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

Dissertação de Mestrado

Mestrado em Engenharia Informática

Avaliação Experimental de Esquemas de Pré-Distribuição de Chaves em Redes de Sensores sem Fios

Pedro Miguel dos Reis Nunes (28037)

Lisboa
(2011)

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

Dissertação de Mestrado

Avaliação Experimental de Esquemas de Pré-Distribuição de Chaves em Redes de Sensores sem Fios

Pedro Miguel dos Reis Nunes (28037)

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.

Lisboa
(2011)

Aos meus pais, à minha irmã e à Luciana...

Agradecimentos

Em primeiro lugar gostaria de agradecer ao Professor Henrique Domingos pela oportunidade dada em realizar a presente dissertação neste tema e pelo seu apoio e orientação ao longo da mesma.

Agradeço também a todos os meus colegas e amigos que me acompanharam ao longo deste percurso e que enriqueceram o meu processo de formação académica e pessoal.

Mas quero agradecer sobretudo aos meus pais, por todo o apoio e valores que me transmitiram; à minha irmã, pela amizade e pelos conselhos dados; à Luciana, por todo o amor e carinho; e ainda aos meus tios, Tila e João, que apesar da distância que nos separa são um dos pilares da minha vida.

Resumo

A presente dissertação pretende, através de uma análise experimental por simulação, complementar os estudos e análises teóricas existentes na literatura sobre o desempenho de esquemas de pré-distribuição de chaves em redes de sensores sem fios. Os referidos estudos não levam em linha de conta factores subjacentes ao funcionamento deste tipo de redes em condições aproximadas a condições reais, nomeadamente quanto à avaliação de critérios de auto-organização e cobertura topológica da rede bem como de condicionalismos dessa topologia face a condições de fiabilidade, latência e consumo energético. Neste âmbito, factores que influenciam a operação das redes que estão associados às características dos protocolos de acesso ao meio de comunicação por radiofrequência e a condições ambientais têm que ser considerados.

Na dissertação, o foco da análise é dirigido a redes 802.15.4 e ambientes experimentais de simulação e emulação de código para sensores de tecnologia TinyOS, nomeadamente compostas por dispositivos do tipo MicaMotes e TelosB, programáveis em linguagem de programação nesC.

O ambiente de simulação utiliza, na sua base, um núcleo baseado no emulador TOSSIM, complementado pela utilização do módulo de simulação de consumo energético PowerTOSSIM-Z.

A este núcleo foram acrescentados três outros módulos que correspondem a contribuições relevantes da dissertação: uma camada de suporte à comunicação segura que concretiza o modelo TinySec, um módulo de visualização gráfica de topologias e um módulo de interface gráfica para mecanismos de gestão de simulações. Este último módulo permite definir diversos parâmetros de simulação, nomeadamente: número de nós da rede, tipo de topologia e esquema de pré-distribuição de chaves a analisar.

De forma a analisar o desempenho dos vários esquemas de pré-distribuição de chaves que foram objecto de estudo neste trabalho, acrescentaram-se também módulos para obtenção de resultados e indicadores associados à eficácia do estabelecimento de chaves. Assim, foram concebidos, implementados e adicionados ao ambiente de simulação os seguintes módulos: módulo para teste de consumo energético que considera os consumos introduzidos pela camada de suporte à comunicação segura (não contemplados no módulo de simulação de consumo energético PowerTOSSIM-Z); módulo para teste de cobertura e avaliação de condições de conectividade da rede, após estabilização do processo de distribuição de chaves; módulo para teste de fiabilidade com medição de taxas de entrega de pacotes; módulo para teste de latência de estabilização do processo de estabelecimento de chaves bem como da latência da disseminação de dados sobre a topologia resultante de cada esquema de distribuição de chaves.

Palavras-chave: Redes de Sensores sem Fios (RSSF), Métodos e protocolos para gestão e distribuição de chaves, Segurança em RSSF, Análise experimental

Abstract

This dissertation is focused on the experimental simulation, evaluation and analysis of key-distribution and key-establishment schemes and protocols for wireless sensor networks (WSNs). The objective is to complement the theoretical studies and theoretical analysis of some well-known protocols, found in the literature, with a more wide and complete perspective of their expected performance in operation conditions, closer to real operation settings.

In general, the existent studies in the literature don't take into account factors underlying the real operation of WSNs, namely regarding the evaluation of auto-organization and topological coverage and connectivity of the network, as well as constraints of the generated topology when faced with reliability, latency and energetic consumption criteria. These criteria are closely-related with the characteristics of radio communication protocols, data-link behavior and medium access control schemes. At the same time, the network operation is strongly influenced by environmental conditions like radio interferences, collisions, packet losses and the life-cycle of sensor operation, as supported by specific operating systems. In the theoretical analysis these factors are not considered, requiring complementary evaluation combining simulation techniques and subsequent calibration from real-operation.

In this thesis, the focus is oriented to the experimental analysis and simulation of key-distribution and key-establishment schemes in 802.15.4 WSNs, composed by TinyOS sensor motes, namely: MicaMotes and TelosB devices, which are programmable in nesC. The simulation and emulation environment used, at the base of the dissertation, is a TOSSIM simulation and emulation kernel, complemented by the PowerTOSSIM-Z energy consumption simulation module.

The above core was used as a leverage solution to build a more complete simulation environment, composed by three additional modules that correspond to relevant contributions: a secure communication layer implementing TinySec cryptographic computations, a topological graphic visualization module and a graphic interface module for simulations' management. This last module allows to set up several simulation parameters, namely: number of network nodes, type of initial topology and key predistribution scheme to be analyzed, allowing the visualization of the generated topology after the stabilization of the key-distribution process and evaluating relevant criteria for a comparative analysis of different key-distribution protocols, from a uniform and systematic simulation base.

The implemented simulation environment includes modules to evaluate the following metrics: energetic consumption, including consumptions introduced by the secure communication layer (that weren't contemplated on PowerTOSSIM-Z model); network coverage and connectivity metrics; reliability metrics; latency of the key-distribution process stabilization; and latency of data-dissemination over the topology generated after the stabilization of each key distribution scheme.

In the environment, each key-distribution scheme appears as a configurable simulation component applied to a starting topology, to organize the output topology after the key-establishment process.

