho foi o *BlinkToRadio*. As posições dos sensores nos eixos das ordenadas e das abscissas na topologia utilizada para a comparação foram determinadas aleatoriamente. Este trabalho difere do trabalho de Liu e Yu (LIU e YU 2008) em termos de número de papéis dos sensores, pois, naquele trabalho, o SDI era o mesmo (um único papel) e fora instalado em todos os sensores. Neste experimento, foi proposta a utilização de apenas alguns sensores com os diferentes papéis do ACD (sensor-cd e sensor-linfo), não sendo necessária a instalação destes papéis em todos os sensores. Neste trabalho, assumiu-se que os sensores realizavam leituras periódicas a cada 1 segundo, conforme calibrado nas seções anteriores.

Foi utilizado para os experimentos um intervalo de verificação do MCAV igual a 5 segundos, um limiar de migração igual a 10 mensagens e as quantidades de sensores-cd variaram entre 5 e 10 sensores. Em alguns dos experimentos, parte dos sensores-cd permaneceu fora do raio de ação do *Jammer*, contribuindo de forma negativa para a identificação do ataque, pois emitiam apenas mensagens representando CDs semi-maduras. O nó 50 foi escolhido aleatoriamente para assumir o papel de *Jammer* sendo mantido com este papel para todos os experimentos. Importante observar que os nós com identificação 0, 1, 2, 3, 4, 10, 11, 13, 14, 19, 20, 31, 32, 37, 40, 41, 47, 48 e 49 ficaram fora do raio de atuação do *Jammer* devido à distância que se encontravam do mesmo. Foram realizados 4 experimentos, conforme descrito a seguir.

No experimento 1 foi escolhido o nó com identificação 16 para assumir o papel de sensor-linfo. Os nós com identificação 7, 12, 22, 28, 35, 4, 10, 18, 36 e 41foram escolhidos aleatoriamente para assumirem o papel de sensores-cd.

No experimento 2 foram escolhidos os nós com identificação 22 e 27 para assumirem o papel de sensor-linfo. Os nós com identificação 0, 7, 12, 15, 5, 28, 33, 35, 23 e 44 foram escolhidos aleatoriamente para assumirem o papel de sensores-cd e que emitiam para o nó 22. Os nós 1, 2, 10, 17, 20, 30, 41, 36, 47 e 45 assumiram o papel de sensores-cd que emitiam mensagens para o nó 27.

No experimento 3 foram escolhidos os nós 16, 22 e 27 para assumirem o papel de sensores-linfo. Os nós com identificação 0, 7, 8, 17, 23, 29, 35, 42, 44 e 45 foram escolhidos aleatoriamente para assumirem o papel de sensores-cd e que emitiam mensagens para o nó 16. Os nós 5, 6, 12, 15, 21, 28, 38, 33, 34 e 43 assumiram o papel de sensores-cd que emitiam mensagens para o nó 22. Os nós 1, 2, 3, 10, 11, 14, 31, 37, 41 e 49 assumiram o papel de sensores-cd que emitiam mensagens para o nó 27.

No experimento 4 foram escolhidos os nós 6, 10, 30 e 34 para assumirem o papel de sensores-linfo. Os nós com identificação 0, 5, 7, 8, 12, 15, 16, 22, 9 e 17 foram escolhidos aleatoriamente para assumirem o papel de sensores-cd e que emitiam mensagens para o nó 6. Os nós 1, 2, 3, 4, 11, 13, 14, 18, 19 e 20 foram escolhidos aleatoriamente para assumirem o papel de sensores-cd que emitiam mensagens para o nó 10. Os nós 25, 26, 27, 31, 32, 36, 37, 39, 40 e 41 foram escolhidos aleatoriamente para assumirem o papel de sensores-cd que emitiam mensagens para o nó 30. Os nós 28, 21, 33, 38, 35, 23, 42, 43, 44 e 29 foram escolhidos aleatoriamente para assumirem o papel de sensores-cd que emitiam mensagens para o nó 34.

Em Liu e Yu (LIU e YU 2008), os autores conseguiram um resultado de 100% de acerto na identificação de um ataque (VP) de *jamming*, porém, também mostraram 92,3% de erro na indicação de um ataque quando o mesmo não estava ocorrendo (FP). Na Tabela 12 são comparados os valores obtidos pelo SDI proposto neste trabalho com os resultados obtidos por Liu e Yu (LIU e YU 2008).

Tabela 11. Dados da comparação.

	Seleção Negativa (*)	**Teoria do Perigo**
Cenário	50 sensores	50 sensores
Modelo de sensor	MICA2	MICAz
Papéis do SDI	1	2 (CD e LN)
SDI	Todos os sensores	Alguns sensores
Processamento de dados	Centralizado	Distribuído
Interação humana	Sim	Não

Tabela 12. Resultados da comparação.

