Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia
Departamento de Informática

Preparação da Dissertação

Mestrado em Engenharia Informática

Secure and reliable routing for dependable wireless sensor networks

Pedro Miguel Oliveira Marques da Silva (nº 26649)

1º Semestre de 2009/10

5 de Fevereiro de 2010

Universidade Nova de Lisboa
Faculdade de Ciências e Tecnologia
Departamento de Informática

Preparação da Dissertação

Secure and reliable routing for dependable wireless sensor networks

Pedro Miguel Oliveira Marques da Silva (nº 26649)

Orientador: Prof. Doutor Henrique João Lopes Domingos

Trabalho apresentado no âmbito do Mestrado em Engenharia Informática, como requisito parcial para obtenção do grau de Mestre em Engenharia Informática.

1º Semestre de 2009/10

5 de Fevereiro de 2010

Resumo

As redes de sensores são uma tecnologia emergente no domínio da monitorização, de forma autónoma, de ambientes físicos. São formadas por pequenos dispositivos que se auto-organizam de modo a cobrirem uma área geográfica, podendo formar uma rede de larga escala com milhares de nós. Esta autonomia e auto-organização apresentam alguns desafios relacionados com os aspectos de segurança, nomeadamente no que concerne ao encaminhamento de dados.

O trabalho a realizar pretende contribuir para a criação de um modelo sistémico para o estudo de protocolos de encaminhamento seguro em redes de sensores sem fios (RSSF). A definição do modelo de adversário é o passo inicial para o enquadramento das tipologias de ataque que se pretendem avaliar. Aliado ao modelo formal de Dolev-Yao, orientado para os ataques ao meio de comunicação, o estudo de novos modelos de adversário, relacionados com a intrusão e captura de nós, é pertinente e apresentado dentro do âmbito deste trabalho. Com vista a tornar as RSSF resistentes a algumas tipologias de ataques preconizadas no modelo de adversário, têm vindo a ser desenvolvidos diversos algoritmos de encaminhamento seguro. Pretende-se estudar alguns destes algoritmos, representativos do estado da arte neste domínio, estabelecendo uma matriz de medidas de resistência ao modelo de adversário, que permita então avaliar a efectividade destes .

Como contributo principal deste trabalho destaca-se a concepção de um ambiente de simulação inovador, uma vez que se pretendem implementar funcionalidades não encontradas nos sistemas de simulação para as RSSF existentes. Este sistema proporcionará a possibilidade de desenhar e avaliar algoritmos de encaminhamento, concebidos para serem seguros, quando sujeitos a ataques definidos no modelo de adversário. Esta avaliação estará centrada fundamentalmente na análise de propriedades como o consumo de energia, fiabilidade, latência, correcção dos dados e correcção do comportamento do protocolo.

Palavras-chave: Redes de sensores sem fios, protocolos de encaminhamento seguros, simulação de redes de sensores, ataque por intrusão

Abstract

Sensor networks are an emerging technology in the field of monitoring physical environments, in an autonomous manner. They are formed by small, self-organized devices which cover a geographical area and can form a large scale network with thousands of nodes. This autonomy and self-organization present some security challenges, concerning data routing, in particular.

This work aims to contribute to the creation of a systemic model for the study of secure routing protocols in wireless sensor networks (WSN). The definition of the opponent model is the first step to an enhanced understanding of the different types of attack. Coupled with the formal Dolev-Yao model, which focuses on the attacks on communication media, the study of new adversary models, related to intrusion and capture of nodes, is relevant and presented within this work In order to make the WSN resistant to some types of attacks, several secure routing algorithms have been developed. The aim is to study some of these algorithms, representatives of the state of the art in this field, establishing a matrix of resistance to each type of opponent, which then allows the evaluation of their effectiveness.

The major contribution of this study is the design of an innovative simulation environment, since some features to implement are not found in existing WSN simulation systems. It will provide the opportunity to design and evaluate routing algorithms, designed to be secure, when subjected to the attacks defined in the opponent model. This evaluation will primarily focuses on the analysis of certain properties, such as energy consumption, reliability, latency, data accuracy and correction of the protocol's behavior.

Keywords: Wireless Sensor Networks, Secure Routing Protocols, WSN Simulation, Intrusion Atack

Conteúdo

Lista de Figuras

Lista de Tabelas

1. Introdução

1.1 Introdução geral

Recentemente, têm-se observado avanços na concepção e fabrico de sistemas computacionais programáveis, baseados em *hardware* de pequena dimensão, [67] com capacidade para desempenhar tarefas específicas. Estes avanços permitiram integrar, nesses sistemas, processadores miniaturizados, memória, dispositivos de processamento de sinal e de conversão analógica-digital para a detecção de diferentes fenómenos físicos (através de diversos tipos de sensores) e capacidades de comunicação sem fios por rádio frequência (com base nas normas 802.15.4 [16, 21] e Zigbee [21])). A possibilidade de construção destes dispositivos (que se designam mais simplesmente por nós sensores) fez surgir, nos últimos anos, um novo campo da investigação conhecido por redes de sensores sem fios (RSSF).

Uma RSSF é formada por um conjunto de dispositivos com as características descritas anteriormente, distribuídos numa certa área geográfica, que podem funcionar de forma autónoma ou sem supervisão humana. Estas redes permitem monitorizar, com maior ou menor densidade, diferentes fenómenos físicos associados ao meio ambiente envolvente. As RSSF possuem características de auto-organização, podendo ser formadas por um menor ou maior número de sensores, permitindo cobrir desde pequenas a vastas áreas de monitorização. Um ambiente de instalação de uma rede pode ser um edifício, uma instalação industrial, uma área de combate, uma vasta zona de monitorização de um habitat natural, um veículo ou o próprio corpo humano [65, 48, 49, 18].

A componente básica e fundamental de uma rede de sensores é, pois, o nó sensor (também designado por *mote*) [8, 1, 13, 9]. Cada um destes nós constitui um substrato computorizado que pode possuir diversos sensores para monitorizar, por exemplo, temperatura, luz, movimento e outros fenómenos físicos, consoante as necessidades das aplicações. Sendo um dispositivo miniaturizado e concebido para possuir um baixo custo de produção, apresenta um poder de computação limitado, baixa largura de banda de comunicação, curto alcance de comunicação rádio e energia autónoma limitada (com base em baterias do tipo AAA ou de células fotovoltaicas) [71]. Numa RSSF a energia pode constituir um recurso finito. Em certos ambientes de instalação pode não ser possível, ou viável, realizar operações que exijam intervenção ou supervisão humana, por exemplo, para o carregamento ou substituição de baterias. Em certos

ambientes de instalação pode não ser possível, ou viável, realizar operações que exijam intervenção ou supervisão humana, por exemplo, para o carregamento ou substituição de baterias. Conhecidas estas limitações, para se poderem atingir distribuições de grande cobertura geográfica, os sensores têm de ser distribuídos em grande número, podendo-se também, por esse meio, aumentar a redundância dos nós que, assim, formam redes de larga escala que chegam a atingir milhares de nós.

Os *motes* podem diferir uns dos outros consoante a sua função na rede e poderão desempenhar, fundamentalmente, dois papeis: nó genérico gerador de informação (*source-nodes*) e nó base ou de sincronização (nó colector de dados da rede ou de execução de comandos de pesquisa). Numa RSSF os nós podem também actuar como nós de interligação (ou de encaminhamento e processamento intermédio através da rede) ou *gateways* (que permitem ligar o ambiente da rede de sensores a outras redes e sub-sistemas externos). Uma RSSF pode ser concebida de forma a ser interligada a outras infraestruturas computacionais que, com maior capacidade de armazenamento e processamento, permitem efectuar análise de dados coligidos. Na tecnologia actual existem ainda nós de desenvolvimento, que possuem ligações a computadores (ex: ligação de rede *Ethernet*, RS-232 ou USB), permitindo o carregamento expedito de código desenvolvido em estações de desenvolvimento. Os sensores dotados de ligações Ethernet, RS-232 ou USB podem ainda funcionar como nós de tipo *gateway*, permitindo, num cenário concreto, ligar a rede de sensores a aplicações, executando em sistemas de computadores usuais [3].

1.1.1 Caracterização de RSSF

As redes de sensores sem fios podem ser abordadas como um caso especial de redes *ad-hoc*, embora exibam características específicas [24]. As RSSF para aplicações de larga escala fazem emergir algumas problemáticas inerentes a aplicações distribuídas, com especificidades e desafios próprios [25], nomeadamente ao nível dos mecanismos de gestão, da organização topológica autónoma, das necessidades de sistemas de encaminhamento *multi-hop*, de requisitos de tolerância a falhas, de requisitos de escalabilidade ou de necessidade de serviços de segurança.

1.1.2 Aplicações

Muitas foram as aplicações [18, 43] encontradas na investigação ou na utilização emergente de RSSF com diferentes requisitos de escala [71]. O carácter autónomo destas redes oferece um sem número de vantagens que propicia a sua utilização em locais remotos de acessibilidade difícil e onde não é possível a sua manutenção e supervisão. De entre as aplicações das RSSF, podem destacar-se as seguintes:

Detecção de alvos/objectos(*Target Tracking*): [65] associada à detecção de movimento (trajectória/presença) em áreas vigiadas (como, por exemplo, em teatros operacionais militares ou na vigilância e monitorização de recursos ou infraestruturas);

Monitorização de fenómenos naturais: [48] associada à detecção de eventos ou anomalias ambientais (com aplicações na agricultura, monitorização de poluição ou monitorização de habitats naturais), bem como de vigilância ou controlo de fenómenos naturais (sismos, actividade de vulcões, etc);

Recolha de dados: [49] associada ao controlo de indicadores físicos ou biomédicos de pessoas ou de animais (com recurso a sensores especiais, associados a aplicações da medicina) ou como ambientes de monitorização de operação de infraestruturas críticas (pontes, edifícios, sistemas electromecânicos ou equipamentos de instalações fabris).

1.2 Segurança em RSSF

A segurança nas RSSF é, *de facto*, um problema, quando se perspectiva a sua utilização para sistemas críticos. A segurança deve ser pensada em tempo de concepção [57], tendo em vista a abrangência do sistema e tendo em atenção as particularidades específicas da tecnologia inerente, bem como dos ambientes onde são implementadas. Importa analisar as hipóteses de desencadeamento de ataques a estas redes (a partir da definição de modelos de adversário) bem como as repercussões das potenciais tipologias de ataques que representem o modelo de adversário [73, 41]. Esta análise deve ser feita tendo em conta a pilha de protocolos [18] e de serviços associados ao *software* [14, 2, 42, 59] que executa em cada nó, uma vez que cada uma das camadas de serviços e protocolos pode ser vulnerável a esses ataques.

Na abordagem de uma plataforma usual e genérica para um nó de uma RSSF, verifica-se

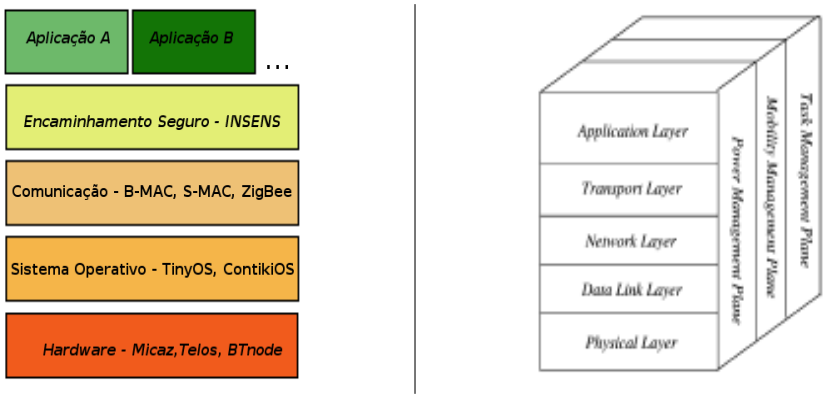

Figura 1.1 Pilha de serviços e pilha de protocolos [18] de uma RSSF

que, em geral, cada nó apresenta uma pilha minimalista de protocolos e serviços, por comparação com uma pilha associada a uma rede de computadores usual (ex., TCP/IP ou pilha OSI) [64]. As limitações impostas pelas dimensões e as capacidades de operação não permitem uma arquitectura muito ambiciosa e, por outro lado, as RSSF possuem geralmente uma vocação orientada para aplicações específicas, que condicionam os serviços que devem ser suportados na pilha. As camadas de operação de um nó sensor são essencialmente cinco [18]: camada física, camada de ligação de dados, camada de rede, camada de transporte e camada de aplicação. Todavia, na maior parte dos casos, a camada de transporte de dados e a funcionalidade inerente à camada de rede são concebidas de forma mais ou menos específica, tendo em vista as características particulares das aplicações. Na investigação, verifica-se ainda que a camada de ligação de dados (nível MAC e protocolos data-link) foram objecto de várias propostas, com diferentes variantes que podem apresentar vantagens particulares, dados os requisitos de operação das aplicações [60, 70, 26].