During the dissertation, three key-distribution schemes have been implemented, tested and evaluated. This evaluation allowed a systematic analysis of the comparative performance of these protocols. At the same time, the development, validation and evaluation of these protocols allowed to validate the simulation platform, as it was designed.

Keywords: Wireless Sensor Networks (WSN), Key-Management and Distribution: Methods and Protocols, WSN Security, Experimental analysis

Conteúdo

Lista de Figuras

Lista de Tabelas

1

Introdução

1.1 Motivação e enquadramento da dissertação

As redes de sensores sem fios (RSSF) tornaram-se numa área de grande interesse por parte da comunidade de investigação. Atendendo às suas características [34, 36, 23], estas redes são particularmente adequadas ao suporte de aplicações inovadoras, entre as quais se destacam as seguintes: monitorização de habitats naturais ou de condições ambientais [25]; monitorização de indicadores biomédicos [27]; vigilância de instalações industriais críticas; monitorização da operação de infra-estruturas de engenharia civil [12]; controlo e vigilância na monitorização e localização de pessoas ou mercadorias [17]; controlo de fenómenos naturais na área da vulcanologia ou sismologia [40]; e aplicações militares de monitorização de campos de batalha ou de detecção de intrusões em zonas sob controlo militar [19]. Em diferentes cenários e no âmbito das aplicações anteriores, as RSSF operam sem supervisão humana e requerem que a sua instalação se faça a partir de condições de auto-organização.

Algumas das aplicações das RSSF e os cenários em que estas operam impõem requisitos de segurança que estão associados a diferentes níveis da estruturação da arquitectura e da pilha de serviços de software. Os requisitos de segurança podem estar associados a falhas ou ataques que podem ocorrer ao nível das comunicações sem fios, considerando-se as tipologias de ataques semelhantes às conceptualizadas pela Framework X.800 [20] ou modelo de adversário de Dolev-Yao [13]. As possibilidades de ataque podem no entanto alargar-se a tipologias de ataques em que comportamentos

maliciosos podem ser induzidos no processamento dos nós e da rede no seu conjunto, partindo da captura e intrusão ao nível dos nós sensores ou da eventual replicação desses comportamentos através da adição de nós controlados pelo adversário. Por outro lado, os serviços de segurança para RSSF requerem técnicas que se mostrem adequadas às características dos dispositivos que são utilizados como nós dessas redes, atendendo às suas limitações em relação a recursos computacionais, de comunicação e de gestão e consumo energético.

A necessidade de soluções de segurança adequadas às RSSF tem motivado a investigação de diferentes tipos de mecanismos, técnicas e serviços que podem actuar complementarmente a diferentes níveis da organização de uma pilha de suporte: i) serviços ao nível MAC ou de suporte básico de comunicação segura sem fios, onde imperam os ataques que podem estar dirigidos à negação de serviços ou à indução de alterações de funcionamento do protocolo de acesso ao meio com impacto no tempo de vida útil da rede [28]; ii) serviços ao nível rede, introduzindo noções de segurança no processo de auto-organização topológica, bem como em relação ao endereçamento e encaminhamento seguro de dados; iii) serviços ao nível de aplicações específicas, nomeadamente mecanismos para localização segura de dispositivos ou métodos seguros para agregação e processamento intermédio desses mesmos dados.

Em virtude da literatura relativa a esta temática abordar apenas análises teóricas de condições de cobertura da rede, revelou-se preponderante realizar análises de teor experimental que pudessem medir e avaliar o desempenho e eficácia de esquemas e protocolos de distribuição e estabelecimento de chaves.

1.2 Problemática da distribuição e estabelecimento de chaves em RSSF

Uma das dimensões importantes para o estabelecimento de condições de segurança em RSSF baseia-se na adopção de métodos criptográficos. Estes têm que ser concebidos de forma a serem adequados às características dos sensores e às limitações importantes ao nível das capacidades de computação e impacto energético. Associada à utilização dos métodos e modelos de protocolos de comunicação segura que utilizam fundamentos da criptografia computacional aplicável ao domínio das RSSF surge a necessidade de se proporem esquemas seguros de gestão e distribuição de chaves ou segredos criptográficos que também sejam adequados às condições de funcionamento dessas redes. Os protocolos de distribuição e estabelecimento de chaves ou segredos criptográficos devem garantir condições de refrescamento para atender a requisitos de rejuvenescimento ou reconfiguração topológica da rede, ou como reconfiguração de condições de

segurança que se possam associar a mecanismos de detecção ou prevenção de intrusões. Para tal, diversos métodos e soluções para distribuição e estabelecimento seguro de chaves em RSSF têm sido propostos na investigação recente, com abordagens diferenciadas.

No âmbito da presente dissertação interessam particularmente referir os métodos de distribuição e estabelecimento probabilístico de chaves inspirados em condições de auto-organização autónoma das RSSF e baseados no estabelecimento dinâmico de chaves par-a-par, a partir de chaves primárias pré-instaladas anteriormente ao processo de auto-organização e cobertura da rede. De entre estes, são particularmente representativos os esquemas e protocolos definidos em [16, 11, 24, 7, 8, 14, 30]. Em geral, estes métodos visam a utilização de criptografia pouco exigente quanto à complexiade de processamento e quanto a exigências do processo de comunicação, o que permite a sua utilização em condições de baixo consumo energético, limitando-se a soluções que envolvem métodos criptográficos simétricos, funções de síntese segura de dados e protocolos de autenticação inspirados em modelos do tipo HMAC ou CMAC. Para além disso, dependendo do tipo de aplicação a que a rede se destina, esta poderá adoptar uma organização plana ou uma organização hierárquica baseada em *clusters*. A utilização deste último tipo de organização poderá justificar-se caso as operações de processamento na rede e de agregação de dados sejam relevantes para a aplicação em causa.

1.3 Enquadramento dos objectivos da dissertação e sua oportunidade

Os métodos de distribuição e estabelecimento de chaves acima referidos são tanto mais eficazes quanto mais propiciem boas condições de refrescamento dinâmico de chaves com emparelhamento probabilístico par-a-par e suportem critérios de segurança futura e segurança passada. Estas propriedades podem permitir estratégias para reorganizações topológicas da rede ou rejuvenescimento da rede por adição de novos nós. A avaliação experimental dessas características em termos de latência da estabilização do processo de refrescamento, impacto energético ou de análise das condições realistas de cobertura torna-se um aspecto relevante.