MCAV	**Métricas**	**Liu e Yu**	**Experimento 1**	**Experimento 2**	**Experimento 3**	**Experimento 4**
50	**VP**	100%	98,67%	98,79%	96,67%	93,67%
	FN	0%	1,33%	1,21%	3,33%	6,33%
	VN	7,7%	99,00%	97,67%	96,33%	96,33%
	FP	92,3%	1,00%	2,33%	3,67%	3,67%

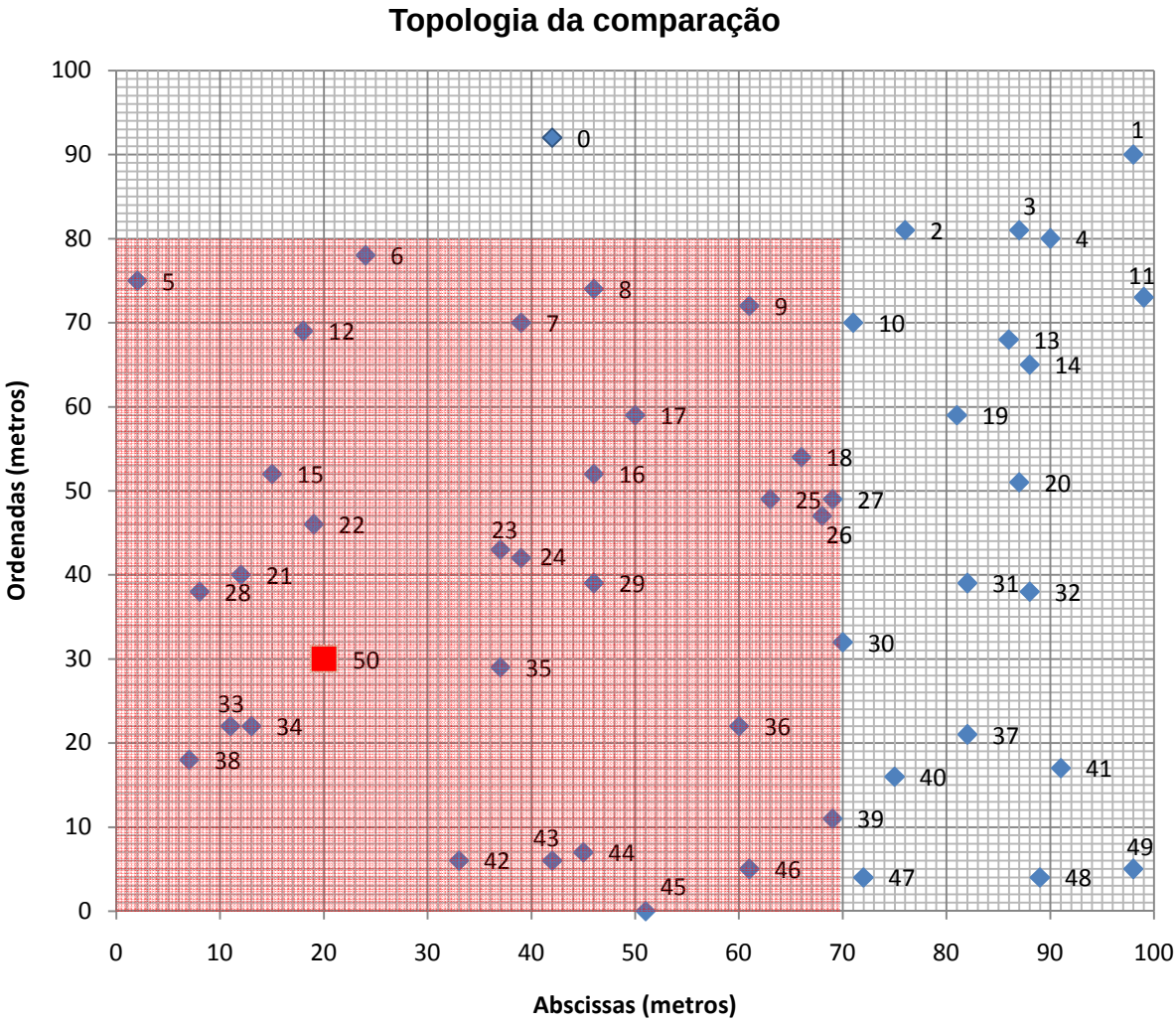


Figura 28. Topologia do cenário de comparação.

Observa-se pela Tabela 12 que os resultados obtidos em todos os experimentos do nosso trabalho foram menores do que os resultados obtidos por Liu e Yu (LIU e YU 2008) em termos de VP. Nas simulações realizadas, os valores de VP diminuem porque os sensores-linfo foram posicionados em locais distantes do *Jammer*.