Alguns autores [68, 27, 55] têm vindo a desenhar algoritmos com vista a minimizar o impacte dos ataques ao encaminhamento, durante a operação das RSSF. Estes algoritmos pretendem garantir algumas propriedades básicas de segurança [62] (ex: confidencialidade, integridade, autenticação, detecção de retransmissão ilícita de dados). Não obstante, devem ser consideradas outras tipologias de ataques especificamente associados ao suporte de encaminhamento de dados na rede. Diferentes protocolos de encaminhamento seguro em RSSF endereçam apenas algumas dessas tipologias mas, em geral, não contemplam contra-medidas globais face a todas. Por outro lado, muitas das propostas apresentadas têm por base modelos matemáticos, análises teóricas ou experiências de pequena dimensão. Torna-se importante que esse estudo

teórico seja complementado e verificado, tendo em conta o desempenho desses protocolos com análises experimentais mais próximas dos ambientes reais em que a rede opera. Para tal, o recurso a ambientes de simulação torna-se uma direcção importante de apoio a este estudo.

1.3 Objectivos e contribuições previstas para a dissertação

Uma das vertentes do estudo da segurança em RSSF tem que ver com a possibilidade de se poderem efectivar ataques ao nível do encaminhamento de dados (da pilha de suporte de serviços de *software*, Figura 1.1). Diferentes tipologias de ataques [41, 36, 57] exigem diferentes tipos de contra-medidas. Estas, normalmente, são combinadas em mecanismos de segurança, consoante as propostas de sistemas de encaminhamento seguro. Estes são implementados tendo em conta as características e o modo de operação das RSSF [59, 45, 42].

Conhecidas que estão as dificuldades existentes no estudo de protocolos de encaminhamento seguro [41, 50], estes permanecem como um dos aspectos em aberto e como um desafio à concepção de RSSF que operem em condições de segurança. Este desafio é tanto mais relevante quanto a análise de segurança pode envolver a avaliação de diferentes modelos e hipóteses de adversário [28, 54] e tipologias de ataques [73, 41]. Estas tipologias nem sempre são estudadas de forma sistemática e comparativa, na abordagem de diferentes protocolos de encaminhamento seguro que vão sendo propostos. Por outro lado, existe uma dificuldade adicional em poder conjugar-se o estudo das contra-medidas de segurança importantes com a sua avaliação experimental face à implementação. Para cada protocolo interessa medir o impacte que diferentes tipologias de ataques podem ter, nomeadamente, em cenários de grande escala, o que exige a utilização de sistemas de simulação [10, 5, 11, 12, 6, 4] de RSSF que permitam simular diferentes hipóteses de ataque e antecipar o seu impacte. Este impacte deve ser analisado não apenas no que refere ao comportamento dos protocolos mas, complementarmente, no que refere à forma como afectam a própria rede nomeadamente, em termos de consumo de energia, fiabilidade e latência. Um sistema que suprima estas dificuldades contribui para um mais rápido desenvolvimento e uma afinação mais cuidada de determinados parâmetros dos protocolos, com vista a garantir as propriedades de segurança desejadas para o ambiente de operação das RSSF, em condições mais realistas.

No âmbito do trabalho que se pretende desenvolver na elaboração da dissertação, irá ser concebido e desenvolvido um sistema de simulação inovador. Este sistema deverá permitir o

estudo sistemático de protocolos de encaminhamento, desenhados para serem seguros, e deverá possuir, em particular, as seguintes funcionalidades:

- Interface de visualização e configuração da rede com informações dos parâmetros de simulação e informação de cada nó (por exemplo, o seu estado energético);
- Implementação de um modelo de energia que permita extrair consumos em diferentes momentos de operação: operação normal e operação perante determinado ataque;
- Modelo parametrizável de geração de topologias, podendo estas ser definidas como: distribuição aleatória, distribuição em grelha, distribuição controlada (estruturada);
- Mecanismo de introdução de falhas/ataques na rede. Com este mecanismo pretende-se permitir o estudo do comportamento dos protocolos face à possibilidade de introduzir ataques tipificados;
- Utilitários de recolha de dados da simulação, em tempo real e em tempo diferido, que permitam a extracção de medições referentes a propriedades importantes, como consumos de energia, latência, fiabilidade, correcção do protocolo e correcção dos eventos, disponibilizando-os de forma gráfica.

O ambiente de simulação em causa permitirá estudar, sistemática e comparativamente, diferentes protocolos de encaminhamento seguro em RSSF. Contribuirá, ainda, com o estabelecimento de uma base de comparação, pelo desenvolvimento de dois protocolos, para futuras avaliações de outros protocolos emergentes. Esta avaliação será referente ao impacte que os mecanismos de segurança têm sobre propriedades tão importantes como: o consumo de energia, a fiabilidade da rede, a latência das comunicações, correcção dos dados e a correcção do protocolo. Estas propriedades são, normalmente,extrapoladas da implementação experimental em pequena escala ou da adopção de modelos matemáticos, os quais, devido a variáveis externas, próprias do ambiente de operação, se podem afastar bastante dos resultados reais.

2 . Trabalho relacionado

Este capítulo apresenta uma visão do estado da arte relacionado com a segurança e modelos de simulação em RSSF. A primeira secção apresenta a definição do modelo de adversário e tipologias de ataques. A segunda secção apresenta protocolos de encaminhamento seguro. A terceira secção apresenta diversos ambientes de simulação relacionados com as RSSF e *ad-hoc*. Por fim, apresenta-se uma discussão e análise crítica do trabalho relacionado, com vista a enquadra-lo com os objectivos da dissertação a elaborar.

2.1 Modelo de Adversário, Ataques ao Encaminhamento e Contra-medidas

2.1.1 Arquitectura de Serviços de Segurança em RSSF

Num sistema seguro é necessário que a segurança esteja integrada em cada um dos seus componentes, não se confinando a um componente isolado do sistema [58]. Nesta secção, apresenta-se, introdutoriamente, alguns requisitos de segurança de uma RSSF e alguns serviços de segurança, que representam um ponto de partida para a garantia de propriedades de segurança, aquando do desenho de RSSF seguras.

2.1.1.1 Requisitos de segurança de uma RSSF

Os requisitos de segurança de uma RSSF podem variar consoante as especificidades da aplicação que a rede visa suportar. No entanto, apresentam-se, de forma genérica, os principais requisitos de segurança de uma RSSF [58]:

Autenticação Sendo que o meio de comunicação é partilhado, é necessário recorrer à autenticação para garantir a detecção de mensagens alteradas ou injectadas no sistema, de forma não autorizada [58]. O uso de criptografia assimétrica ainda não é viável nas RSSF, considerando as limitações destas redes e as exigências computacionais[1] destes mecanismos;

Confidencialidade Sendo uma RSSF uma infraestrutura baseada, fundamentalmente, na disseminação de dados recolhidos sensorialmente em ambiente remoto e/ou não controlado e, normalmente, de fácil acesso, é necessário garantir a confidencialidade dos dados que circulam na

[1] Não somente em termos de memória, mas também em termos de energia

rede. O uso de criptografia é o mais indicado para este tipo de protecção, sendo adequada a selecção de algoritmos de encriptação fiáveis (ex: AES[2] [63], ECC[3][63]. Com a utilização de chaves criptográficas, é necessária a adopção de esquemas seguros de distribuição de chaves [30].

Disponibilidade Entende-se por disponibilidade a garantia do funcionamento de uma rede durante a totalidade do tempo de operação. Os ataques que visam afectar esta propriedade são denominados por ataques de negação de serviço (*Denial of Service - DoS*) [33]. Para além de mecanismos que evitem estes ataques, é necessário garantir que a degradação da rede (na presença de um ataque) é controlada, ou seja, é proporcional ao número de nós comprometidos;

Integridade A integridade garante que os dados recebidos por um nó não são alterados, por um atacante, durante a transmissão. Em alguns casos, esta propriedade é garantida juntamente com a autenticação, usando mecanismos que permitem garantir ambas as propriedades numa só operação. É comum o uso de CMAC [4] [63], uma vez que permite autenticar (com o uso de chave criptográfica simétrica) e verificar a integridade de uma mensagem [59].

Detecção de Retransmissão Ilícita (ou Teste de Frescura da Mensagem) A frescura de uma mensagem garante que não é antiga e/ou não foi reenviada por um atacante [59, 45]. Podem-se considerar dois tipos de frescura: fraca (garantindo ordem parcial e sem informação do desvio de tempo, usada para as medições dos sensores) e forte (que garante ordem total em cada comunicação, permitindo estimar o atraso, sendo usada para a sincronização de tempo).

2.1.1.2 Serviços Básicos de Segurança

Alguns serviços de segurança têm vindo a ser desenvolvidos para as RSSF, com vista a garantir a segurança ao nível da comunicação (ex: criptografia, assinaturas, *digests*). Estes serviços permitem que o arquitecto de sistemas se centre noutras problemáticas relacionadas com o comportamento dos protocolos face a ataques, por exemplo, de intrusão. Apresentam-se de seguida alguns serviços mais comuns que representam as arquitecturas básicas de segurança para RSSF:

[2] *Advanced Encryption System*
[3] *Elliptic Curve Cryptography*
[4] *Cipher based Message Authentication Code*

TinySec [42] TinySec é uma arquitectura para protecção ao nível de ligação de dados em RSSF. O objectivo principal é o de providenciar um nível adequado de segurança, com o mínimo consumo de recursos. Os serviços de segurança disponibilizados são: autenticação de dados (com a utilização de *Message Authentication Codes*(MAC) [63], em particular o CBC-MAC[5]) e confidencialidade (CBC-MAC). Não implementa nenhum mecanismo que garanta a frescura das mensagens, tornando-o vulnerável a ataques de retransmissão ilícita;

MiniSec [45] Minisec é uma camada de rede concebida para possuir baixo consumo de energia (melhor que o TinySec) e alta segurança. Uma das características principais, que a tornam mais eficiente, é o uso do modo *Offset Codebook* (OCB) [63] para encriptação de blocos. Desta forma, é possível, numa única passagem, autenticar e encriptar os dados, sem aumentar o tamanho da mensagem, contribuindo para um menor consumo de energia. Esta arquitectura tem dois modos de operação: um orientado para comunicação *unicast* (MINISEC-U) e outro para comunicação *broadcast* (MINISEC-B);

SPINS [59] Conjunto de protocolos de segurança, constituído por dois componentes principais: SNEP[6] [59] e μTESLA [59, 46]. O primeiro fornece serviços de autenticação e confidencialidade *unicast*, encriptando as mensagens (com o modo CTR[7]) e protegendo-as com um MAC (CBC-MAC). O SNEP gera diferentes chaves de encriptação que derivam de uma chave-mestra, partilhada entre dois nós, com um contador de mensagens para garantir a frescura de cada mensagem. O segundo componente, o μTESLA [59, 46], é um serviço de autenticação de *broadcast*, que evita a utilização de mecanismos mais exigentes, de criptografia assimétrica, recorrendo a criptografia simétrica, autenticando as mensagens com um CMAC;

Norma IEEE802.15.4 [16] Esta norma define a especificação da camada física e de controlo de acesso ao meio das redes pessoais de baixa potência (*LRPAN*[8]). Foca-se, essencialmente, na comunicação entre dispositivos relativamente próximos, sem a necessidade de uma infraestrutura de suporte, explorando o mínimo de consumo de energia. É uma norma que já se encontra implementada em algumas plataformas das RSSF (ex: Micaz [9]). Especifica alguns serviços de segurança [21], representando uma primeira linha de protecção contra ataques à infraestrutura.

[5] Cipher Block Chaining - Message Authentication Code (CBC-MAC))
[6] Secure Network Encryption Protocol
[7] *Counter Mode*
[8] Low Rate Personal Area Networks

Estes mecanismos são os seguintes: i) Cada dispositivo mantém uma lista de controlo de acessos (ACL) dos dispositivos confiáveis, filtrando comunicações não autorizadas; ii) Encriptação de dados, partilha de uma chave criptográfica entre os intervenientes na comunicação; iii) Serviço de integridade de cada *frame*, adicionando a cada *frame* um *Message Integrity Code* (MIC) [63]; iv) Garantia de frescura de mensagens (*Sequential Freshness*), utilizando contadores de *frames* e de chaves.