Por outro lado, as particularidades das RSSF requerem que as soluções para gestão e estabelecimento seguro de chaves permitam a satisfação de critérios, tais como [41]: i) escalabilidade – permitindo o suporte de redes compostas por milhares de nós sensores; ii) eficiência – possibilitando que os protocolos se adaptem às limitações dos recursos de computação, comunicação e de energia; iii) resiliência – garantindo níveis

adequados de segurança mesmo face a condições de eventual comprometimento dos nós por captura de segredos ou chaves criptográficas; e iv) flexibilidade – de modo a poderem ser integradas e reutilizadas no âmbito de protocolos de encaminhamento ou suporte a diferentes aplicações específicas que exijam diferentes estratégias de auto-organização topológica das redes.

Na literatura, o estudo de esquemas e protocolos de distribuição e estabelecimento de chaves abarcam normalmente análises teóricas de condições de cobertura da rede ou análises de complexidade de custo de comunicação em condições teóricas (em relação ao número de mensagens ou rondas dos protocolos que estabelecem as chaves). Estas análises, aplicadas a métodos probabílisticos de chaves, assentam naturalmente numa análise probabílistica, bem como em abordagens de análise de cobertura baseadas em teoria de grafos, o que permite prever condições teóricas ideais de cobertura da rede mas que estão longe da situação real. A avaliação teórica não é complementada por uma avaliação experimental relativa ao impacto desses esquemas nas condições de funcionamento real da rede, nomeadamente face ao modelo de comunicação rádio e protocolo MAC (nas condições de funcionamento das pilhas 802.15.4 [2] ou ZigBee [5]). Também não levam em linha de conta factores externos do meio ambiente e que afectam o funcionamento da rede (condições de humidade ou características mais realistas do raio de alcance de comunicações rádio face à densidade e número de sensores). Na literatura também não é dedicada a atenção devida ao impacto do funcionamento dos protocolos propostos em relação às condições de energia da rede. Finalmente, o impacto da eficácia ou eficiência do processo de estabelecimento de chaves para diferentes condições de topologia (e.g. condições de adaptabilidade a organizações planas ou a organizações hierárquicas) também não é avaliado [30].

Torna-se pois necessário que as avaliações teóricas que se encontram na literatura sejam complementadas por uma análise experimental que se aproxime mais das diferentes premissas de funcionamento real da rede, o que se conseguirá através de uma aproximação por simulação e emulação da mesma. Na abordagem do estudo de protocolos de distribuição e estabelecimento de chaves com base em análise experimental por simulação, devem avaliar-se os critérios anteriormente indicados de escalabilidade, eficiência, resiliência e flexibilidade através de uma avaliação sistemática "vis-à-vis"desses diferentes esquemas. Por outro lado, estes ambientes de simulação e emulação podem servir ainda para calibração posterior das condições de análise dos protocolos, antecipando o funcionamento de RSSF reais em cenários de maior escala.

1.4 Objectivos e contribuições da dissertação

A presente dissertação tem em vista a concepção, implementação e teste de um ambiente de simulação e emulação de RSSF, para avaliação, numa base experimental e sistemática, de diversos esquemas de gestão e distribuição segura de chaves criptográficas. Nessa avaliação visar-se-ão os esquemas e protocolos de estabelecimento e distribuição de chaves inspirados em esquemas de pré-distribuição de chaves de "setup" com posterior refrescamento dinâmico de chaves com emparelhamento par-a-par e em condições probabilísticas. Assim, no âmbito e foco da presente dissertação, visar-se-á a avaliação de esquemas que propiciam a auto-organização flexível da rede com base em processos de cobertura aleatória, tomando-se como referência os seguintes modelos: esquema básico (ou *Random Key Predistribution Scheme* de Gligor e Eschenauer [16]), esquema baseado em pré-distribuição partilhada aleatória Q-composta (*Shared-Key Threshold R-KPS* de Chan, Perrig e Song [11]) e esquema aleatório para arquitecturas em *cluster* (ou modelo SecLEACH [30]).

As contribuições previstas para a dissertação serão assim:

- Concepção e implementação de um ambiente de simulação e emulação para RSSF baseado em dispositivos de tecnologia TinyOS, programados com recurso à linguagem nesC [18] e suportando plataformas de sensores reais MicaMotes ou TelosB Crossbow. O ambiente de simulação utilizará, na sua base, um núcleo baseado no emulador TOSSIM, complementado pela utilização do módulo de simulação de consumo energético PowerTOSSIM-Z. A este núcleo serão acrescentados três outros módulos que correspondem a contribuições relevantes da dissertação: uma camada de suporte à comunicação segura que concretiza o modelo TinySec , um módulo de visualização gráfica de topologias e um módulo de interface gráfica para mecanismos de gestão de simulações. Este último módulo permite definir diversos parâmetros de simulação, nomeadamente: número de nós da rede, tipo de topologia e esquema de pré-distribuição de chaves a analisar. De forma a analisar o desempenho dos vários esquemas de pré-distribuição de chaves que serão objecto de estudo neste trabalho, acrescentar-se-ão também módulos para obtenção de resultados e indicadores associados à eficácia do estabelecimento de chaves. Assim, serão concebidos, implementados e adicionados ao ambiente de simulação os seguintes módulos:

  - módulo para teste de consumo energético que considera os consumos introduzidos pela camada de suporte à comunicação segura (não contemplados no módulo de simulação de consumo energético PowerTOSSIM-Z [31]);

  - módulo para teste de cobertura e avaliação de condições de conectividade da rede, após estabilização do processo de distribuição de chaves;
  - módulo para teste de fiabilidade com medição de taxas de entrega de pacotes;
  - módulo para teste de latência de estabilização do processo de estabelecimento de chaves bem como da latência da disseminação de dados sobre a topologia resultante de cada esquema de distribuição de chaves.