Observa-se ainda na Tabela 12 que os resultados obtidos pelos experimentos do nosso trabalho foram melhores em termos de FP. Os valores apresentados para o primeiro, segundo, terceiro e quarto experimentos foram de 1,00%, 2,33%, 3,67% e 3,67%, respectivamente. Ou seja, valores muito abaixo do obtido por Liu e Yu (LIU e YU 2008), que foi de 92,3%.

Com relação à avaliação de energia, em Liu e Yu (LIU e YU 2008) não foi avaliado o impacto em termos de energia que o SDI proposto por eles incutiu à RSSF. Em nossa proposta, embora apresentando resultados menores de VP, espera-se que o tempo de vida da

rede seja intuitivamente maior uma vez que apenas alguns sensores possuem o SDI instalado e, conseqüentemente, consumindo menos energia.

5.5 Comparação - Simulado versus Plataforma real de sensores

Nesta seção foi avaliado um novo cenário de execução o qual foi implementado em plataformas reais de sensores. Este mesmo experimento foi simulado com o TOSSIM a fim de se comparar os resultados obtidos na plataforma real com os obtidos na simulação. No experimento real, os nós foram mantidos estáticos e dispostos no chão de um ambiente controlado (laboratório).

Foram utilizados 30 sensores, os quais foram posicionados em um *grid*, de coordenadas x e y medidas em metros. Neste *grid*, a EB ficou localizada nas coordenadas (0,1) e possuía uma identificação de nó (NodeId) igual a 0. Os sensores-linfo foram posicionados nas coordenadas (1,2), (2,0), (3,2), (4,0) e (5,2) e possuíam os NodeIds 1, 2, 3, 4 e 5, respectivamente. Os sensores-linfo das posições (2,0) e (4,0) ficaram responsáveis por receber informações de 4 sensores-cd. Os outros sensores-linfo ficaram responsáveis por receber informações de 5 sensores-cd. O *Jammer* foi posicionado nas coordenadas (6,1) e lhe foi atribuído o NodeId 99. Seu raio de atuação não foi alterado, sendo mantido um alcance de 15 metros, e foi mantido funcionando durante todo o experimento. Cada um dos sensores-cd teve seu rádio ajustado para um raio de alcance de 50 centímetros, limitando-o a manter comunicação apenas com o sensor-linfo mais próximo e entre os outros 4 sensores-cd que se comunicavam com o mesmo sensor-linfo. Isto foi feito para simular uma RSSF onde não existe a comunicação entre todos os sensores da rede. As contramedidas foram configuradas para desligar o rádio dos sensores por 20 segundos. A Figura 29 ilustra a topologia utilizada.

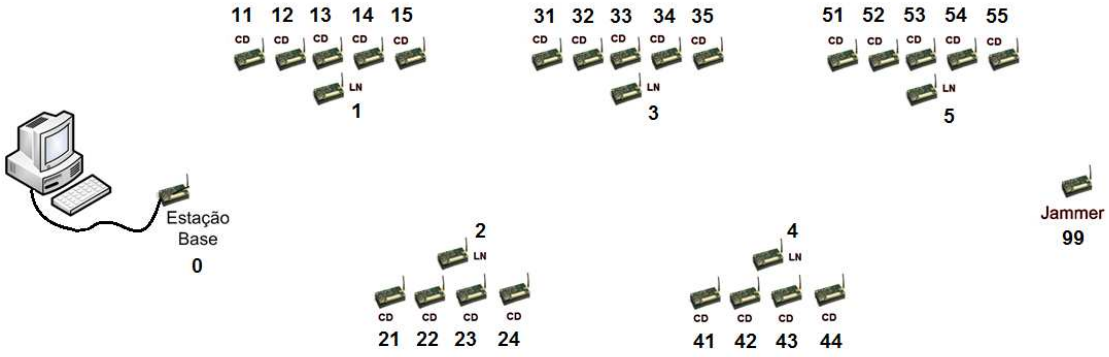


Figura 29. Disposição dos sensores no cenário real.

Os dois experimentos foram configurados da seguinte forma: limiar de migração igual a 10 mensagens, tempo de verificação de MCAV de 5 segundos e limiar de anomalia (MCAV) igual a 50%. Foram utilizados 3 sensores-linfo (NodeIds 1, 3 e 5) utilizando 5 sensores-cd e 2 sensores-linfo (NodeIds 2 e 4) utilizando 4 sensores-cd.

Os resultados dos experimentos simulados e implementados em sensores reais são mostrados na Tabela 13 e na Tabela 14, respectivamente. Os valores representam o percentual das métricas VP, FN, VN e FP.

Pela comparação da Tabela 13 e da Tabela 14 observa-se que tanto no experimento simulado quanto no real os sensores-linfo mais próximos do *Jammer* conseguiram identificá-lo, enquanto que os sensores-linfo mais distantes não.