ZigBee [21, 15] Com a norma 802.15.4, orientada para as duas camadas mais baixas da pilha de protocolos das RSSF, a norma ZigBee define as especificações para a camada de rede e de aplicação. Já incorpora alguns serviços de segurança, nomeadamente: i) Frescura, mantendo contadores associados a cada chave de sessão, que são reiniciados em cada mudança de chave; ii) Integridade, com opções de integridade de mensagens que vão desde os 0 aos 128 bits de verificação; iii) Autenticação, ao nível de rede e ao nível de ligação de dados; iv) Confidencialidade, com o algoritmo AES [63] com 128 bits. Esta arquitectura utiliza o conceito de *trusted center* para gestão da segurança na rede, implementando um coordenador de rede ZigBee. Este, acreditado por todos os nós da rede, pode desempenhar três funções: i) Autenticação de nós participantes na rede; ii) Manutenção e distribuição de chaves; iii) Segurança ponto-a-ponto entre nós da rede.

2.1.2 Modelo de Adversário

A definição do modelo de adversário permite, desde logo, identificar as características e as capacidades dos atacantes e os ataques que estes podem desencadear na rede. Nesta secção, caracteriza-se o modelo de adversário que enforma este trabalho.

2.1.2.1 Modelo de Dolev-Yao

Um dos modelos de adversário mais conhecidos, quando se trata de análise formal de protocolos seguros, é o modelo de Dolev-Yao [28]. Neste modelo, é considerado que a rede está sobre o domínio do adversário o qual, perante este facto, pode extrair, reordenar, reenviar, alterar e apagar as mensagens que circulam entre quaisquer dois nós legítimos. Com esta assumpção, entende-se portanto, que o adversário transporta a mensagem e, com isso, adopta um ataque do tipo *man-in-the-middle* [63], com comportamento incorrecto. Este funcionamento, entenda-se, não é comparado à intrusão mas sim à intercepção de mensagens e pode ser mitigado pela

utilização de mecanismos de criptografia.

As tipologias de ataque consideradas pelo modelo de adversário de Dolev-Yao são instanciadas pela norma X800 [39], que pretende normalizar uma arquitectura de segurança para o modelo OSI [64], através de uma abordagem sistemática para o desenho de sistemas seguros. Esta norma considera a segurança sob três aspectos: ataque, mecanismo e serviço de segurança [63]. O primeiro refere-se à forma usada para comprometer um sistema, por exemplo, alterando ou tendo acesso não autorizado a dados desse sistema[9]. O segundo aspecto considerado são os mecanismos de segurança, que se entendem como o processo que permite detectar, prevenir ou recuperar de um ataque à segurança (ex: encriptação, controlo de acesso, assinatura digital) [63]. Por fim, o terceiro aspecto define os serviços que, fazendo uso de um ou mais mecanismos de segurança, permitem resistir a ataques dirigidos a determinada fonte de informação, quer seja durante o processamento, quer seja durante a comunicação. Considera-se, então, que, para efeitos da futura dissertação, os ataques subjacentes ao modelo de Dolev-Yao são protegidos a partir do estabelecimento de uma camada básica de segurança, concretizada por uma das arquitecturas anteriormente referidas na Secção 2.1.1.2.

2.1.2.2 Modelo de Intrusão em RSSF

Considerando o estudo de segurança numa RSSF e, dada a sua exposição natural, nomeadamente a física, colocando cada sensor ao alcance de um adversário, torna-se relevante a consideração de novos modelos de adversário. Cada rede pode ser constituída por milhares de sensores e cada um destes sensores é um ponto de possível ataque [57]. Este ataque pode ser tipificado como sendo por intrusão ou captura.

Este tipo de ataques pode ser desencadeado desde o nível MAC [69] até ao nível de intrusão física. Neste último, um actor externo captura um ou mais sensores legítimos e descobre os segredos criptográficos. Este facto permite-lhe replicar [54] os segredos para sensores maliciosos, introduzido-os na rede de modo a que, agindo coordenadamente, possam comprometer a rede. Conseguida a intrusão, o atacante pode induzir, nos sensores legítimos, comportamentos incorrectos, baseados na informação falsa introduzida pelos sensores maliciosos, influenciando

[9] Na literatura, algumas vezes usam-se os termos "ataque"e "ameaça"para denominarem o mesmo efeito. No entanto, recorrendo ao RFC 2828 [61], podemos definir ameaça como uma potencial violação de segurança, ou seja, é apenas uma vulnerabilidade que pode ser explorada para desencadear um ataque. No caso do ataque, trata-se da exploração inteligente de uma ou mais ameaças que resultam na violação, com sucesso, de um sistema que se pretendia seguro

o processo de encaminhamento. Estes ataques são de difícil detecção, uma vez que o carácter autónomo das RSSF pode não permitir distinguir um comportamento errado de uma falha. Com a intrusão, um sensor malicioso, embora respeitando o protocolo da rede, pode actuar de forma incorrecta, levando a rede a criar topologias específicas para determinado ataque ou forçando toda a informação a passar por nós maliciosos, podendo estes suprimir ou violar a informação.

2.1.2.2.1 Modelo bizantino: adversários bizantinos O modelo de ataques por intrusão tem algumas parecenças com as denominadas falhas bizantinas [23], que são caracterizadas como falhas arbitrárias com as quais um sistema não está, à partida, preparado para lidar e que se podem traduzir em comportamentos inesperados. Transpondo esta realidade para as RSSF [47], é difícil detectar a introdução de nós maliciosos, autónomos ou replicados a partir de um nó que foi comprometido. No entanto, alguns autores [54, 23] têm-se debruçado sobre esta problemática, a fim de dotarem os algoritmos de encaminhamento com mecanismos que permitam detectar a replicação de nós maliciosos numa RSSF. Para se lidar com ataques com comportamentos bizantinos, recorre-se a mecanismos probabilísticos que, ainda que possam não mitigar o ataque por completo, aumentam a resiliência e acabam por transformar um ataque num mal menor, definindo até onde pode ser comprometida a rede, por forma a, ainda assim, garantir a fiabilidade necessária para o seu funcionamento.

2.1.3 Ataques ao Encaminhamento

Apesar de existirem ataques que podem ser dirigidos a qualquer uma das camadas da pilha da RSSF, nesta secção apresentam-se os ataques relacionados com a camada de rede, responsável pelo encaminhamento de dados. Os protocolos de encaminhamento em MANETs [24] e em RSSF, de uma forma geral, decompõe-se em três fases: descoberta dos caminhos, selecção dos caminhos e manutenção da comunicação pelos caminhos seleccionados. Os ataques a um algoritmo de encaminhamento, normalmente, podem explorar as vulnerabilidades de cada uma destas fases de forma específica. Em seguida, os ataques são associados à fase do protocolo em que se podem desencadear e são apresentadas, também, as contra-medidas que permitem mitigá-los.

2.1.4 Ataques à organização da rede e descoberta de nós

Nos protocolos do tipo *table-driven* [19], após a descoberta dos nós vizinhos é necessário recolher informação para a construção das tabelas de encaminhamento. No entanto, em protocolos do tipo *on-demand* [19], esta fase é desencadeada em cada início de transmissão. Este funcionamento corresponde à organização e descoberta de nós numa RSSF.

Falsificação de Informação de Encaminhamento Este ataque tem impacte na formação da rede e na descoberta dos nós. Induz a criação de entradas incorrectas nas tabelas de encaminhamento, podendo também fazer com que estas fiquem lotadas e inválidas. Nos protocolos *on-demand*, o impacte pode ser menor, uma vez que obriga o atacante a injectar informação errada a cada ciclo de transmissão. Outro ataque que causa estes mesmos efeitos é realizado por nós atacantes que inundam a rede com pacotes do tipo *Route Request* (RREQ), pondo em causa a disponibilidade da rede.

Ataques *Rushing* O *Rushing attack* [37] é definido pela exploração, por parte do atacante, de uma janela de tempo para responder a um pedido de caminho para um destino. Este ataque é efectivo quando um protocolo (ex: AODV [56]) aceita a primeira resposta que recebe *Route Reply*(RREP). Explorando isto, o atacante é sempre um candidato a ser o próximo encaminhador, uma vez que não respeita temporizadores nem condições de resposta, podendo depois influenciar o estabelecimento das rotas.

2.1.4.1 Contra-medidas

Os mecanismos de autenticação fazem com que ataques de falsificação de informação ou de inundação de RREQ sejam minimizados. Os nós da rede podem partilhar chaves simétricas (par-a-par) como forma de autenticar as mensagens de dados e controlo do encaminhamento (RREQ e RREP). Desta forma, o atacante, não possuindo a chaves necessárias para a comunicação, não poderá participar no protocolo.

Para fazer face a ataques de *Rushing*, alguns autores [37] apresentam dois mecanismos de defesa: reenvio aleatório de RREQ *Randomized RREQ Forwarding*) e detecção segura (*Secure Detection*). No primeiro caso, cada nó intermédio guarda um conjunto de mensagens RREQ, escolhendo depois, aleatoriamente, um para reenviar. Ainda assim, pode ser seleccionada uma

mensagem RREQ maliciosa, daí a existência do segundo mecanismo, que permite a autenticação de mensagens entre dois nós, garantindo que estas pertencem a nós legítimos. Outros mecanismos passam pela selecção de mais do que uma resposta (permitindo que a mensagem seja enviada por outro caminho) ou pela colecção de várias respostas (escolhendo, aleatoriamente, uma para responder).

2.1.5 Ataques ao estabelecimento de rotas

Os ataques desencadeados nesta fase aumentam a probabilidade de um atacante pertencer a uma rota. Estabelecida a rota através de si pode alterar as mensagens ou agir de forma a desencadear ataques na fase de manutenção de rotas.

HELLO Flooding Este ataque explora os protocolos que se anunciam aos vizinhos, emitindo mensagens de *HELLO* [66, 41]. Os protocolos baseados na localização podem ser vulneráveis a este ataque, uma vez que, com um dispositivo do tipo *laptop-class* [41], que possua um alcance rádio suficientemente potente para cobrir toda a rede, é possível anunciar-se a todos os nós como vizinho, forçando a informação a fluir através dele.

Ataque *Sinkhole* No ataque *sinkhole* [52], o atacante, induz os nós da rede a fazerem passar a informação por dele. Assim, anuncia-se aos nós vizinhos, como tendo boa comunicação com o nó *sink*, tornando-se, assim, um ponto de passagem da informação. O ataque é realizado enviando pacotes de RREQ, alterando a origem e aumentando o número de sequência, como forma de garantir que esta informação se sobrepõe a qualquer informação legítima. Assim, um atacante poderá participar num número elevado de rotas, podendo alterar ou encaminhar, de forma selectiva, a informação. Os protocolos *table-driven* são vulneráveis a estes ataques, enquanto os protocolos baseados em localização não o são, no caso das suas rotas serem estabelecidas *on-demand* [41, 66, 73].

Ataque *Wormhole* Neste tipo de ataque, apresentado por Perrig *et al* [35], dois nós maliciosos colaboram para a realização do ataque. Os atacantes estabelecem uma ligação (em geral, de melhor qualidade) para comunicarem entre si, permitindo a um nó malicioso capturar pacotes ou partes de pacotes e enviá-los pela ligação privada para o outro atacante, noutro extremo da rede. Este ataque é particularmente eficaz em redes RSSF baseadas em localização que, caso sejam comprometidas, não conseguirão estabelecer caminhos maiores do que dois *hops* [72, 66].

Para além disso, os atacantes transformam-se em nós muito solicitados, pois apresentam-se aos outros nós como tendo uma melhor ligação e estando a menor distância do destino.

Ataque *Sybil* Este ataque foi definido como uma acção que permitia atingir os mecanismos de redundância em armazenamento distribuído em sistemas ponto-a-ponto [29]. Outra definição que surge, agora associada às RSSF, é a que o define como "um dispositivo malicioso que ilegitimamente assume múltiplas entidades" [51]. Com estas definições e, devido à sua taxonomia, é um ataque bastante efectivo contra protocolos de encaminhamento [41]. Em particular, nos protocolos que adoptam múltiplos caminhos, o que permite que um nó assuma múltiplas identidades, ocultando o facto de os dados estarem a passar por um único nó malicioso [66, 73].

2.1.5.1 Contra-medidas

Uma das formas de prevenir um ataque HELLO *flooding* [41] é a implementação de mecanismos de resposta (*acknowlege*) a anúncios HELLO. Desta forma, caso o meio de comunicação do atacante cubra toda a rede, um nó legítimo, que não o alcance e, portanto, não receba a resposta, não considerará o anúncio como válido. É possível proceder à autenticação da mensagem, certificando-a numa entidade central que, ao detectar que um nó se anuncia como vizinho de muitos outros nós, toma precauções, repudiando o nó perante a rede [66].