- Validação e utilização do anterior ambiente de simulação/emulação para a condução de um estudo sistemático e comparativo de métodos representativos de opções de distribuição, estabelecimento e refrescamento probabilístico de chaves com base em esquemas de emparelhamento aleatório a partir de pré-distribuição de chaves "setup", nomeadamente os seguintes:
  - esquema baseado em cobertura aleatória a partir de pré-distribuição de chaves (*Random Key Predistribution Scheme* de Gligor e Eschenauer) [16];
  - esquema baseado em pré-distribuição partilhada aleatória Q-composta (*Shared-Key Threshold R-KPS* de Chan, Perrig e Song) [11];
  - esquema aleatório para arquitecturas em *cluster* (modelo SecLEACH) [30].

- A anterior validação basear-se-á nos seguintes critérios de avaliação (que estarão associados a módulos de teste e avaliação a serem concebidos e implementados no ambiente de simulação/emulação a criar):
  - Avaliação do custo energético associado ao processo de auto-organização e cobertura da rede a partir da distribuição e estabelecimento de chaves par-a-par (energia gasta para estabilização do processo de auto-organização e estabelecimento de chaves iniciais face a diferentes níveis de densidade geográfica de nós e número de nós em presença) bem como do custo energético relativo à fase operacional da rede após o processo de estabelecimento de chaves estar concluído;
  - Avaliação de critérios de latência associada ao processo de auto-organização através do estabelecimento de chaves par-a-par (tempo necessário para que seja efectuado o processo de auto-organização e estabelecimento de chaves iniciais face a diferentes níveis de densidade geográfica de nós e número de nós em presença) bem como da latência de disseminação de mensagens sobre a topologia resultante de cada esquema de distribuição de chaves;

  - Avaliação de critérios de escala e cobertura da rede (número de nós cobertos após o processo de estabelecimento de chaves face a diferentes níveis de densidade geográfica de nós e número de nós em presença);
  - Avaliação de taxas de entrega de pacotes de dados (qual a taxa de pacotes que, após o processo de organização e estabelecimento de chaves iniciais, foram entregues com sucesso).

1.5 Estrutura do documento

Os restantes capítulos do presente relatório estão organizados do seguinte modo:

- O capítulo 2 apresenta uma introdução resumida das RSSF, salientando as suas principais características e realçando aspectos relevantes para a problemática tratada na presente dissertação;
- O capítulo 3 é dedicado a uma apresentação do estado da arte, cobrindo os aspectos de trabalho relacionado mais directamente associados aos objectivos da dissertação, a saber:
  - Apresentação de tipologias de ataques que permitam ter uma maior percepção dos perigos e ameaças que este tipo de redes enfrenta;
  - Definição de um modelo de adversário que servirá como base para identificar as tipologias de ataques a considerar durante todo o estudo e análise a realizar no âmbito da presente dissertação (tendo por base a tipologia de ataques inspirados na Framework X.800 [20] complementada pela visão de ataques à disseminação e encaminhamento de dados tratados em [39]);
  - Apresentação dos princípios de funcionamento de diferentes esquemas de distribuição, estabelecimento e refrescamento probabilístico de chaves, com base em esquemas de emparelhamento aleatório a partir de pré-distribuição de chaves "setup" [16, 11, 24, 7, 8, 14, 30];
  - Apresentação de uma análise comparativa de ambientes de simulação ou de emulação para RSSF, cujos serviços base possam ser utilizados para alavancar um núcleo de simulação/emulação com as características associadas aos objectivos da dissertação.
- No capítulo 4 será apresentado o modelo arquitectural da plataforma de simulação a desenvolver. Este modelo arquitectural compreende os seguintes componentes: núcleo de simulação TOSSIM, módulo para cálculo de consumo energético PowerTOSSIM-Z, módulo de visualização gráfica de topologias, camada de

suporte à comunicação segura, esquemas de pré-distribuição de chaves e, finalmente, módulos de extracção de indicadores.

- O capítulo 5 irá abordar particularidades e decisões tomadas durante a fase de implementação de alguns dos elementos que compõem a arquitectura desenvolvida no âmbito desta dissertação, mais concretamente: módulos de extracção de indicadores, esquemas de pré-distribuição de chaves e camada de suporte à comunicação segura.

- O capítulo 6 apresenta os testes efectuados, os resultados obtidos e as análises efectuadas a cada uma das experiências.

- No capítulo 7 tecem-se algumas conclusões sobre o trabalho no seu aspecto global, bem como uma análise sintética de alguns dos aspectos mais relevantes observados durante as avaliações experimentais.

2

Introdução às RSSF

2.1 Redes de sensores sem fios (RSSF)

As redes de sensores sem fios (RSSF) são redes de comunicação por radiofrequência que utilizam a norma IEEE 802.15.4 [2], ZigBee [5], ou outras possíveis variantes de protocolos de acesso ao meio e suporte de ligação de dados [44]. Estas redes são formadas por dispositivos (sensores) de baixo custo e de pequenas dimensões (da ordem de grandeza das dezenas de milímetros cúbicos às dezenas de centímetros cúbicos). Os sensores são dotados de capacidades computacionais, energéticas e de comunicação muito limitadas [21, 39], podendo ser distribuídos ao longo de uma determinada área geográfica formando redes de comunicação mais ou menos densas, aproveitando o baixo custo dos dispositivos e a facilidade de instalação dos mesmos sem custos operacionais relevantes. Para o efeito são particularmente importantes as características de auto-organização da rede e não necessidade de supervisão da operação [37].

Numa RSSF, os sensores cooperam entre si podendo monitorizar eventos que resultam da interacção entre os dispositivos e o meio ambiente, medindo valores associados a diferentes tipos de fenómenos físicos. Na tecnologia actual, os sensores para RSSF são pequenos dispositivos computorizados baseados em substratos mais ou menos genéricos, sobre os quais poderão existir um ou mais sensores (propriamente ditos). Aquele tipo de substrato envolve um microprocessador, um circuito de comunicação por rádio (IEEE 802.15.4) e um conversor analógico-digital [1] com capacidades para processamento de sinal associado a determinado tipo de eventos: temperatura, som ou ruído,

humidade, movimento, poluição ou níveis de concentração de certo tipo de substâncias, pressão, vibração ou luminosidade. Existem ainda sensores especializados para medir indicadores fisiológicos ou sinais vitais de saúde.