Ao analisar as tabelas, percebe-se que os resultados de VP obtidos pelos sensores–linfo vão se tornando mais precisos à medida que estes estão localizados mais próximos do *Jammer*. O sensor-linfo 3 obteve resultados de VP para os experimentos simulados e real de 10,33% e 13,73%, respectivamente. O sensor 4 conseguiu identificar o *Jammer* com uma precisão maior do que o sensor 3, onde seus resultados de VP para os experimentos simulado e real foram 71,00% e 79,21%, respectivamente. O sensor 5, se comparado aos outros sensores conseguiu identificar de forma mais precisa o *Jammer*, obtendo 94,67% e 95,62% de VP para o experimento simulado e real, respectivamente. A proximidade do sensor 5 em relação ao *Jammer*, e, conseqüentemente, a identificação de um valor de RSSI maior, permitiu a identificação do *Jammer* de forma mais precisa.

Observou-se uma pequena diferença entre os valores obtidos no experimento simulado e no real. Atribuiu-se tais diferenças às aproximações realizadas pelo simulador TOSSIM, entre elas a escolha dos valores de atenuação entre os nós, que definem as distâncias entre os sensores. No simulador TOSSIM estes valores, uma vez definidos, não variam, enquanto que no ambiente real existem vários fatores que podem gerar interferência, afetando o resultado final.

Tabela 13. Resultados do cenário simulado.

MCAV	Métricas	Sensor-linfo 1	Sensor-linfo 2	Sensor-linfo 3	Sensor-linfo 4	Sensor-linfo 5
50	**VP**	0,00%	0,00%	10,33%	71,00%	94,67%
	FN	100,00%	100,00%	89,67%	29,00%	5,33%
	VN	100,00%	100,00%	100,00%	100,00%	100,00%
	FP	0,00%	0,00%	0,00%	0,00%	0,00%

Tabela 14. Resultados do cenário real.

MCAV	Métricas	Sensor-linfo 1	Sensor-linfo 2	Sensor-linfo 3	Sensor-linfo 4	Sensor-linfo 5
50	**VP**	0,00%	0,00%	13,73%	79,21%	95,62%
	FN	100,00%	100,00%	86,27%	20,79%	4,38%
	VN	100,00%	100,00%	99,21%	92,33%	97,33%
	FP	0,00%	0,00%	0,79%	7,67%	2,67%

6 Conclusões

Este trabalho apresentou uma proposta de arquitetura para Sistemas de Detecção de Intrusos (SDI) em RSSF baseada na Teoria do Perigo e no Algoritmo das Células Dendríticas (ACD), duas técnicas inspiradas no Sistema Imunológico Humano. Esta arquitetura seguiu as normas estabelecidas pelo *Common Intrusion Detection Framework* (CIDF), que, apesar de terem sido propostas para redes de computadores, mostraram-se aplicáveis também a RSSFs por meio da separação dos componentes estabelecidos na CIDF e customização de suas funcionalidades. A customização para RSSFs consistiu em se utilizar diferentes papéis, os quais assumiam as funcionalidades do SIH, tendo como objetivo principal incrementar os níveis de segurança da RSSF por meio da observação e utilização de parâmetros típicos/comuns nestas redes. Tais parâmetros foram utilizados para alimentar o ACD, o qual também foi customizado.

A utilização destas técnicas em RSSFs é facilitada pelas semelhanças encontradas nas características das RSSFs e do SIH. Assim como no SIH, as RSSFs são auto-organizadas, ou seja, vários sensores executando funções distintas trabalham em conjunto a fim de realizar uma tarefa comum. No SIH diversos órgãos e células cooperam para a eliminação de patógenos. O SIH é dito autônomo, pois não necessita de outro sistema para controlá-lo. Da mesma forma, as RSSFs uma vez implantadas e inicializadas no meio que será monitorado, passam a funcionar independentemente do comando de outro sistema. Outra característica abordada foi a robustez do SIH, onde a presença de diversos pontos de detecção permite a identificação de um patógeno suportando falhas eventuais em alguns pontos do sistema e, conseqüentemente, gerando redundância. Ou seja, todo o organismo sabe que foi invadido e que medidas devem ser tomadas para que ocorra a eliminação de um patógeno identificado por uma das partes do organismo. Nas RSSFs, a presença de diversos sensores (dezenas ou mesmo centenas) na rede permite que haja redundância de nós permitindo inclusive que sensores não tenham o SDI instalado para economizar energia. Finalmente, o SIH baseado na Teoria do Perigo e no ACD consegue identificar substâncias do próprio organismo, impedindo uma reação contra ele mesmo, gerando assim uma tolerância destas substâncias. Nas RSSFs também foi obtida tal característica, pois os sensores conseguiram diferenciar as mensagens próprias da rede daquelas produzidas pelo *Jammer*. A arquitetura proposta baseou-se nas características apresentadas.