Alguns autores têm vindo a desenvolver algoritmos que visam a detecção de ataques do tipo *Sinkhole* [52]. Um desses mecanismos é o *Sinkhole Intrusion Detection System* (SIDS) [52], orientado para a detecção destes ataques ao protocolo DSR [40]. Este sistema propõe três mecanismos de detecção: i) Descontinuidade de números de sequência (tendo em conta que um atacante tentará usar números de sequência muito grandes, um nó pode identificar os que crescem rapidamente ou os que não respeitam uma ordem crescente, considerando-os um ataque); ii) Verificação de pacotes (os vizinhos podem certificar a origem dos pacotes enviados por um nó. Isto será difícil de realizar nos pacotes atacantes, uma vez que a origem é alterada. Assim, se circularem muitos pacotes não certificados poder-se-á detectar que a rede está sob ataque); iii) Número de caminhos a passar por um nó (cada nó pode observar a sua tabela de encaminhamento e detectar que existem muitos caminhos a passar pelo mesmo nó, logo pode estar na presença de um ataque *Sinkhole* [66, 73]). A utilização de chaves ponto-a-ponto, como forma de garantir que a informação dos pacotes é legítima, evita que um atacante altere dados da mensagem (origem e número de sequência)[68, 27].

A utilização de *packet leashes* [34] permite mitigar o ataque *wormhole*. Assim, existem dois

tipos de condições para se aceitar os pacotes vindos de uma origem: a localização e o tempo. A primeira permite que um nó receptor, conhecendo a localização da origem, saiba se um pacote atravessou a rede por um *wormhole*, calculando a distância entre os dois pontos. O segundo baseia-se, essencialmente, no tempo de transmissão do pacote, exigindo, então, a sincronização de relógios. Se for muito rápido a chegar ao destino, este nó assume que está perante um ataque de *wormhole*. A implementação de encaminhamento por múltiplas rotas, também, permite fazer face a ataques *wormhole* [55].

Para o ataque *sybil* em [51, 66], são possíveis dois esquemas de protecção: i) *Radio resource testing* (cada vizinho só pode transmitir num canal, seleccionando um canal para receber e enviar uma mensagem. Os nós que não responderem são tratados como falsos, ao nível MAC); ii) *Random key distribution* (usando um modelo de *key-pool*, são atribuidas n chaves de um conjunto de m. Se dois nós partilharem q chaves então estarão em condições de comunicar de forma segura. Uma função de dispersão, com base no identificador do nó, permite gerar chaves, evitando que um nó possa conhecer uma grande parte das chaves). A noção de reputação dos nós vizinhos pode também permitir detectar comportamentos incorrectos de atacantes *sybil* [68] ou, alternativamente, realizar o anúncio dos vizinhos de forma autenticada [55].

2.1.6 Ataques à manutenção de rotas

Ataque *Blackhole* No ataque *blackhole* [32], o atacante intercepta os pacotes destinados ao nó/área alvo de ataque, informando a origem de que se trata de um caminho de melhor qualidade ou a menor distância forçando todo o tráfego, dirigido ao destino, a circular através dele. Assim, um nó malicioso intermédio, pode anunciar-se com um caminho melhor, apesar de não ter sequer caminho para o destino, originando um "vazio" e interrompendo o processo de comunicação [66, 73].

2.1.6.1 Contra-medidas

Para mitigar os ataques de *blackhole* existem várias propostas [20, 73, 32], das quais se destacam as que implementam os seguintes mecanismos: i) Confirmação do destino, em que é enviada uma mensagem de ACK por cada mensagem recebida, pelo caminho inverso; iii) Definição de limites de tempo de recepção das mensagens de ACK ou de mensagens de falha; iii) Mensagens de falha, geradas quando um nó intermédio detecta o fim do temporizador de ACK; iv) Caminho definido pela origem, significando que, em cada pacote, é indicado, pela origem, o

caminho seguido até ao destino. Os mecanismos que não se baseiem em informação qualitativa do caminho também permitem resistir a estes ataques [55].

2.1.7 Ataques por Intrusão/Replicação

Alguns dos ataques tipificados anteriormente podem ser desencadeados a partir de nós maliciosos, [52] introduzidos na rede de forma incorrecta e, posteriormente, replicados. Os ataques por intrusão nas RSSF são tipificados pela capacidade de um atacante se apropriar de material criptográfico que, durante o desenrolar do protocolo de encaminhamento, lhe permita participar nas comunicações, fazendo passar-se por um nó legítimo. Este ataque, quando conseguido, é bastante devastador para a rede, por exemplo, quando utilizado para coordenar um ataque *sybil*.

Os ataques por replicação [54] correspondem à introdução de novos nós na rede, clonados de nós legítimos, mas que possuem comportamentos incorrectos, como seja a transmissão de informação para outros nós atacantes, originando ataques tipificados na Secção 2.1.3. Estes ataques, resultantes da replicação, são particularmente efectivos em sistemas do tipo de votação ou cuja operação da rede dependa de mecanismos de eleição. Pode-se, então, dizer que, mitigar os ataques por intrusão/replicação permite que, à partida, se possam reduzir algumas condições para a indução de outros ataques.

2.1.7.1 Contra-medidas

Autenticação central Uma primeira forma de defesa contra replicação é a autenticação dos nós, numa estação central, usando os seus identificadores e permitindo detectar inconsistências ou duplicações[27].

Múltiplas Rotas A existência de diversos caminhos para entregar uma mensagem no destino faz com que se possa aliar a tolerância a falhas à resistência a intrusões. Assim, se a mensagem for interceptada por um intrusor, existe alta probabilidade de esta alcançar o destino, usando outro dos caminhos de envio [27].

Detecção de replicação Perrig e Parno em [54] apresentam mecanismos de detecção distribuída de replicações. Um dos mecanismos é de carácter aleatório *Randomized Multicast* e outro, denominado por *Line-Selected Multicast*, serve-se do modelo de comunicação multi-*hop* da rede para detectar nós duplicados. Outro exemplo é o a criação de estruturas auxiliares (ex: árvores binárias), como acontece no protocolo *Clean-Slate* [55].

2.2 Estudo de Protocolos de Encaminhamento Seguro para RSSF

Como ponto introdutório da discussão e apresentação de algoritmos de encaminhamento em RSSF, importa identificar algumas tipologias ou classes destes algoritmos.

2.2.1 Caracterização dos protocolos de encaminhamento em RSSF

Podem-se estabelecer três classes de protocolos [17]: os baseados na localização, os centrados nos dados e os hierárquicos. Os protocolos baseados na localização usam esta informação para tomarem as melhores decisões para alcançarem os destinos (ex: IGF [22]). Os centrados nos dados, ou seja, os que exploram a semântica dos dados, normalmente são baseados em algoritmos que efectuam pesquisas lançadas a partir de nós de sincronização (ex: Directed Diffusion [38]). Por fim, os protocolos hierárquicos, cuja concepção é baseada na construção de grupos de nós, normalmente denominados como *clusters* (ex: LEACH [31]), que funcionam baseados no princípio de agregação de dados do grupo e na transferência da informação para os nós base.

Para além destas classificações, podemos ainda considerar algoritmos quanto ao momento em que são determinadas as rotas de encaminhamento de dados [19]. Assim, consideram-se os protocolos como *table-driven* ou *on-demand*. Os primeiros referem-se a protocolos que mantêm as tabelas de encaminhamento, trocando mensagens de controlo durante a sua operação. Desta forma, observa-se um maior consumo de energia, devido à regular troca de mensagens. No segundo caso, nos protocolos *on-demand*, as rotas são determinadas em cada envio de mensagem. Apesar de acarretar alguma sobrecarga, em cada envio, acaba por compensar em redes mais móveis e com eventos mais espaçados.

2.2.2 Protocolos de encaminhamento seguro em RSSF

Muitos dos protocolos de encaminhamento para RSSF não foram concebidos tendo em conta o factor da segurança [41, 19]. Em vez disso, pretendiam adaptar-se ao ambiente das aplicações e às características e capacidades das RSSF. No entanto, quando se pretende estender a sua utilização para outros domínios, cuja segurança é indispensável, estas preocupações aumentam, uma vez que os mecanismos de segurança implicam directamente um aumento da computação e um aumento no custo da comunicação, reflectindo-se na autonomia dos sensores.

Nesta secção apresentam-se alguns protocolos de encaminhamento seguro em RSSF que visam cobrir todo o espectro da temática deste trabalho.

2.2.2.1 *Secure Implicit Geographic Forwarding* (SIGF)

Uma das formas de abordar o desenvolvimento de protocolos de encaminhamento seguro é implementar mecanismos de segurança em protocolos já existentes, mas que não seguros. Um destes casos é o algoritmo de encaminhamento *Implicit Geographic Forwarding* (IGF) [22], que deu origem a uma implementação segura: o SIGF [68].

O IGF é um protocolo *on-demand*, baseado na localização que, não mantendo o estado ao longo do seu funcionamento, faz com que não seja necessário o conhecimento da topologia da rede ou a presença de outros nós. O seu carácter não determinístico de encaminhamento já representa um mecanismo de segurança perante determinados ataques, mas não é de forma alguma suficiente para manter uma aplicação, com requisitos de segurança, a executar em ambientes críticos.

Funcionamento do protocolo IGF No protocolo IGF o ambiente está definido por coordenadas que permitem a cada nó saber exactamente a sua localização. Com a agregação do nível de rede e do nível MAC[10] num único protocolo *Network/MAC*, é possível [22], no momento do envio do pacote, determinar qual o próximo melhor candidato para encaminhar os dados. O protocolo inicia-se com a origem a enviar uma mensagem do tipo *Open Request To Send* (ORTS) para a vizinhança (com a localização e o destino). Cada nó que se encontre no sextante válido [11] inicia um temporizador de CTS (*Clear To Send*) inversamente proporcional a determinados parâmetros (distância à origem, energia existente e distância perpendicular ao destino), favorecendo os nós com melhores condições. Ao expirar o temporizador, é enviada uma mensagem de CTS que, ao ser recebida, dá início ao envio de mensagens do tipo DATA a partir da origem. Como este protocolo não mantém estado, resiste a mudanças de topologia da rede. O facto de escolher o nó seguinte, em cada envio, constitui um mecanismo de tolerância a falhas que, em caso de ataque, confina os danos à vizinhança do nó comprometido.

[10] *Medium Access Control*

[11] Ângulo de 60º centrado na origem, orientado para o destino e determinado por cada nó, com base na sua localização

Funcionamento do protocolo SIGF [68] A introdução de mecanismos de segurança, num protocolo existente, compreende um acréscimo de sobrecarga no seu funcionamento. Contudo, o protocolo SIGF [68] pretende manter um bom desempenho e uma elevada taxa de sucesso de entrega das mensagens, mesmo durante um ataque. Uma das características deste protocolo é o facto de ser configurável e, como tal, permitir adaptar os mecanismos de segurança ao grau de ameaça. O SIGF apresenta três extensões ao protocolo IGF [22], o que possibilita a evolução gradual de um protocolo seguro, sem estado, para um protocolo seguro, com manutenção de estado, e, com isto, mais pesado e exigente em recursos.

A primeira extensão é a mais simples e a menos exigente em recursos, o SIGF-0. Continua a não manter o estado e a ter um carácter não determinístico. No entanto, não sucumbe a ataques do tipo *rushing* [37], por não emitir logo para o primeiro nó que lhe envie um CTS. Em vez disso, mantém um conjunto de possíveis candidatos a próximo nó. A extensão intermédia, SIGF-1, já mantém estado, mas ao nível local, podendo com isto constituir listas de reputação dos seus vizinhos, por forma a escolher melhor o próximo nó. Por fim, e tratando-se já de um protocolo mais robusto, mas mais exigente, o SIGF-2 partilha o estado com os seus vizinhos. Permite usar mecanismos criptográficos que garantem integridade, autenticidade, confidencialidade e frescura. Acumula as propriedades de segurança das extensões anteriores: SIGF-0 e SIGF-1.

2.2.2.2 *INtrusion-tolerant routing protocol for wireless SEnsor NetworkS* (INSENS)

Este protocolo [27] foi concebido tendo em vista a tolerância a intrusões e, como tal, faz face a uma das tipologias do modelo de adversário preconizado neste trabalho. Para cumprir com este objectivo, foram identificados dois tipos de ataques: ataques por negação de serviço [33] e ataques ao encaminhamento. O protocolo assenta na existência de uma estação base, constituindo-se como um centro confiável, que partilha chaves criptográficas simétricas com cada um dos nós da rede. Esta característica permite que, em caso de comprometimento de um nó, o atacante não terá acesso a mais do que uma chave segura da rede, isolando, de alguma forma, o ataque.