Neste tipo de redes, os eventos são medidos, detectados e pré-processados localmente com base em técnicas de pré-processamento de sinal, sendo posteriormente difundidos pela rede através de outros sensores com base em topologias de organização *multi-hop*. Os dados transmitidos podem ainda ser processados e agregados por outros sensores ao longo da rede, durante a transmissão e encaminhamento dos mesmos. Finalmente, a informação gerada pela rede é geralmente recolhida em nós de captura de dados com características mais ou menos especializadas (habitualmente designados por *base stations* ou *sink nodes*) que poderão estar interligados a sistemas que executam aplicações para tratamento e análise de dados. Esta ligação poderá fazer-se com base em qualquer tipo de comunicação (e.g. redes locais com ou sem fios, através da Internet, etc.).

Serão, de seguida, apresentados alguns exemplos de sensores [1] utilizados neste tipo de redes:

- TelosB – microprocessador TI MSP430F1611 de 16 bits a 8 MHz, 10 KBytes de memória RAM, 48 KBytes de memória dedicada a aplicações, 1024 KBytes de memória para armazenamento de dados e duas baterias AA;
- MICA2DOT – microprocessador ATMEL ATmega128L de 8 bits a 4 MHz, 128 KBytes de memória dedicada a aplicações, 512 KBytes de memória para armazenamento de dados e uma bateria do tipo célula-moeda de 3V;
- MICAz – microprocessador ATMEL ATmega128L de 8 bits a 7.37 MHz, 128 Kbytes de memória para aplicações, 512 KBytes de memória para dados e duas baterias AA, estando os seus principais componentes identificados na figura 2.1.

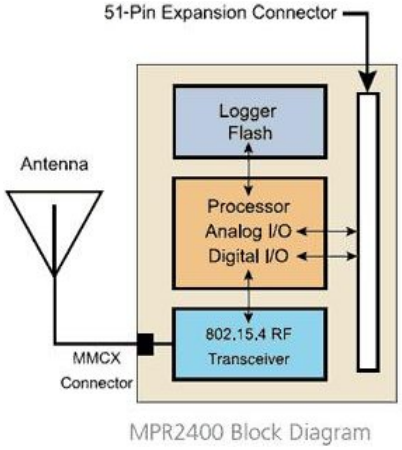


Figura 2.1: Constituição e organização interna de um sensor MICAz

2.2 Normalização IEEE 802.15.4

O standard IEEE 802.15.4 [2] especifica as duas camadas de nível mais baixo do modelo OSI: a camada física (PHY) e a sub-camada de controlo de acesso ao meio (MAC), que foram idealizadas tendo em consideração as baixas taxas de dados enviados/recebidos que caracterizam este tipo de redes, bem como as suas limitações a nível energético. O espaço de operação das redes 802.15.4 está bem delimitado – o raio de acção de um nó dessa rede é da ordem das dezenas de metros – pelo que estas redes são também conhecidas por WPAN (*Wireless Personal Area Network*). Uma rede 802.15.4 pode seguir uma topologia em estrela ou uma topologia ponto-a-ponto (cf. figura 2.2).

Numa topologia em estrela é adoptado um modelo de *master* e *slave*, em que o *master* representa o papel de coordenador da PAN (*Personal Area Network*) e os outros nós comunicam apenas com o coordenador. Para ser definido como coordenador, um nó terá de ser FFD (*Full Function Device*). Os diferentes tipos de dispositivos serão abordados mais à frente, na secção 2.2.2.

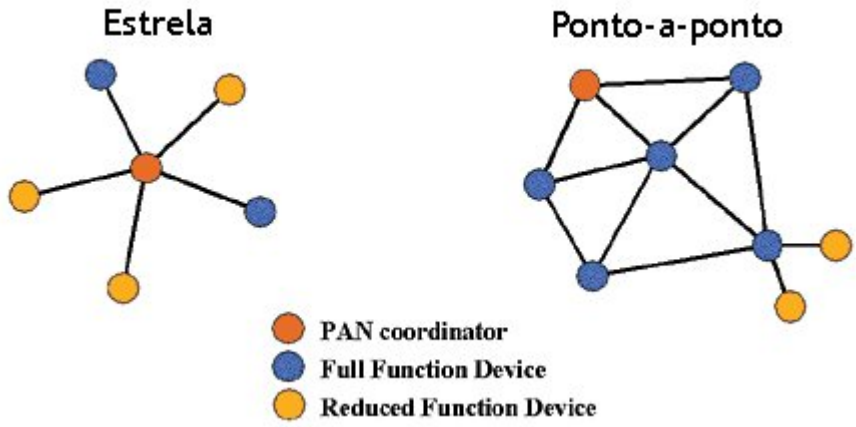


Figura 2.2: Topologias em estrela e ponto-a-ponto

Numa topologia ponto-a-ponto, é permitido a um coordenador comunicar com outros FFDs mesmo que estes não se encontrem no seu raio de acção (i.e. o sinal rádio não é suficientemente forte para chegar até esses dispositivos), utilizando para o efeito FFDs intermédios que se encarregam de estabelecer a comunicação, formando um caminho *multi-hop*. Um dispositivo pertencente a uma rede 802.15.4 poderá ter um endereço IEEE de 64 bits ou um endereço mais curto de 16 bits, o que faz com que uma rede possa ter até 64 000 (2^{16}) dispositivos.

As ligações entre dois dispositivos podem operar em três bandas distintas identificadas na tabela 2.1, que se inserem nas bandas de frequência ISM (*Industrial, Scientific*

and Medical).

Banda	Região	Número de canais	Taxa de transf. de dados
868 MHz	Europa e Japão	1	20 kbps
915 MHz	USA	30	40 kbps
2.4 GHz	Mundial	16	250 kbps

Tabela 2.1: Bandas de frequência utilizadas no IEEE 802.15.4

O standard IEEE 802.15.4 é essencialmente um standard que tenta satisfazer um dos principais requisitos deste tipo de redes (as suas limitações energéticas), tentando assim obter custos de operação mínimos juntamente com uma simplicidade a nível tecnológico, sem que para isso seja necessário sacrificar a flexibilidade ou a usabilidade do sistema.