Foram realizados diversos experimentos onde o SDI proposto foi calibrado e testado de forma a atender o interesse da aplicação em execução na rede, seja optando por uma

segurança maior em detrimento da economia de energia ou vice-versa. Por meio destes testes foi comprovada a eficiência do SDI para RSSFs. Além do consumo de energia pelos sensores, também foram analisados os recursos de memória consumidos pelos diferentes papéis assumidos pelos sensores. Nos experimentos realizados foram comparados os resultados do trabalho proposto com a técnica utilizada em outro trabalho, chamada de Teoria da Seleção Negativa. Tal comparação mostrou que, apesar de atingir valores menores de taxas de identificação de ataque durante sua ocorrência (verdadeiros positivos), as taxas de erro geradas pelo SDI durante uma condição normal do sistema foram muito menores (falsos positivos), mostrando a eficiência do algoritmo customizado proposto.

Outro experimento conduzido foi a comparação entre um experimento implementado em uma plataforma real de sensores e seu equivalente via simulador. Os resultados obtidos demonstram a proximidade dos resultados e a eficiência do SDI proposto.

Durante o aprendizado, foram experimentadas duas ferramentas para verificação de consumo de energia em sensores: o PowerTOSSIMz e o AvroraZ.

A ferramenta PowerTOSSIMz foi desenvolvida especificamente para os sensores MICAz (PERLA 2008), a fim de realizar a medição da energia consumida pelos sensores em uma RSSF, diferenciando tal consumo por processamento, *leds*, envio e recebimento de mensagens. O PowerTOSSIMz é considerado um "*plug in*" para o simulador TOSSIM mas, após uma bateria de testes em laboratório, onde foram testadas as aplicações disponíveis no repositório padrão do TinyOS, verificou-se que os resultados obtidos não eram consistentes. O mesmo problema vem sendo relatado nas listas de discussão da TinyOS.net, *site* oficial do TinyOS, e nenhuma solução foi proposta até o presente momento.

Já a ferramenta AvroraZ (ALBEROLA e PESCH 2008), que também foi desenvolvida para atender às especificações dos sensores MICAz, não permite a variação dos valores do RSSI. Esta ferramenta considera o mesmo valor para todos os envios e recebimentos de mensagens entre os sensores. A utilização de valores diferentes de RSSI é fundamental para a simulação do trabalho proposto, tornando o AvroraZ uma ferramenta impossível de ser utilizada nos experimentos realizados.

6.1 Trabalhos Futuros

Quanto aos trabalhos futuros, podem ser investigadas outras formas de se melhorar os resultados obtidos como, por exemplo, a utilização de pesos diferentes dos originais para os sinais de entrada da equação 1 do capítulo 2, de forma a melhorar os sinais de saída e,

conseqüentemente, os valores obtidos para VP, VN, FP e FN. Outra forma de se tentar uma melhora nestas métricas seria a verificação de outras maneiras de controlar a migração das células dendríticas para os linfonodos, de forma a se obter uma redução no consumo de energia imposto pela transmissão das mensagens entre os sensores-cd e os sensores-linfo. Por exemplo, em vez de se utilizar a contagem de mensagens recebidas como limiar de migração nos sensores-cd, poder-se-ia utilizar o tempo ou uma função que também levasse em conta o tempo, além da contagem de mensagens. Do ponto de vista de segurança, ao se utilizar a quantidade de mensagens, a solução fica muito dependente do tipo de ataque. Por outro lado, se fosse adotado o tempo como limiar de migração o resultado do ponto de vista de segurança poderia ser pior. Para avaliar esta resposta, novos experimentos precisariam ser realizados.

Já está sendo pesquisada por alunos do Laboratório de Redes do NCE a utilização de outros tipos de sinais de entrada para a identificação de outros ataques em RSSFs utilizando o SDI proposto neste trabalho. A escolha de sinais de entrada adequados é vital para uma detecção correta e rápida. Por meio da observação de diversos tipos de ataques, outra linha de pesquisa possível seria a implementação uma rede onde diversos sensores-linfo pudessem, simultaneamente, identificar mais de um tipo de ataque. Para isso, poderia ser elaborado um mecanismo de controle dinâmico para a escolha do ataque a ser monitorado pelos sensores.

Outra linha de pesquisa seria a implementação de um mecanismo que representasse as funcionalidades do Sistema Imunológico Adaptativo, ou seja, a capacidade de guardar em memória as características de ataques ocorridos de forma a identificá-los mais rapidamente, possibilitando a ativação de contramedidas em menor tempo. Poderia ser investigada uma atualização da arquitetura proposta de forma a fazer com que as Bases (Regras e Parâmetros) representassem esta memória.