O uso de caminhos redundantes permite aumentar a resiliência a atacantes não detectados, bastando que exista apenas um caminho sem interposição de atacantes, para que as mensagens cheguem ao destino sem serem comprometidas. Note-se que, neste protocolo, não é possível a comunicação directa entre nós da rede, sem que esta não passe pela estação base. O papel fundamental do protocolo, em termos de encaminhamento seguro, é desempenhado pela estação

base. Uma das vantagens, apontadas pelos autores, é a redução das computações nos nós da rede (ex: para geração de chaves, construção de tabelas de encaminhamento), cujas limitações são as conhecidas. A formação das tabelas de encaminhamento divide-se em três fases: Pedido de rotas (*route request*); Recolha dos dados de encaminhamento; Propagação das rotas. A primeira fase corresponde ao envio, por parte da estação base, de uma mensagem destinada a todos os nós da rede, por forma a obter dados sobre as vizinhanças. Numa segunda fase, cada nó envia a sua vizinhança para a estação base. Por fim, depois da estação base tratar toda a informação recolhida, são elaboradas as tabelas de encaminhamento. As tabelas são depois propagadas para cada nó, podendo prosseguir-se com o encaminhamento dos dados, baseado nas tabelas recebidas.

2.2.2.3 *Secure Sensor Network Routing: Clean-Slate approach*

O algoritmo Clean-Slate [55] foi concebido desde o início, de forma sistemática, com características de segurança. É orientado para a comunicação ponto-a-ponto entre nós da rede, visando a resistência mesmo na presença de um ataque (ataque activo). Classifica-se como um protocolo *table-driven*.

Funcionamento do Protocolo Cada sensor da rede recebe um identificador único global, um certificado assinado por uma autoridade de certificação da rede (AR), a chave pública desta entidade e um conjunto de valores (desafios) baseados numa função de dispersão de um sentido (*one way hash function*). Neste protocolo, podem-se identificar as três fases de operação: organização da rede, estabelecimento dos caminhos e manutenção das rotas.

O protocolo estabelece as tabelas de encaminhamento e os endereços dinâmicos (de tamanho variável) para cada nó da rede, usando um algoritmo recursivo de agrupamento, que executa de forma determinística, mediante uma topologia. Os grupos são formados de forma recursiva e hierárquica, até que a rede forme apenas um único grupo. Em cada fusão é acrescentado um bit (0/1) à esquerda, que permitirá distinguir o endereço de cada nó. Dentro de um mesmo grupo, a comunicação é feita usando *broadcast* autenticado, inspirada no protocolo μTESLA [59, 46].

Este algoritmo incorpora mecanismos de detecção de comportamentos incorrectos dos nós, por exemplo, caso pretendam assumir múltiplas identidades (*sybil* [51, 29]). Este mecanismo é desencadeado após a formação dos grupos, com cada nó a anunciar o seu endereço para os vizinhos, aplicando-se um algoritmo de detecção de replicação de nós [54]. Outro mecanismo

para a detecção de formação incorrecta de grupos é a utilização de *Grouping Verification Trees (GVT)*, baseado em tabelas de dispersão que providenciam autenticação ao nível das folhas, usando a raiz para certificação. Cada nó tem uma GVT, permitindo verificar qualquer comunicação trocada com outros nós da rede.

Durante a fase de manutenção das rotas e encaminhamento, o algoritmo incorpora operações que permitem tratar a saída e entrada de nós. Ao detectar a saída de outro, um nó procura, num dos seus vizinhos, um novo nó fronteira, que lhe permita alcançar o grupo antes acessível pelo nó que saiu. A definição de épocas (*ephocs*) permite que, ao fim de algum tempo, o algoritmo de agrupamento se repita, de forma a incluir novos nós. No que respeita ao encaminhamento, o protocolo usa múltiplas rotas, fazendo com que possa contornar áreas comprometidas da rede. Os nós maliciosos são retirados do algoritmo, usando uma técnica denominada por *Honeybee*. Corresponde ao seguinte: quando um nó malicioso (replicado ou não) é detectado, a rede é inundada com um pacote que indica que o atacante deve ser retirado das tabelas e, tratando-se de uma replicação, o nó replicado autosacrifica-se saindo da rede.

De forma sumária, o protocolo Clean-Slate incorpora os três conceitos para o desenho de protocolos de encaminhamento seguro: prevenção (autenticação), resiliência (múltiplas rotas) e detecção/recuperação (GVT/Honeybee). Implementa-os em simultâneo, ao contrário do que acontece com alguns protocolos que apenas implementam um destes conceitos. É, por isso, um protocolo base, indicado para o estudo comparativo com outros protocolos.

2.3 Ambientes de Simulação

Os ambientes de simulação de RSSF surgem como uma necessidade, inevitável, para o teste e desenvolvimento das redes de sensores e de todas as tecnologias associadas [53, 44]. Alguns ambientes têm sido desenvolvidos especificamente para determinados problemas. Outros são adaptados a partir de ambientes já existentes, como é o caso do NS2 [10] ou J-Sim [5], que foram concebidos para simulações relacionadas com redes convencionais (ex: IEEE802.3, IEEE802.11). A característica importante destes ambientes é a capacidade de repetição de experiências, perante as mesmas condições, facilitando, assim, uma análise sistemática do objecto de estudo.

Nesta secção, apresentam-se diversos ambientes de simulação, mais comuns, e que permitam simular um sistema de RSSF. Foram seleccionados, em primeiro lugar, seguindo critérios relacionados com engenharia de *software*. Seguidamente, com vista à avaliação de um ambiente que se mostre adequado para ser utilizado no trabalho de dissertação, foram estabelecidos critérios relacionados com as RSSF.

2.3.1 Critérios Relacionados com Engenharia de Software

Portabilidade da Linguagem Devido às características da linguagem de programação Java, inerente ao seu ambiente de execução, à consequente portabilidade e à programação orientada a objectos, foram seleccionados apenas ambientes desenvolvidos nesta linguagem.

Código Aberto e Livre Esta propriedade permite que se contornem obstáculos inerentes a licenciamento de *software*, ao mesmo tempo que possibilita a análise e aproveitamento de todas as funcionalidades existentes, permitindo introduzir algumas melhorias ou alterações específicas.

Modularidade e extensibilidade Tendo em conta que os ambientes não possuem todos as mesmas características e funcionalidades e, considerando que a componente experimental da dissertação irá introduzir novos mecanismos, o princípio da modularidade e da fácil extensibilidade facilitará o desenrolar do trabalho.

Documentação Alguns ambientes podem não estar bem documentados. Este critério será importante como ponto de partida para o aprofundamento do conhecimento de cada uma das arquitecturas destas ferramentas.

2.3.2 Critérios Relacionados com as RSSF

Escalabilidade da Rede Uma das características mais importantes das RSSF é o conceito de escala, que se deve ao facto de estas compreenderem, normalmente, um grande número de sensores distribuídos por uma vasta área. Assim, é importante que o ambiente de simulação possa suportar experiências com milhares de nós, uma vez que o factor escala é um das propriedades que se querem ver analisadas no trabalho a desenvolver;

Modelo de Colisões/Comunicação Rádio É fundamental que este modelo se encontre presente no sistema de simulação, por ser um componente base das RSSF, relacionado com a camada de acesso ao meio e de ligação de dados (ex: B-MAC, S-MAC) [60, 70];

Modelo de Gestão de Energia A existência de um modelo de energia permitirá adaptar esta funcionalidade e incluí-la na plataforma final, visto tratar-se de uma das propriedades que se deseja estudar.

Capacidade de Emulação Alguns simuladores possuem a capacidade de emular um sensor real, permitindo efectuar o carregamento de código directamente para o *mote*, sem recurso a recompilação. Não sendo um critério mandatório, reveste-se de algum interesse;

Modelo de Mobilidade Ainda que as RSSF sejam maioritariamente instaladas com características estáticas ou de pouca mobilidade, a existência de um modelo de mobilidade poderá possibilitar a avaliação dos comportamentos dos protocolos, mediante esta propriedade;

Interface de Visualização É importante que o ambiente de simulação possua uma interface de visualização, permitindo uma percepção mais fácil dos comportamentos dos protocolos (ex: topologia, cobertura), bem como o controlo da simulação e extracção de resultados.

Modelo de Gestão de Topologia Um factor que pode influenciar o comportamento de um protocolo de encaminhamento é a topologia. Como tal, é relevante a existência deste modelo por forma a avaliar os protocolos desenhados, perante diferentes topologias.

2.3.3 Prowler/JProwler

[6]Esta ferramenta resulta da conversão de um simulador de eventos discretos[12], Prowler [7], implementado em MATLAB, pela Universidade de Vanderbilt, para a linguagem Java. Este simulador pode ser configurado para simular, de forma determinística ou probabilística. Permite a simulação, com diversos nós, podendo atingir os 5000 (ainda que o número possa ser maior,

[12]Fila global, onde são inseridos todos os eventos da rede, tratados sequencialmente ou por prioridade.

por razões de performance, este é o valor máximo aconselhado), usando diversas topologias (dinâmicas/aleatórias), às quais se podem sujeitar diversos algoritmos.

O JProwler modela os aspectos mais importantes do modelo de comunicação de uma RSSF. A natureza não-determinística da propagação rádio é caracterizada por um modelo de rádio probabilístico simples e preciso, que descreve a operação da camada MAC. Possui uma janela de visualização da topologia da rede. Para o desenvolvimento de aplicações ou protocolos são disponibilizadas classes base que se podem estender. Estão presentes dois modelos de rádio: um de Gauss, para topologias estáticas, e outro de Rayleigh, para topologias móveis.

2.3.4 J-Sim

J-Sim (anteriormente conhecido como JavaSim) é um ambiente de simulação baseado em componentes [5], implementado em Java. Não foi desenvolvido inicialmente com vista à sua utilização em RSSF, como é o caso do ambiente SENSE [11], mas o objectivo de extensibilidade é comum. Este ambiente é amplamente usado e implementa um modelo de rede em camadas. No entanto, este simulador não é o mais adequado para o estudo do desempenho em RSSF, visto que este é condicionado pelo *hardware*, pelo sistema operativo, pelos protocolos de rede e pelas aplicações, assim como pelas optimizações específicas entre camadas da pilha de protocolos. Apesar disto, o J-Sim é um importante ambiente de simulação, dada a natureza fracamente ligada dos seus componentes, a qual permite o desenvolvimento e prototipagem de aplicações. Exige, no entanto, algum conhecimento profundo da arquitectura, mesmo para a implementação de protocolos simples.

2.3.5 Freemote

Fremote é uma ferramenta de emulação [4] distribuída[13], desenvolvida em Java, utilizada para o desenvolvimento de *software* para RSSF. O emulador suporta *motes* (Squawk, Sentilla) e plataformas (Java Cards, SunSpot [13]), baseados em Java. Divide a arquitectura em camadas bem definidas por interfaces: Aplicação, Encaminhamento e Ligação de Dados/MAC. Tem um interface gráfico para configuração. Suporta experiências de grande escala (10.000 nós), incluindo a sua integração com nós reais, baseados em Java. Os principais pontos negativos são: i) O modelo de propagação rádio é muito simples, uma vez que não considera obstáculos

[13]Funciona em rede, com a possibilidade de ter diversos clientes de visualização, ligados a um servidor central.

entre os nós; ii) Só existe um modelo de comunicação real, limitado a emulação simples de plataformas específicas (JMote); iii) Não é orientado para a análise de performance das redes, característica que pode ser importante no desenvolvimento de algoritmos para RSSF.

2.3.6 ShoX

A ideia principal deste simulador [12] é a de proporcionar, de uma forma fácil e intuitiva, a implementação e desenho de protocolos de rede, modelos de mobilidade, modelos de propagação de sinal ou de tráfego de rede. Tal como outros simuladores, incorpora um simulador de eventos discretos, que faz a gestão de todos os eventos da rede. Todos os conceitos conhecidos no domínio das redes sem fios são modelados neste simulador (modelo OSI, pacotes, mobilidade e energia). Uma das vantagens é a existência de classes abstractas para reimplementação de novos modelos em cada um dos componentes, facilitando a programação de novos protocolos ou de novas funcionalidades. A comunicação entre componentes é feita por intermédio de eventos, ou seja, não existe acesso de um componente a outro. Deve-se destacar o interface gráfico, que permite operar todas as configurações da ferramenta, sem a necessidade de editar directamente os ficheiros de XML. Para além disso, é ainda possível visualizar a topologia de rede e extrair resultados gráficos da simulação. O facto de o modelo de propagação de sinal ser baseado na norma IEEE802.11 dificulta a adaptação às condições das RSSF. No entanto, a modularidade do sistema permite o desenvolvimento de uma camada IEEE802.15.4 para se aproximar da norma mais recente de comunicação das RSSF[14]. A arquitectura deste simulador aproxima-se bastante daquilo que deve ser um simulador de RSSF, em que as diversas camadas estão bem definidas.

[14]Não cabe no âmbito da dissertação o desenvolvimento do módulo de comunicação

2.4 Discussão e Resumo do Trabalho Relacionado

As RSSF representam um enorme desafio para a investigação de sistemas e protocolos de segurança. As características que as tornam uma mais-valia para a operação em ambientes remotos, apresentam-se, simultaneamente, como as suas maiores vulnerabilidades, em termos de segurança. Este paradoxo é contornado com mecanismos de segurança inovadores e que se distinguem dos existentes nas redes convencionais. Assim, passada em revista as diversas dimensões que se pretendem abarcar na futura dissertação (protocolos de encaminhamento seguro em RSSF e plataformas de simulação de RSSF), importa apresentar uma visão crítica do trabalho relacionado.