2.2.1 Camada física

A camada física fornece uma interface entre a camada MAC e o canal físico de rádio. Esta interface disponibiliza dois serviços que são acedidos através de dois SAP (*Service Access Point*): o serviço de dados do nível físico e o serviço de gestão do nível físico, que dão acesso a funções de gestão e mantêm uma base de dados com informações de outras PANs. Deste modo, esta camada gere o emissor/receptor dos sinais rádio e efectua a selecção de canais, para além de fornecer funções relacionadas com a energia e o sinal do dispositivo. A camada física está responsável pelas seguintes tarefas: activação e desactivação do receptor/emissor de rádio; detecção de energia de um canal; identificação da qualidade de um canal; avaliação da disponibilidade de um canal para se iniciar uma transmissão; selecção da frequência do canal; e enviar e receber dados.

2.2.2 Camada MAC

O protocolo de controlo de acesso ao meio, MAC (*Medium Access Control*), especifica um conjunto de técnicas que permite aos nós de uma determinada rede partilharem um meio de comunicação onde existe coordenação para a transmissão de dados. Encontram-se, portanto, bem definidos os procedimentos a executar caso existam colisões de dados e qual o tempo de espera de um participante até nova tentativa de transmissão. Face às características inerentes às redes de sensores sem fios (essencialmente limitações energéticas), os protocolos tradicionais para controlo de acesso ao meio revelam-se, assim, inadequados, sendo necessário introduzir novos protocolos

para este tipo de redes. Tal como a camada física, a camada MAC fornece dois serviços (serviço de dados e o serviço de gestão).

Nas comunicações sem fios são utilizados, tipicamente, protocolos que seguem uma política TDMA (*Time Division Multiple Access*), FDMA (*Frequency Division Multiple Access*) ou CDMA (*Code Division Multiple Access*). O conceito base por trás destas três técnicas consiste em evitar colisões através de um escalonamento que poderá ser temporal, por frequência ou por códigos, que visa atribuir slots a todos os nós de uma forma justa (i.e. todos os nós têm o mesmo "tempo de antena"). Como as comunicações são escalonadas, estes protocolos dizem-se livres de colisões. No entanto, devido à atribuição de slots a todos os nós, poderá dar-se o caso de desperdício de recursos (i.e. foi atribuída uma slot a um nó mas ele não tem dados para transmitir). Além disso, este tipo de técnicas requer geralmente uma coordenação centralizada.

Existe uma outra classe de protocolos que se baseia em contenção, isto é, em vez de se atribuir determinados recursos a cada nó, eles irão competir por um canal partilhado, sendo necessária uma coordenação probabilística para tentar evitar colisões. É o caso do ALOHA e do CSMA (*Carrier Sense Multiple Access*).

Para controlar o acesso ao meio, são suportados dois modos de operação que podem ser seleccionados pelo coordenador da PAN: o modo *beacon-enabled*, onde todas as comunicações são realizadas com a utilização de *superframes*, e o modo sem *beacons*, onde a comunicação é feita sem qualquer sinal dado pelo coordenador.

No primeiro modo (*beacon-enabled*), uma comunicação é iniciada com o envio de um *beacon*, que adquire um papel de descritor da estrutura da *superframe*, permitindo efectuar uma sincronização entre os nós. Neste modo existe um período de actividade no qual poderão ser feitas comunicações e um período de inactividade no qual os nós poderão entrar num estado de poupança de energia. Durante o período de actividade, a *superframe* está dividida noutros dois períodos: num primeiro período, CAP (*Contention Access Period*), os nós competem pelo acesso ao meio com o uso de um *slotted* CSMA/CA, e no período seguinte, CFP (*Contention Free Period*), os nós não necessitam de contenção e é permitido o uso de GTSs (*Guaranteed Time Slots*), seguindo a política de um protocolo TDMA. Depois de efectuada a comunicação, o coordenador poderá confirmar a recepção dos dados através da utilização de um *acknowledge*.

No segundo modo (*non beacon-enabled*), quando um nó deseja transferir dados para o coordenador, ele procede à transmissão dos dados com *unslotted* CSMA/CA, ao qual o coordenador poderá responder com um *acknowledge*.

Um dos grandes desafios dos protocolos MAC passa por manter uma taxa de transferência elevada juntamente com uma elevada eficiência energética. Para tal, é definido

um *duty-cycle* que determina os intervalos de tempo em que os nós se encontram activos ou adormecidos. Assim sendo, existem três abordagens que os protocolos MAC podem seguir: uma abordagem assíncrona, onde os *duty-cycle* dos nós são absolutamente independentes uns dos outros; uma abordagem síncrona, onde os nós transitam para um estado activo ou para um estado adormecido de uma forma coordenada no tempo; e ainda uma abordagem híbrida onde os nós transitam do modo síncrono para o modo assíncrono de forma a tirar vantagens das duas abordagens.

Numa aproximação assíncrona temos como exemplos o B-MAC [32] e o X-MAC [9] (que surgiu como uma melhoria ao B-MAC). Numa aproximação síncrona existem os seguintes protocolos: o S-MAC [42], que através do envio de pacotes SYNC permite que os nós estejam constantemente sincronizados e, dessa forma, possam coordenar os seus *duty-cycle*; o T-MAC [35], que surgiu para colmatar alguns aspectos que não foram considerados no S-MAC; o SCP-MAC [43]; entre outros. Numa abordagem híbrida é possível encontrar protocolos como o Z-MAC [33] e o MH-MAC [6], entre outras variantes.

A camada MAC define, ainda, dois tipos de nós: os RFD (*Reduced Function Devices*) e os FFD (*Full Function Devices*). Os RFDs são dispositivos de complexidade reduzida, enquanto os FFDs são dispositivos que possuem um conjunto completo de serviços MAC. Nesta camada são também definidos três tipos de papéis que um nó pode adoptar: coordenador da PAN, coordenador e participante. O coordenador da PAN desempenha a função de *gateway*, garantindo ligação aos nós de outras redes (existe apenas um destes nós por cada PAN). Os coordenadores realizam tarefas de encaminhamento de dados e desempenham funções de organização da rede. Os participantes limitam-se a comunicar com os coordenadores, que lhes permitem estabelecer comunicações com os outros nós da rede.

A camada MAC está assim responsável pelas seguintes tarefas: gerar *beacons* se o dispositivo for um coordenador; efectuar sincronização através dos *beacons*; suportar alterações à PAN associando e desassociando dispositivos; utilizar um mecanismo para evitar colisões ao aceder ao canal; gerir e manter o mecanismo de GTS; e fornecer uma ligação fiável entre duas entidades.