Outro ponto a se observar é que tanto o comportamento da aplicação (taxa de mensagens transmitidas) quanto o comportamento do *Jammer* (duração dos degraus indicando se o *Jammer* está ativo ou inativo) foram mantidos constantes em todos os experimentos realizados. Para avaliar a resposta do SDI considerando a possibilidade de mudança no comportamento da aplicação ou do *Jammer*, novos experimentos precisam ser realizados.

Para tornar o SDI mais independente das variações de comportamento da aplicação e do *Jammer* duas questões podem ser consideradas. Uma delas seria permitir que o intervalo de verificação do MCAV variasse ao longo da execução do sistema, de forma a se adaptar às possíveis variações de comportamento. Outra seria fazer o limiar de migração ser baseado em intervalos de tempo, ao invés do número de mensagens recebidas. Ambas as questões

requerem uma reavaliação completa do SDI proposto. Estas possíveis alterações afetarão os resultados de atraso na identificação de um atacante, necessitando de novos experimentos que meçam os novos resultados a fim de compará-los com os já obtidos. A implementação destas alterações compreenderiam as seguintes alterações: no sensor-linfo, a fim de tornar o intervalo de verificação do MCAV variável, a criação de uma variável que armazenasse o tempo de duração médio do degrau de ativação de um *Jammer*, tornando esta verificação dinâmica; enquanto que nos sensores-cd, a fim de fazer com que o limiar de migração seja baseado no tempo, a inclusão de um *Timer* que controle o envio de mensagens ao invés do contador que é incrementado a cada mensagem recebida.

Neste trabalho, a avaliação da energia consumida foi feita para a rede como um todo. O consumo poderia ser verificado para cada um dos sensores, necessitando para isso que novas simulações fossem realizadas onde a contabilidade da energia consumida fosse feita individualmente por cada sensor.

Referências

AICKELIN, U e CAYZER, S. (2002) “The Danger Theory and Its Application to Artificial Immune Systems”, 1st International Conference on Artificial Immune Systems, Canterbury, páginas 141-148.

AICKELIN, U. (2003) “Artificial immune system and intrusion detection tutorial”, In Introductory Tutorials in Optimization, Search and decision support methodologies, Capítulo 13.

AICKELIN, U. *et al.* (2003) “Danger Theory: The Link between AIS and IDS?”, Lecture Notes in Computer Science, v. 2787, páginas 147-155.

AKYILDIZ, I. *et al.* (2002) “Wireless Sensor Networks: a Survey”, Computer Networks, v. 38, páginas 393-422.

ALBEROLA, R. e PESCH, D. (2008) “AvroraZ: extending Avrora with an IEEE 802.15.4 compliant radio chip model”, Proceedings of the 3rd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks, Vancouver, páginas 43-50.

BACHMAYER, S. (2008). “Artificial immune systems”, Department of Computer Science, University of Helsinki.

BAE SYSTEMS (2010) “WolfPack: Unattended Ground Sensors in the RF Domain”, http://www.baesystems.com/BAEProd/groups/public/documents/bae_publication/bae_pdf_eis_wolfpack.pdf. Acessado em Março de 2011.

BARBOSA, A. (2000) “Sistemas de Detecção de Intrusão – Seminários Ravel – CPS760”, http://www.lockabit.coppe.ufrj.br/downloads/academicos/IDS.pdf.

BROWNFIELD, M. *et al.* (2005) "Wireless sensor network denial of sleep attack", Proceedings from the Sixth Annual IEEE SMC Information Assurance Workshop, páginas 356-364.

CC2420 (2010) “CC2420 Data Sheet”, http://*inst.eecs.berkeley.edu/~cs150/Documents/CC2420.pdf* . Acessado em Março de 2011.

CHEN, R. *et al.* (2009) "A new method for intrusion detection on hierarchical wireless sensor networks", Proceedings of the 3rd International Conference on Ubiquitous Information Management and Communication, Suwon, Korea, páginas 238-245.

CROSSBOW (2010) “Crossbow Technology”, http://www.xbow.com/. Acessado em Março de 2011.

DA SILVA, A. (2005) “Detecção de Intrusos Descentralizada em Redes de Sensores Sem Fio”, Dissertação de Mestrado, UFMG.

DA SILVA, A. *et al.* (2006) “Detecção de Intrusos Decentralizada em Redes de Sensores Sem Fio”, 24º Simpósio Brasileiro de Redes de Computadores, Curitiba, Brasil.

DASGUPTA, D. (2006) “Advances in artificial immune systems”, Computational Intelligence Magazine, v. 1, edição 4, páginas 40-49.