Na Tabela 2.1 apresenta-se uma visão estruturada das contra-medidas que permitem mitigar ou diminuir o impacte dos ataques nas RSSF. O uso de criptografia simétrica é predominante, uma vez que representa uma forma de garantir propriedades, tais como, confidencialidade, autenticidade e integridade. O uso de funções de dispersão de um sentido permite verificar a integridade e, aliado ao uso de "desafios"[15], permite garantir a frescura das mensagens, a um custo computacional reduzido. É de salientar que as implementações destes mecanismos são optimizadas para a redução dos custos computacionais e de comunicação. Esta tabela propicia uma visão mais actualizada das contra-medidas face aos ataques referenciados no modelo de adversário, fruto da consolidação do trabalho relacionado.

Modelos	Ataque	Contramedidas
Dolev-Yao	Ataque ao meio de comunicação	Criptografia simétrica, *One Way Hashing*
Organização e Descoberta da Rede	Falsificação de informação de Routing	Autenticação, *One Way Hashing*
	Ataques de *Rushing*	Selecção aleatória de RREQ, autenticação, verificação bidirectional
Estabelecimento de Rotas	HELLO flooding	Autenticação com verificação bidirectional(*acknowledge*)
	Ataques *Sinkhole*	Autenticação, Distribuição de chaves *pairwise*
	Ataques *Wormhole*	*Packet leaches*, MAC
	Ataques *Sybil*	Distribuição de chaves *pairwise*, selecção aleatória de canais de rádio
Manutenção de Rotas	Ataques de *Backhole*	Definição de temporizadores e mecanismos de confirmação (ACK) autenticados
Modelo de Intrusão	Intrusão	Encaminhamento multi-rota; *One Way Hashing*
	Replicação	Certificação central; Autenticação; Nós vizinhos como testemunhas

Tabela 2.1 Tabela de Ataques *vs* Contramedidas

Tendo em conta as contra-medidas apresentadas, cabe analisar, comparativamente, os algoritmos e a capacidade de resistir aos ataques definidos no modelo de adversário. Assim, na

[15]Números sequenciais, cuja sequência depende da aplicação de uma função $f^n(x) = C$, n vezes, por forma a obter C (*challenge*)

Tabela 2.2, estão marcados com o símbolo ✓ os ataques defendidos por cada protocolo estudado, sendo que os marcados com × não defendem ou só o fazem em condições especiais.

O protocolo SIGF distingue-se dos demais protocolos estudados, particularmente pela sua origem (extensão do IGF) e por ser baseado em localização. Esta característica é específica para determinadas aplicações e obriga à existência de dispositivos especializados nos *motes*. Para além disto, é particularmente indicado para a monitorização de eventos, cuja ocorrência é espaçada no tempo. Por ser configurável, faz com que se adapte consoante o grau de ameaça, aumentando os custos de operação com o aumento da ameaça.

Os protocolos Clean-Slate e INSENS, que serão usados como prova de conceito na elaboração da dissertação, não necessitam de conhecimento de localização, diminuindo a complexidade da plataforma de rede. Estes protocolos, devido às características que os definem, são excelentes candidatos a um estudo comparativo. Uma das características que os distingue é a utilização da estação base como unidade central de encaminhamento, pelo INSENS. Contrariamente, o Clean-Slate tem uma abordagem completamente distribuída. A questão que se levanta, no INSENS, prende-se, essencialmente, com o impacte no consumo energético dos nós próximos (a um *hop*) da estação base, uma vez que esta encaminhará todo o tráfego da rede.

	Ataques ao Encaminhamento							**Intrusão**	**Comunicação**
Protocolos	**Info. Falsa**	***Rushing***	**HELLO flooding**	***Sinkhole***	***Wormhole***	***Sybil***	***Blackhole***	***Intrusão/Replicação***	***Dolev-Yao***
SIGF	✓	✓	✓	×	×	✓	✓	×/×	✓
INSENS	✓	✓	✓	✓	✓	✓	✓	✓/×	✓
Clean-Slate	✓	✓	✓	✓	✓	✓	✓	✓/✓	✓

Tabela 2.2 Tabela de Protocolos de Encaminhamento *vs* Ataques

Ambos os protocolos implementam resiliência à intrusão. Uma das diferenças é que o Clean-Slate tem uma acção preventiva, correspondente à detecção dos nós maliciosos (replicados). Além disso, faz encaminhamento multi-rota, o que minimiza o impacte dos intrusores que resistam à detecção. No caso do INSENS, apenas existe um mecanismo redundante multi-rota. Observa-se que os autores do Clean-Slate efectuaram algum estudo comparativo, teórico, relativamente ao INSENS (por este ser já existente). A implementação destes dois algoritmos, numa mesma base de simulação, permitirá verificar se as observações realizadas vão ao encontro das conclusões dos autores. Este estudo comparativo será de relevante importância para o estudo de outros protocolos que surjam na academia, relacionados com a problemática da segurança no encaminhamento de dados nas RSSF. Em relação aos ataques que podem ser direccionados a cada um dos protocolos, é importante realçar o ataque *sybil*, como um caso de difícil resolução, quando este derivar de uma intrusão, em que um atacante obteve as chaves necessárias para

se poder anunciar com qualquer uma das identidades que assumir. Assim, a resistência a este ataque, por parte dos protocolos estudados, não leva em conta este modo de intrusão, porque assume que cada atacante não tem todas as chaves necessárias para se autenticar com falsas identidades e que, como tal, pode ser detectado.

		Ambientes de Simulação				
Critérios		**Prowler/JProwler**	**J-Sim**	**Freemote**	**ShoX**	**Nova Plataforma**
Software	**Portabilidade da linguagem**	Java	Java	Java	Java	Java
	Código Livre Aberto	Sim	Sim	Sim	Sim	Sim
	Modularidade e extensabilidade	Sim	Sim (JTcl)	Sim	Sim	Sim
	Documentação	Apenas comentários no código	Papers e Online	Pouca	Pouca	
Propriedades das RSSF	**Escalabilidade**	Aprox. 5000 nós	Documentado na ordem dos milhares	Documentado na ordem dos milhares	Foram testados 100 nós sem sucesso	Na ordem dos milhares
	Colisões/Comunicação	Sim, modelo B-MAC	Sim, mas 802.11	Sim, mas muito simples	Sim, mas 802.11	Sim
	Gestão de Energia	Não	Não	Não	Sim	Sim
	Emulação	Não	Não	Sim, para plataformas Java *Based*	Não	Não
	Mobilidade	Sim, mas rudimentar	Sim	Sim	Sim	Não
	Visualização	Sim, mas só de visualização da topologia	Existem ferramentas auxiliares	Sim	Sim, mas não em tempo real. Apenas depois de executar a simulação	Sim
	Topologia	Não existe de raiz, pode ser modelado	Não existe de raiz, pode ser modelado	Não existe de raiz, pode ser modelado	Existem alguns de raiz, podendo ser estendido	Existirão de raiz alguns modelos, permitindo a adição de mais

Tabela 2.3 Tabela de Critérios de Avaliação *vs* Ambientes de Simulação

Por fim e sendo a análise de ambientes de simulação um dos focos do trabalho relacionado, apresenta-se na Tabela 2.3 a sua sistematização, contrapondo os critérios aos ambientes estudados. Como uma primeira nota, deve-se salientar que, tendo em vista a concepção de uma plataforma de simulação, a observação dos critérios de *software* para a selecção de um simulador base teve um peso bastante grande, nomeadamente no que se refere à capacidade de extensibilidade reflectida na simplicidade. Com efeito, afigura-se que, observando a Tabela 2.3,

a selecção do ambiente base poderia recair sobre as plataformas Shox ou Freemote, uma vez que estas apresentam características que se aproximam mais dos objectivos da plataforma final. No entanto, depois de alguma engenharia inversa em cada um dos componentes, observa-se que, apesar da riqueza dos componentes, a complexidade para integração de novas funcionalidades ou a introdução de mecanismos de controlo que permitam extrair medições, se revela muito grande.

Perante a necessidade de seleccionar um ambiente que vise alcançar os objectivos da dissertação em tempo útil, esta selecção recai, preliminarmente, no simulador JProwler. A sua simplicidade é uma mais valia, uma vez que, sendo composto por oito classes, bem documentadas, é de mais fácil extensibilidade quanto à implementação das funcionalidades requeridas na plataforma a conceber. Cada componente de um nó sensor é mapeado numa classe abstracta. O modelo de comunicação é inspirado no Mica2 com a gestão de eventos baseada no TinyOS [14], o que o torna, do ponto de vista da sua aproximação à realidade, bastante vantajoso. É certo que algumas funcionalidades têm que ser implementadas de raiz, como é o caso da energia (existente no ShoX) ou gestão de topologias (existente no ShoX e no Freemote), o que permitirá desenvolver modelos que podem ser bem integrados de forma observar-se melhor as propriedades desejadas. O caso da emulação (existente no Freemote) não é um requisito mandatório da plataforma. Logo, a sua utilização não se apresenta como uma mais-valia, face aos aspectos menos positivos. O Freemote assenta sobre uma interacção cliente/servidor, o que torna difícil a depuração de erros. Possui um modelo de comunicação demasiado básico, não incorporando atenuações do meio ambiente, o que inviabiliza a sua selecção. O J-Sim é realmente uma plataforma poderosa, mas a sua dimensão obrigaria a um esforço adicional, demasiado grande, na medida em que necessitaria de ver desenvolvido muitos dos seus módulos de raiz.

Desta forma, perante a maior complexidade de algumas plataformas, o JProwler parece constituir uma base simples, eficiente e, simultaneamente, com capacidade para ser enriquecida, de forma a integrar a plataforma de simulação. Ainda assim, alguns dos conceitos presentes nos outros simuladores poderão prestar significativos contributos conceptuais para o resultado final da plataforma a conceber na dissertação.

3 . Abordagem à fase de elaboração da dissertação

Tendo sido abordadas as temáticas relacionadas com a problemática da segurança numa RSSF, importa, então, definir uma estratégia para a concepção de uma plataforma que vise a análise e avaliação de protocolos de encaminhamento, principalmente os concebidos com requisitos de segurança. Neste capítulo, apresenta-se, de forma preliminar, as fases da elaboração da dissertação, referentes à concepção dos modelos que suportam a arquitectura da plataforma. Adicionalmente, apresenta-se uma prova de conceito, referente à utilização da plataforma, com a implementação de dois protocolos, Clean-Slate e INSENS, e o consequente estudo comparativo.

3.1 Desenho e concepção da plataforma de simulação

3.1.1 Consolidação da avaliação de ambientes de simulação e a sua incorporação

Do estudo das plataformas de simulação existentes e apresentadas neste relatório, será seleccionado um motor de simulação com os modelos base para simulação de um ambiente de RSSF (comunicação, gerador de eventos discretos e plataformas de sensores). A fase de elaboração da dissertação deverá contar, inicialmente, com o aprofundamento da avaliação dos ambientes de simulação apresentados, com o intuito de certificar a selecção do simulador base. Uma mais-valia deste estudo será o aproveitamento de características presentes em diferentes ambientes e que possam ser implementadas na plataforma final.

3.1.2 Apresentação preliminar da arquitectura da plataforma de simulação

A visão mais simples para representar a plataforma é sob a forma de uma pilha de serviços. Como se pode ver na Figura 3.1, os principais serviços são: i) Mecanismo de Geração de Topologias; ii) Mecanismo de Gestão de Consumo de Energia; iii) Mecanismo de Injecção de Faltas/Ataques ao Encaminhamento; iv) Mecanismo de Configuração; v) Consola de Visualização e Controlo de Simulação.