2.3 Nível rede (encaminhamento)

A terceira camada do modelo OSI (camada de rede) não se encontra especificada no standard 802.15.4 [2], pelo que existe mais do que uma implementação. A mais conhecida é a ZigBee [5] que implementa quatro camadas utilizando como base as camadas definidas em 802.15.4. As camadas definidas na especificação ZigBee são: a camada de

rede, a camada de aplicação, os ZDOs (*ZigBee Device Objects*) e os objectos de aplicação definidos pelo fabricante.

De forma a suportar a comunicação *multi-hop*, vários protocolos de encaminhamento foram propostos. Eles podem ser classificados segundo duas categorias: *table-driven* (orientados para a utilização de tabelas – protocolos proactivos) ou *on-demand* (só actuam se necessário – protocolos reactivos) [39]. Nos protocolos *table-driven*, os nós trocam informação entre eles de forma constante de modo a manter as tabelas de encaminhamento sempre actualizadas independentemente de existirem ou não pedidos de estabelecimento de comunicação. O DSDV (*Destination-sequenced distance-vector*) e o OLSR (*Optimized Link State Routing*) são alguns exemplos de protocolos de encaminhamento *table-driven*, onde cada nó mantém uma tabela de encaminhamento que, para cada entrada, terá um endereço de destino, o próximo *hop*, o número total de *hops* até esse destino e um número de sequência. Os protocolos *on-demand* são mais populares entre as redes *ad-hoc*, pois os nós só trocam informação quando existem comunicações por estabelecer. Este facto faz com que não seja necessário manter informação de encaminhamento de rotas. Quando um nó pretende estabelecer comunicação com outro, ele envia um pedido de rota (*route request* – RREQ) para a rede. Os nós que recebem esse RREQ enviam um *route reply* (RREP) de volta caso conheçam uma rota até ao destino pretendido. Caso não conheçam a rota, limitam-se a enviar o pacote novamente para a rede, encaminhando-o para outros nós. O AODV (*Ad-hoc On-demand Distance Vector*) é um exemplo de um protocolo *on-demand*. Existem também variantes que seguem uma política de *source-routing*, como é o caso do DSR (*Dynamic Source Routing*), onde os nós, ao receberem um RREQ, adicionam o seu endereço a esse pacote e reenviam-no para a rede. Assim, os nós que recebem esse pacote poderão enviar uma resposta conhecendo à partida o caminho que o pacote irá percorrer.

2.4 Organização e operação das RSSF

Uma RSSF forma um sistema distribuído capaz de capturar dados através da actividade autónoma e assíncrona dos sensores, com possível pré-processamento desses dados no nó que os registou. Os dados são depois transmitidos e processados de forma interna (ao nível dos nós intermédios) e enviados para os nós de agregação de dados (*base stations* ou *sink nodes*).

Este tipo de redes pode assim ser visto como um caso particular das redes *ad-hoc* sem fios, utilizando um modelo de encaminhamento *multi-hop* para condições de cobertura geográfica superior ao raio de alcance da transmissão rádio de cada um dos sensores que formam a rede. Contudo, as RSSF apresentam diferenças significativas

face às redes *ad-hoc* sem fios [37]:

- Os nós de uma RSSF apresentam limitações de computação, comunicação e energia;
- Na maior parte dos casos, as RSSF não podem ser estruturadas e planeadas com base no conhecimento antecipado da topologia de cobertura;
- As RSSF possuem em geral um elevado número e densidade de nós;
- Em geral, a operação das RSSF não é supervisionada e os dispositivos têm mais possibilidade de falhas motivadas pelo ambiente em que operam, podendo exibir maiores taxas de erros e falhas de comunicação motivadas por factores externos.

Por norma, uma rede de sensores sem fios é formada por nós sensores comuns e por um ou mais nós de agregação de dados (do tipo *base station* ou *sink node*). O modelo de comunicação consiste assim no encaminhamento multi-hop da informação ao longo dos vários nós da rede até aos nós de agregação de dados. Em determinadas circunstâncias podem tornar-se possíveis outros modelos de comunicação tais como *broadcast* (difusão de informação para todos os nós) ou *multicast* (comunicação orientada de um para muitos normalmente associada a grupos).

É também importante salientar que o custo energético das comunicações pode ser várias ordens de grandeza superior ao custo associado à aquisição de sinal e ao processamento local. Pode, portanto, tornar-se útil optar por agregar e processar a informação dos vários sensores através de computações intermédias ao longo da rede, ao invés de enviar toda a informação (possivelmente redundante) entre cada sensor e a *base-station*, o que iria consumir bastantes mais recursos.

2.4.1 Escala e auto-organização

Um dos principais objectivos das redes de sensores sem fios é permitir que estas sejam utilizadas em ambientes de grande escala (na ordem dos milhares de nós), sendo assim possível cobrir áreas geográficas enormes desde que se verifiquem determinadas propriedades de densidade e distribuição espacial (i.e. o número de nós tem de ser suficiente de forma a cobrir toda a área tendo em conta as limitações existentes ao nível das comunicações). Para aumentar a densidade de uma rede deverá ser apenas necessário adicionar mais nós a essa mesma rede, sem que isso represente um aumento na complexidade da gestão da mesma.

Desta forma, revela-se muito interessante que o processo de descoberta e auto-organização da rede exija pouca ou nenhuma intervenção humana. Ou seja, torna-se

particularmente importante que as RSSF se possam formar e gerir de forma autónoma, com base em determinados critérios de descoberta de nós e auto-organização da rede.

2.4.2 Topologia e flexibilidade

Do ponto de vista da topologia, os nós de uma rede de sensores podem estar interligados de forma directa ou indirecta, podendo as redes ser mais ou menos densamente povoadas de modo a assegurarem conectividade global numa perspectiva de extremo-a-extremo.

A topologia adoptada por uma RSSF pode assumir diversas formas, tais como: estruturas hierárquicas, estruturas em estrela, estruturas centradas em grupos, estruturas em malha, etc. Em alguns casos poderá também considerar-se a utilização de uma estrutura híbrida que combine duas ou mais das estruturas anteriores. A topologia escolhida para uma determinada rede poderá trazer vantagens ou desvantagens, dependendo dos padrões de comunicação entre os nós e dos requisitos particulares da aplicação em causa.