DATEMA, S. (2005) “A Case Study of Wireless Sensor Network Attacks”, Master's Thesis in Computer Science, Parallel and Distributed Systems Group, ‘Faculty of Electrical Engineering, Mathematics, and Computer Science’, Delft University of Technology, Setembro.

DE CASTRO, L. N. (2001) “Engenharia Imunológica: Desenvolvimento e Aplicação de Ferramentas Computacionais Inspiradas em Sistemas Imunológicos Artificiais”. Tese de Doutorado, Departamento de Engenharia de Computação e Automação Industrial - Universidade Estadual de Campinas, Maio.

DEBAR, H. *et al.* (1999) “Towards a Taxonomy of Intrusion-Detection Systems”, Computer Networks, v. 31, páginas 805-822.

DELICATO, F. (2005) “Middleware Baseado em Serviços para Redes de Sensores sem Fio”, Tese de Doutorado, Universidade Federal do Rio de Janeiro, Julho.

DIETRICH, I. e DRESSLER, F. (2009) “On the Lifetime of Wireless Sensor Networks”, ACM Transactions on Sensor Networks, volume 5, edição 1, páginas 1-39.

DROZDA, M. *et al.* (2007) “AIS for Misbehavior Detection in Wireless Sensor Networks: Performance and Design Principles”, IEEE Congress on Evolutionary Computation, páginas 3719-3726, Singapore.

GARCÍA-TEODORO, P. *et al.* (2008) “Anomaly-based network intrusion detection: Techniques, systems and challenges”, Computers & Security, v. 28, páginas 18-28.

GREENSMITH, J. (2007) “The Dendritic Cell Algorithm”, PhD Thesis, University of Nottingham.

GREENSMITH, J. *et al.* (2005) “Introducing Dendritic Cells as a Novel Immune-Inspired Algorithm for Anomaly Detection”, 4th International Conference on Artificial Immune Systems, Canada.

GREENSMITH, J. *et al.* (2006) “Articulation and Clarification of the Dendritic Cell Algorithm”, Lecture Notes in Computer Science, páginas 404-417.

GREENSMITH, J. *et al.* (2007) “Dendritic Cells for SYN Scan Detection”, Proceedings of the 9th annual conference on Genetic and evolutionary computation, England, páginas 49-56.

GREENSMITH, J. *et al.* (2008) “Detecting Danger: The Dendritic Cell Algorithm”, HP Laboratories, páginas 89-112.

HART, E. e TIMMIS, J. (2008) “Application areas of AIS: The past, the present and the future”, Applied Soft Computing 8, páginas 191–201, Elsevier.

HOFMEYR, S e FORREST, S. (2000) “Architecture for an Artificial Immune System”, Evolutionary Computation,, volume 8, número 4, páginas 443-473.

HONG, L. e YANG, J. (2009) “Danger Theory of Immune Systems and Intrusion Detection Systems”, International Conference on Industrial Mechatronics and Automation, Chengdu, páginas 208 – 211.

HUSSAIN, A. *et al.* (2009) “WSN Research Activities for Military Application”, International Conference on Advanced Communication Technology, páginas 271-274.

IOANNIS, K. *et al.* (2007) “Towards Intrusion Detection in Wireless Sensor Networks”, 13th European Wireless Conference.

KARLOF, C. e WAGNER, D. (2003) "Secure routing in wireless sensor networks: attacks and countermeasures", Proceedings of the First IEEE Sensor Network Protocols and Applications, Berkeley, páginas 113-127.

KAUR, K. e SINGH, B. (2010) “Wireless Sensor Network based: Design Principles & measuring performance of IDS”, International Journal of COmputer Applications, volume 1, número 28, páginas 81-85.

KIM, J. *et al.* (2006) “Danger Is Ubiquitous: Detecting Malicious Activities in Sensor Networks Using the Dendritic Cell Algorithm”, Artificial Immune Systems, Springer Berlin, v. 4163/2006, páginas 390-403.

KIM, J. *et al.* (2007) “Immune system approaches to intrusion detection - a review”, Natural Computing, páginas 413-466.

LEVIS, P. e GAY, D. (2009) “TinyOS Programming”, Cambridge University Press, Cambridge.

LIU, Y. e YU, F. (2008) “Immunity-Based Intrusion Detection for Wireless Sensor Networks”, Neural Networks, IJCNN, páginas 439-444.

LOUREIRO, A. *et al.* (2002) “Rede de sensores sem fio”, Capítulo 5 do Livro texto da XXI Jornada de Atualização em Informática do XXII Congresso da Sociedade Brasileira de Computação, páginas 193–234.

MARGI, C. *et al.* (2009) “Segurança em Redes de Sensores Sem Fio”, SBSeg 2009, Minicurso: Segurança em Redes de Sensores Sem Fio, páginas 149-194.