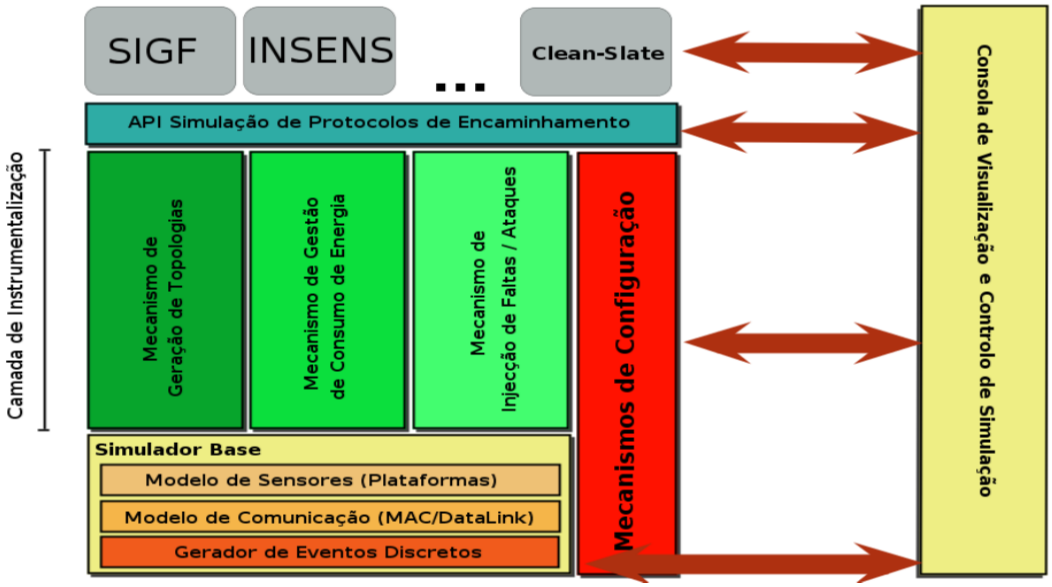

Figura 3.1 Arquitectura de Simulação

3.1.2.1 Mecanismo de Configuração

Para dotar a plataforma de maior flexibilidade, a existência de um componente gestor de configurações revela-se importante. Este componente deve ser transversal a toda plataforma. Para que as parametrizações possam ser persistentes e portáveis, adoptar-se-á a tecnologia XML para a definição dos ficheiros de configuração da plataforma. As principais funcionalidades que se prevêem existir vão desde as configurações dos parâmetros do simulador base, até à configuração de cada uma das simulações que se pretendem estudar, como forma de possibilitar a repetição de experiências nas mesmas condições.

3.1.2.2 Mecanismo de Geração de Topologias

As RSSF, normalmente, são caracterizadas por diferentes formas de distribuição dos nós sensores. Estas distribuições podem ser essencialmente divididas em dois modelos: aleatório e estruturado. Assim, para que se consiga observar as características que se pretendem analisar num protocolo de encaminhamento, será fornecido um componente cuja função é gerar topologias na rede. Sabe-se que, de facto, a topologia da rede pode influenciar o comportamento de um protocolo. Pretende-se que este componente possibilite a extensão para novas topologias (específicas para determinada simulação).

3.1.2.3 Mecanismo de Gestão de Consumo de Energia

As características fundamentais que se deverão implementar neste componente são: a definição de uma interface para o desenvolvimento de novos modelos consoante as necessidades da simulação; a possibilidade de introduzir parâmetros relacionados com o consumo energético dos sensores. Associadas a este componente estarão funcionalidades que permitirão a recolha de informação em tempo real dos consumos da rede, quer no seu total, quer individualmente, em cada sensor. Este é um dos componentes de maior importância, uma vez que um dos indicadores que se pretende observar na análise de protocolos de encaminhamento é o impacte sobre o tempo útil de operação da rede, quer em condições de funcionamento normais, quer em condições de ataque efectivo, tempo este que está dependente da energia.

3.1.2.4 Mecanismo de Injecção de Falhas / Ataques ao Encaminhamento

Sendo o tema central da futura dissertação o estudo de protocolos de encaminhamento seguro em RSSF, este componente é o de maior importância nesta plataforma, por diferentes ordens de razões: i) Não existe nenhum sistema de simulação que permita a indução de ataques de forma simples e intuitiva, consubstanciado-se, por isso, num contributo para a inovação; ii) Deverá ser suficientemente flexível para se adaptar à lógica de cada algoritmo; iii) Poderá permitir a mutação de código em tempo de execução da simulação, por forma a alterar comportamentos do protocolo; iv) Idealmente deverá permitir acrescentar mais modelos de ataques, dos já tipificados neste relatório ou de outros que venham a ser identificados.

3.1.2.5 Consola de Visualização e Controlo de Simulação

Existe a necessidade de dotar a plataforma de uma consola de operação. Como tal, é necessário implementar um componente que permita a visualização gráfica de toda a simulação, bem como o controlo de parâmetros de execução. Pretende-se desenvolver um ambiente gráfico integrado, que permita a configuração da plataforma, a configuração e visualização das simulações e a extracção de resultados relacionados com as medidas que se pretendem avaliar: energia, fiabilidade e cobertura, principalmente sob a forma de gráficos.

3.1.2.6 Avaliação da Solução

Uma vez que a contribuição efectiva para a componente de investigação de protocolos de encaminhamento seguros em RSSF será obtida com a concepção de uma plataforma de simulação que suporte o estudo e a análise desta problemática, importa sujeitá-la a uma avaliação primária que permita comprovar a sua utilidade e/ou identificar eventuais lacunas neste domínio. Assim, tendo esta avaliação em vista, pretende-se contribuir com o estudo dos protocolos de encaminhamento seguro Clean-Slate e INSENS. Para isso, definem-se duas fases complementares na elaboração da tese: uma que visa a implementação de um protocolo simples, que permita aferir o correcto funcionamento da plataforma, e outra que vise a implementação dos dois protocolos referidos, que serão alvo de uma análise comparativa com recurso às funcionalidades da plataforma.

3.1.3 Implementação de Protocolos de Encaminhamento Seguro em RSSF

3.1.3.1 Fase de desenho dos algoritmos baseado nas especificações

No início desta fase, será necessário re-aprofundar o funcionamento de cada algoritmo a implementar, conhecer e identificar cada mecanismo/técnica especificados, de modo a que se possa, dentro do possível, generalizar operações ou interfaces, com vista à reutilização para outros algoritmos. Assim sendo, esta fase exigirá uma aprendizagem/conhecimento de cada algoritmo, contribuindo também para a especialização neste domínio.

3.1.3.2 Fase de avaliação dos algoritmos

Recorrendo às ferramentas disponibilizadas pela plataforma, deverá ser possível, no final da implementação, sistematizar as simulações, de maneira a extrair resultados. Estes resultados, por si só, devem caracterizar os algoritmos em matéria de segurança e quanto à sua correcção em determinados parâmetros, a saber: i) correcção do protocolo; ii) análise do consumo de energia; iii) fiabilidade/entrega de mensagens; iv) correcção dos eventos; v) latência. Esta fase contribui, também, para a aferição a usabilidade da plataforma em termos de avaliação/comparação de protocolos de encaminhamento seguro em RSSF.

4 . Plano de Elaboração da Tese

A elaboração da tese realizar-se-á durante o 2º semestre de 2009/2010, iniciando a 22 de Fevereiro de 2010. O plano apresentado estabelece cinco grandes actividades: análise, desenvolvimento, prova de conceito, avaliação e relatório, como se apresenta na Figura 4.1.

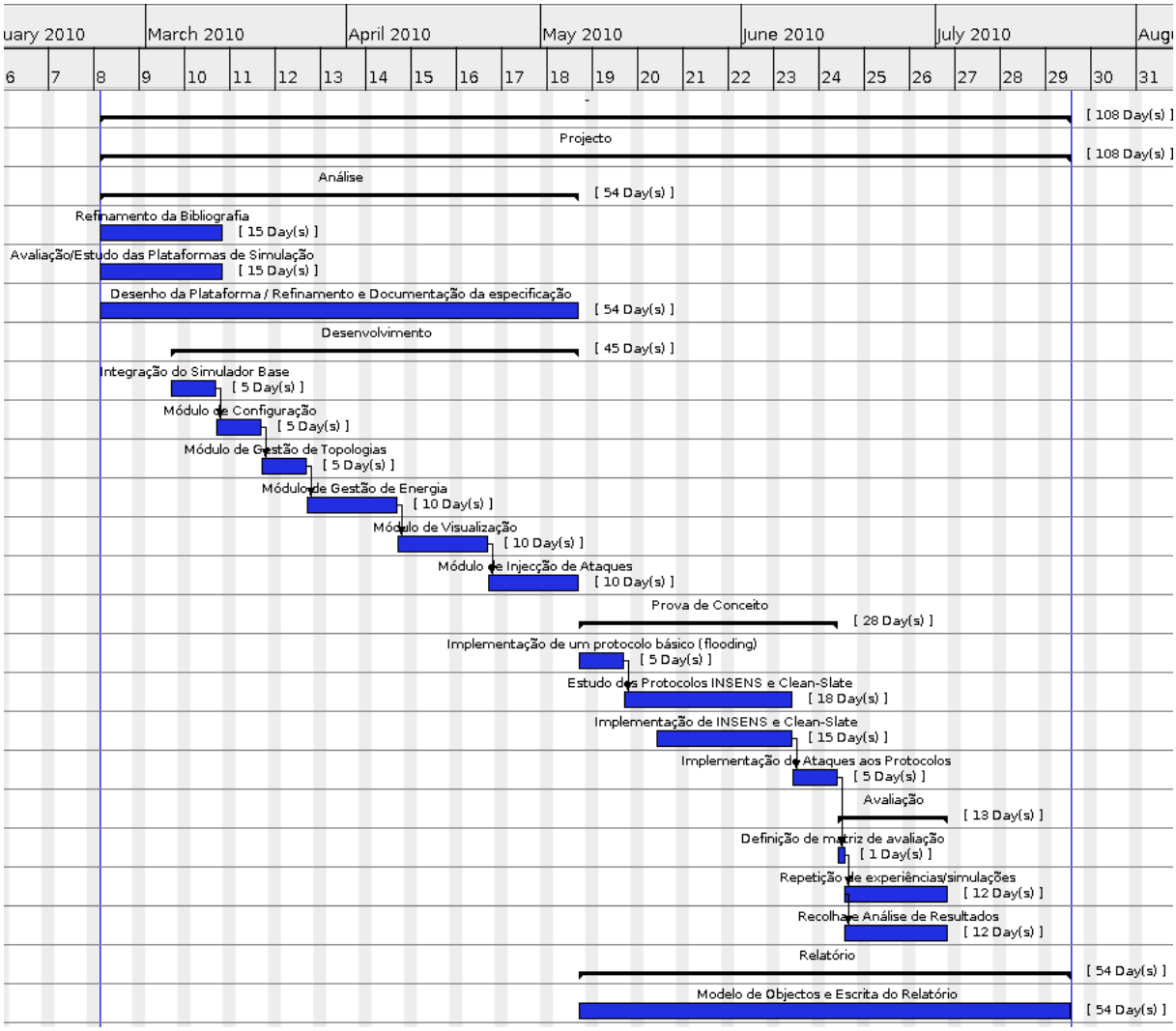

Figura 4.1 Plano da Dissertação

Apresenta-se, em seguida, uma breve descrição de cada uma das actividades.

Análise Esta actividade corresponde à revisão da bibliografia complementar, com um duplo objectivo: aprofundar a problemática em estudo e avaliar o simulador base, como preconizado na secção 3.1.1. Inicia-se, também, o desenho e a especificação da plataforma, consistindo na definição formal de algoritmos, de interfaces e do modelo de interacção dos componentes da plataforma. Durante a fase de desenvolvimento, esta actividade será revisitada, com vista a refinar/actualizar a especificação da plataforma, nomeadamente pelo recurso a ferramentas de modelação de sistemas orientados a objectos (UML).

Desenvolvimento Esta actividade corresponde à concepção e implementação da arquitectura da plataforma, como apresentada na secção 3.1.2, em que cada um dos componentes corresponderá às seguintes tarefas: Integração do Simulador Base, Módulo de Configuração (secção 3.1.2.1), Módulo de Gestão de Topologias (3.1.2.2), Módulo de Gestão de Energia (secção 3.1.2.3), Módulo de Visualização (secção 3.1.2.5) e Módulo de Injecção de Ataques (secção 3.1.2.4).

Prova de Conceito Esta actividade corresponde à implementação de um algoritmo básico (ex: *flooding* sem repetição de mensagens enviadas) e dos algoritmos de encaminhamento seguro propostos (ver secção 3.1.3.1): INSENS e Clean-Slate. Esta implementação deverá ser precedida de um estudo mais aprofundado das particularidades de cada um. Consequentemente, cada protocolo será sujeito a um modelo de ataque, que permitirá avaliar o seu comportamento, com vista a verificar as propriedades, identificadas anteriormente como importantes, para a análise de um protocolo de encaminhamento seguro em RSSF.

Avaliação Esta actividade preconiza a avaliação dos protocolos implementados, usando as ferramentas da plataforma, como referido na secção 3.1.3.2. Esta avaliação permitirá, também, retirar conclusões acerca da usabilidade da plataforma (secção 3.1.2.6) e dos objectivos pretendidos. Estes passam pela capacidade de estudo de protocolos de encaminhamento em RSSF, em geral, e, em particular, os que têm preocupações de segurança.

Relatório Esta actividade corresponde à escrita da dissertação e deverá iniciar-se assim que termine o processo de especificação da plataforma, permitindo a realização do modelo de objectos. Esta fase poderá decorrer em paralelo com a actividade de avaliação e, eventualmente, com a de prova de conceito. Culminará com a entrega da dissertação até à data limite.

Bibliografia

[1] BTnodes - a distributed environment for prototyping ad hoc networks. http://www.btnode.ethz.ch/Documentation/BTnodeRev3HardwareReference.

[2] The contiki operating system - home. http://www.sics.se/contiki/.

[3] Gateways - crossbow technology. http://www.xbow.com/Products/productdetails.aspx?sid=159.

[4] Home | freemote emulator | assembla. http://www.assembla.com/wiki/show/freemote.

[5] Home (J-Sim official). http://sites.google.com/site/jsimofficial/.

[6] ISIS - JProwler. http://w3.isis.vanderbilt.edu/projects/nest/jprowler/.

[7] ISIS - prowler. http://w3.isis.vanderbilt.edu/projects/nest/Prowler/index.html.

[8] Mica2 Datasheet. http://www.xbow.com/Products/productdetails.aspx?sid=174.

[9] MicaZ Product Details. http://www.xbow.com/Products/productdetails.aspx?sid=164.

[10] The network simulator - ns-2. http://www.isi.edu/nsnam/ns/.

[11] SENSE-3.0: sensor network simulator and emulator. http://www.ita.cs.rpi.edu/sense/index.html.

[12] ShoX project page. http://shox.sourceforge.net/.

[13] SunSPOTWorld - documentation. http://sunspotworld.com/docs/index.html.

[14] TinyOS community forum || an open-source OS for the networked sensor regime. http://www.tinyos.net/.

[15] ZigBee alliance. http://www.zigbee.org/Default.aspx.

[16] IEEE Standard for Information Technology - Telecommunications and information exchange between systems - Local and metropolitan area networks - specific requirement Part 15.4: Wireless Medium Access Control (MAC) and Physical Layer (PHY) Specifications fo, 2007.

[17] Kemal Akkaya and Mohamed Younis. A survey on routing protocols for wireless sensor networks. *Ad Hoc Networks*, 3(3):325–349, May 2005.

[18] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci. Wireless sensor networks: a survey. *Computer Networks*, 38(4):393–422, March 2002.

[19] J.N. Al-Karaki and A.E. Kamal. Routing techniques in wireless sensor networks: a survey. *Wireless Communications, IEEE*, 11(6):6–28, 2004.

[20] Mohammad Al-Shurman, Seong-Moo Yoo, and Seungjin Park. Black hole attack in mobile ad hoc networks. In *ACM-SE 42: Proceedings of the 42nd annual Southeast regional conference*, pages 96–97, New York, NY, USA, 2004. ACM.

[21] Paolo Baronti, Prashant Pillai, Vince W.C. Chook, Stefano Chessa, Alberto Gotta, and Y. Fun Hu. Wireless sensor networks: A survey on the state of the art and the 802.15.4 and ZigBee standards. *Computer Communications*, 30(7):1655–1695, May 2007.

[22] M. Blum, Tian He, Sang Son, and John A Stankovic. IGF: a State-Free robust communication protocol for wireless sensor networks.

[23] Miguel Castro and Barbara Liskov. Practical byzantine fault tolerance. In *OSDI '99: Proceedings of the third symposium on Operating systems design and implementation*, pages 173–186, Berkeley, CA, USA, 1999. USENIX Association.

[24] S Corson and J Macker. Mobile Ad hoc Networking (MANET): Routing Protocol Performance Issues and Evaluation Considerations, January 1999.

[25] George F. Coulouris and Jean Dollimore. *Distributed systems: concepts and design*. Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA, 1988.

[26] I. Demirkol, C. Ersoy, and F. Alagoz. Mac protocols for wireless sensor networks: a survey. *Communications Magazine, IEEE*, 44(4):115–121, April 2006.

[27] Jing Deng, Richard Han, and Shivakant Mishra. Insens: Intrusion-tolerant routing for wireless sensor networks. *Comput. Commun.*, 29(2):216–230, 2006.

[28] D. Dolev and A. Yao. On the security of public key protocols. *Information Theory, IEEE Transactions on*, 29(2):198–208, 1983.

[29] John Douceur and Judith S Donath. The Sybil Attack. pages 251—-260, 2002.

[30] Laurent Eschenauer and Virgil D. Gligor. A key-management scheme for distributed sensor networks. In *Proceedings of the 9th ACM conference on Computer and communications security*, pages 41–47, Washington, DC, USA, 2002. ACM.

[31] Wendi Rabiner Heinzelman, Anantha Chandrakasan, and Hari Balakrishnan. Energy-efficient communication protocol for wireless microsensor networks. *Hawaii International Conference on System Sciences*, 8:8020, 2000.

[32] Hongmei Deng, Wei Li, and D.P. Agrawal. Routing security in wireless ad hoc networks. *Communications Magazine, IEEE*, 40(10):70–75, 2002.

[33] Fei Hu and Neeraj K. Sharma. Security considerations in ad hoc sensor networks. *Ad Hoc Networks*, 3(1):69–89, January 2005.

[34] Y.-C. Hu, A. Perrig, and D.B. Johnson. Packet leashes: a defense against wormhole attacks in wireless networks. In *INFOCOM 2003. Twenty-Second Annual Joint Conference of the IEEE Computer and Communications. IEEE Societies*, volume 3, pages 1976–1986 vol.3, March-3 April 2003.

[35] Yih-Chun Hu, A. Perrig, and D.B. Johnson. Wormhole attacks in wireless networks. *Selected Areas in Communications, IEEE Journal on*, 24(2):370–380, Feb. 2006.

[36] Yih-Chun Hu and Adrian Perrig. A Survey of Secure Wireless Ad Hoc Routing. *IEEE Security and Privacy*, 2(3), 2004.

[37] Yih-Chun Hu, Adrian Perrig, and David B. Johnson. Rushing attacks and defense in wireless ad hoc network routing protocols. In *WiSe '03: Proceedings of the 2nd ACM workshop on Wireless security*, pages 30–40, New York, NY, USA, 2003. ACM.

[38] Chalermek Intanagonwiwat, Ramesh Govindan, Deborah Estrin, John Heidemann, and Fabio Silva. Directed diffusion for wireless sensor networking. *IEEE/ACM Trans. Netw.*, 11(1):2–16, 2003.

[39] ITU-T. Recommendation X.800: Security Architecture for Open Systems for CCITT Applications, 1991.

[40] David B. Johnson and David A. Maltz. Dynamic source routing in ad hoc wireless networks. In *Mobile Computing*, pages 153–181. Kluwer Academic Publishers, 1996.

[41] Chris Karlof and David Wagner. Secure routing in wireless sensor networks: attacks and countermeasures. *Ad Hoc Networks*, 1(2-3):293–315, September 2003.

[42] Chris Karlof, David Wagner, and Naveen Sastry. TinySec: a link layer security architecture for wireless sensor networks. pages 162–175, Baltimore, MD, USA, 2004. ACM.

[43] Mauri Kuorilehto, Marko Hännikäinen, and Timo D. Hämäläinen. A survey of application distribution in wireless sensor networks. *EURASIP Journal on Wireless Communications and Networking*, 2005(5), 2005.

[44] Johannes Lessmann, Peter Janacik, Lazar Lachev, and Dalimir Orfanus. Comparative study of wireless network simulators. In *ICN '08: Proceedings of the Seventh International Conference on Networking*, pages 517–523, Washington, DC, USA, 2008. IEEE Computer Society.

[45] Mark Luk, Adrian Perrig, Ghita Mezzour, and Virgil Gligor. MiniSec: a secure sensor network communication architecture. pages 479–488, Cambridge, Massachusetts, USA, 2007. ACM.

[46] Mark Luk, Adrian Perrig, and Bram Whillock. Seven cardinal properties of sensor network broadcast authentication. pages 147–156, Alexandria, Virginia, USA, 2006. ACM.

[47] Ruiping Ma, Liudong Xing, and H.E. Michel. Fault-intrusion tolerant techniques in wireless sensor networks. In *Dependable, Autonomic and Secure Computing, 2nd IEEE International Symposium on*, pages 85–94, 29 2006-Oct. 1 2006.

[48] Alan Mainwaring, David Culler, Joseph Polastre, Robert Szewczyk, and John Anderson. Wireless sensor networks for habitat monitoring. In *WSNA '02: Proceedings of the 1st ACM international workshop on Wireless sensor networks and applications*, pages 88–97, New York, NY, USA, 2002. ACM.

[49] Aleksandar Milenković, Chris Otto, and Emil Jovanov. Wireless sensor networks for personal health monitoring: Issues and an implementation. *Comput. Commun.*, 29(13-14):2521–2533, 2006.

[50] Wireless Networks. Security Vulnerabilities In Wireless Sensor Networks: A Survey. *Journal of Information Assurance and Security*, 5(2010):031–044, 2009.

[51] J. Newsome, E. Shi, D. Song, and A. Perrig. The sybil attack in sensor networks: analysis & defenses. In *Information Processing in Sensor Networks, 2004. IPSN 2004. Third International Symposium on*, pages 259–268, April 2004.

[52] E.C.H. Ngai, Jiangchuan Liu, and M.R. Lyu. On the intruder detection for sinkhole attack in wireless sensor networks. In *Communications, 2006. ICC '06. IEEE International Conference on*, volume 8, pages 3383–3389, June 2006.

[53] Sung Park, Andreas Savvides, and Mani B. Srivastava. Simulating networks of wireless sensors. In *Proceedings of the 33nd conference on Winter simulation*, pages 1330–1338, Arlington, Virginia, 2001. IEEE Computer Society.

[54] B. Parno, A. Perrig, and V. Gligor. Distributed detection of node replication attacks in sensor networks. pages 49–63, 2005.

[55] Bryan Parno, Mark Luk, Evan Gaustad, and Adrian Perrig. Secure sensor network routing: a clean-slate approach. In *CoNEXT '06: Proceedings of the 2006 ACM CoNEXT conference*, pages 1–13, New York, NY, USA, 2006. ACM.

[56] C.E. Perkins and E.M. Royer. *Ad-hoc on-demand distance vector routing*. IEEE, 1999.

[57] Adrian Perrig and Haowen Chan. Security and Privacy in Sensor Networks.

[58] Adrian Perrig, John Stankovic, and David Wagner. Security in wireless sensor networks. *Commun. ACM*, 47(6):53–57, 2004.

[59] Adrian Perrig, Robert Szewczyk, Victor Wen, David Culler, and J. D. Tygar. Spins: Security protocols for sensor networks. In *Wireless Networks*, pages 189–199, 2001.

[60] Joseph Polastre, Jason Hill, and David Culler. Versatile low power media access for wireless sensor networks. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 95–107, New York, NY, USA, 2004. ACM.

[61] RFC2828. Internet Security Glossary, 2000.

[62] E. Shi and A. Perrig. Designing secure sensor networks. *Wireless Communications, IEEE*, 11(6):38–43, Dec. 2004.

[63] William Stallings. *Cryptography and Network Security (4th Edition)*. 2005.

[64] Andrew S. Tanenbaum and Maarten Van Steen. *Distributed Systems: Principles and Paradigms*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2001.

[65] Hua-Wen Tsai, Chih-Ping Chu, and Tzung-Shi Chen. Mobile object tracking in wireless sensor networks. *Comput. Commun.*, 30(8):1811–1825, 2007.

[66] Yong Wang, G. Attebury, and B. Ramamurthy. A survey of security issues in wireless sensor networks. *Communications Surveys & Tutorials, IEEE*, 8(2):2–23, Quarter 2006.

[67] B.A. Warneke and K.S.J. Pister. Mems for distributed wireless sensor networks. In *Electronics, Circuits and Systems, 2002. 9th International Conference on*, volume 1, pages 291–294 vol.1, 2002.

[68] Anthony D. Wood, Lei Fang, John A. Stankovic, and Tian He. SIGF: a family of configurable, secure routing protocols for wireless sensor networks. pages 35–48, Alexandria, Virginia, USA, 2006. ACM.

[69] Yang Xiao, Hsiao-Hwa Chen, Bo Sun, Ruhai Wang, and Sakshi Sethi. MAC security and security overhead analysis in the IEEE 802.15.4 wireless sensor networks. *EURASIP J. Wirel. Commun. Netw.*, 2006(2):81–81, 2006.

[70] Wei Ye, J. Heidemann, and D. Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *Networking, IEEE/ACM Transactions on*, 12(3):493–506, June 2004.

[71] Jennifer Yick, Biswanath Mukherjee, and Dipak Ghosal. Wireless sensor network survey. *Computer Networks*, 52(12):2292–2330, August 2008.

[72] Hu. Yih-Chun and A. Perrig. A survey of secure wireless ad hoc routing. *Security & Privacy, IEEE*, 2(3):28–39, May-June 2004.

[73] W. You-Chiun and Y Tseng. Attacks and defenses of routing mechanisms in ad hoc and sensor networks. In *Security in Sensor Networks*.