MARTYNOV, D. *et al.* (2007) “Design and Implementation of an Intrusion Detection System for Wireless Sensor Networks”, IEEE International Conference on Electro/Information Technology, páginas 507–512, Chicago.

MATZINGER, P. (1994) “Tolerance, danger, and the extended family”, Annual Review of Immunology, v. 12, páginas 991-1045.

MATZINGER, P. (2002) “The Danger Model: A Renewed Sense of Self”, Science, v. 296, número 5566, páginas 301-305.

MAZHAR, N. e FAROOQ, M. (2008) “A sense of danger: dendritic cells inspired artificial immune system for manet security”, Proceedings of the 10th Annual Conference on Genetic and Evolutionary Computation, Atlanta, páginas 63-70.

ONAT, I. e MIRI, A. (2005) “An intrusion detection system for wireless sensor networks,” in Proceeding of the IEEE International Conference on Wireless and Mobile Computing, Networking and Communications, v. 3, Montreal, Canada, páginas 253–259.

PERLA, E. *et al.* (2008) “PowerTOSSIM z: Realistic Energy Modelling for Wireless Sensor Network Environments”, Proceedings of the 3rd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks, Vancouver, páginas 35-42.

PERRIG, A. *et al.* (2004) “Security in Wireless Sensor Networks”, Communications of the ACM, volume 47, número 6, páginas 53-58.

RAYMOND, D. *et al.* (2009) “Effects of Denial of Sleep Attacks on Wireless Sensor Network MAC Protocols”, IEEE Transactions on Vehicular Technology, v. 58, páginas 367-380.

ROMAN, R. *et al.* (2005) “On the Security of Wireless Sensor Networks”, Computational Science and Its Applications – ICCSA 2005, páginas 681-690.

ROMAN, R. *et al.* (2006) “Applying Intrusion Detection Systems to Wireless Sensor Networks”, 3rd Consumer Communications and Networking Conference, v. 1, páginas 640-644.

SALMON, H. *et al.* (2010) “Sistema de Detecção de Intrusão Imuno-inspirado customizado para Redes de Sensores Sem Fio”, X Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais, Fortaleza.

SHI, E. *et al.* (2004) “Designing Secure Sensor Networks”, IEEE Wireless Communications, Wireless Sensor Networks, páginas 38-43.

SILVA, G. (2009) “Detecção de Intrusão em Redes de Computadores: Algoritmo Imunoinspirado Baseado na Teoria do Perigo e Células Dendríticas”, Tese de Mestrado, Programa de Pós-Graduação da Engenharia Elétrica, Universidade Federal de Minas Gerais, Março de 2009.

SIMON, G. et al. (2004) “Sensor Network-Based Countersniper System”, ACM Conference on Embedded Networked Sensor Systems, páginas 1-12.

SU, X. *et al.* (2007) “Secure routing in ad hoc and sensor networks”, Wireless Network Security, Springer US, Part V, páginas 381-402.

TIMOFTE, J. (2008) “Wireless Intrusion Prevention Systems”, Revista Informática Economica, número 3, páginas 129-132.

TIWARI, M. *et al.* (2009) “Designing Intrusion Detection to Detect Black hole and Selective Forwarding Attack in WSN based on local Information”, Fourth International Conference on Computer Sciences and Convergence Information Technology, Seoul, páginas 824-828.

VIANNA, N. (2006) “EWIDS: Uma Extensão para Arquiteturas de Sistemas de Detecção de Intrusos para Redes Sem Fio Metropolitanas”, Dissertação de Mestrado, PPGI, UFRJ.

WALLENTA, C. *et al.* (2010) “Detecting interest cache poisoning in sensor networks using an artificial immune algorithm”, Applied Intelligence, volume 32, número 1, páginas 1-26.

WANG, Y. e TSENG, Y. (2006) “Attacks and Defenses of Routing Mechanisms in Ad Hoc and Sensor Networks”, Security in Sensor Networks, Capítulo 1, Auerbach Publications.

WINKLER, M. *et al.* (2008) “Theoretical and practical aspects of military wireless sensor networks”, Journal of Telecomunications and Information Technology, páginas 37-45.

XU, W. *et al.* (2005) "The feasibility of launching and detecting jamming attacks in wireless networks", 6th ACM international symposium on Mobile ad hoc networking and computing, Urbana-Champaign, páginas 46-57.

XUE, G. e HASSANEIN, H. (2006) “On current areas of interest in wireless sensor networks designs”, Computer Communications, número 29, páginas 409-412.

YICK, J. *et al.* (2008) “Wireless sensor network survey”, Computer Networks, número 52, páginas 2292-2330.

ZAMANI, M. *et al.* (2009) "A DDoS-Aware IDS Model Based on Danger Theory and Mobile Agents", Proceedings of the 2009 International Conference on Computational Intelligence and Security, volume 1, páginas 516-520.