Convém também salientar que a topologia de uma RSSF será sempre fortemente influenciada pelas características que os seus nós constituintes apresentam em termos de mobilidade. Se estivermos perante uma rede dinâmica, onde a posição dos nós varia ao longo do tempo, uma organização do tipo malha será a topologia que mais se adequa (cf. Figura 2.3). Numa topologia deste tipo, todos os nós desempenham as mesmas funções e partilham as mesmas características de hardware. Assim, garante-se um elevado nível de flexibilidade e maximiza-se a área de cobertura da rede, visto que é possível estabelecer uma ligação entre dois nós quaisquer desde que as condições necessárias para se efectuarem as comunicações rádio se verifiquem. Por outro lado, se o número de ligações entre os nós for muito elevado, isso poderá ter um impacto negativo no desempenho energético do sistema.

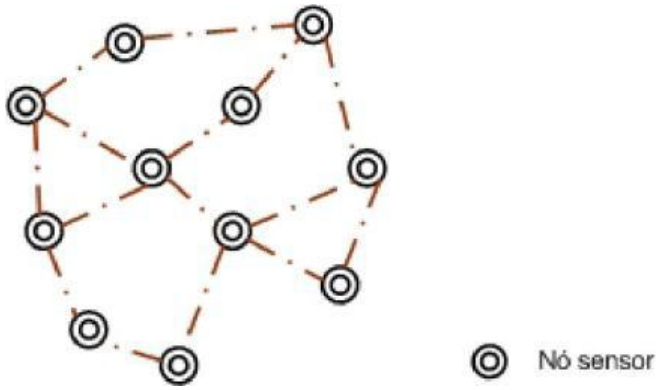


Figura 2.3: Rede organizada segundo uma topologia de malha

No entanto, caso a rede seja estática (i.e. a posição dos nós é fixa), poder-se-á optar, por exemplo, por uma organização em grupo (cf. Figura 2.4). Cada grupo terá assim um nó agregador, responsável por reunir toda a informação gerada pelos nós pertencentes a esse mesmo grupo. Esta aproximação poderá trazer algumas vantagens, como por exemplo:

- o número de ligações entre os nós é menor, o que irá libertar algum espaço na memória dos mesmos, para além de reduzir a quantidade de informação que é necessário trocar entre eles, o que se reflecte num melhor desempenho energético;
- o processo de distribuição e substituição de chaves está facilitado nas redes com esta topologia, pois os nós de cada grupo podem partilhar uma chave simétrica com elevada frequência de refrescamento, de forma a evitar comprometimento caso uma chave seja descoberta. A geração da chave pode ficar a cargo do nó agregador ou pode ser gerada de forma contributiva por todos os elementos desse grupo, garantindo assim um maior nível de segurança e fiabilidade.

No entanto, os nós agregadores têm tendência a ter um período de vida mais curto, dada a sua propensão para realizar mais processamento e transmissão de dados. Para ultrapassar este problema, os nós agregadores poderão ter capacidades diferentes dos outros nós, mais concretamente uma maior capacidade de processamento, de armazenamento e de energia disponível. Outra alternativa seria a rede proceder a reconfigurações elegendo novos nós agregadores quando necessário.

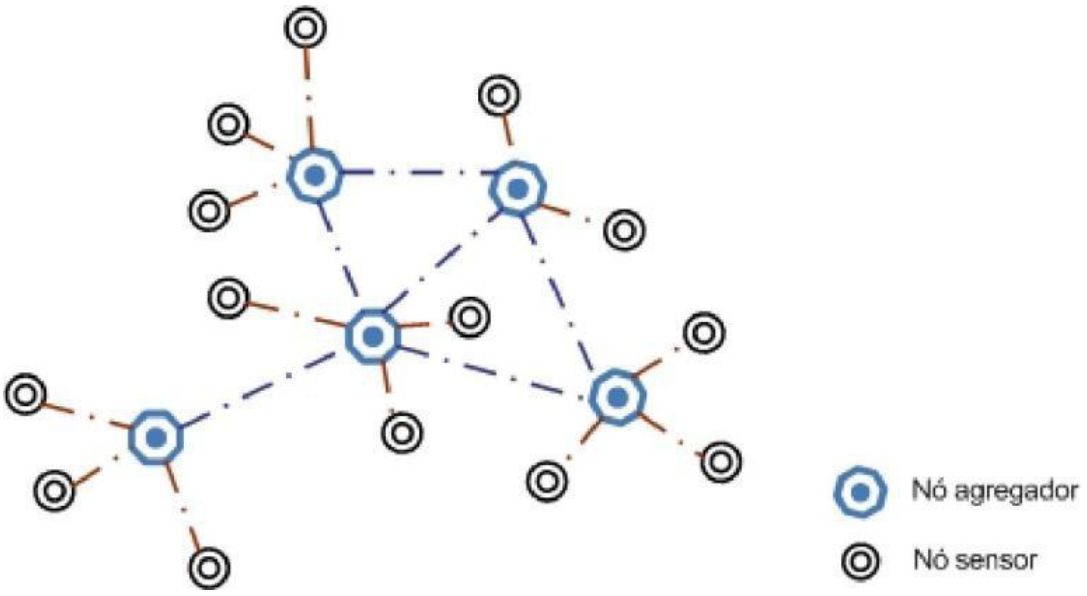


Figura 2.4: Rede organizada segundo uma topologia de grupos

As topologias híbridas tentam conjugar a flexibilidade obtida pelas redes com topologias em malha com o funcionamento estrutural das redes organizadas hierarquicamente. Dependendo da aplicação para a qual a rede irá servir, poderá dar-se o caso em que se consegue obter uma rede dinâmica com uma estrutura leve, permitindo assim uma melhor gestão dos recursos da rede.

2.5 Pilha de serviços para suporte de aplicações em RSSF

As arquitecturas de software para aplicações em RSSF são estruturadas tendo em conta requisitos específicos de aplicações. Não existe, portanto, um conjunto de serviços estruturados a nível *middleware* que possa ser utilizado por diferentes tipos de aplicações, levando a que a pilha de serviços fornecidos seja definida com base nos requisitos próprios de cada aplicação (cf. figura 2.5). No entanto, a estruturação de camadas *middleware* que facilitem o desenvolvimento de aplicações poderá revelar-se um aspecto decisivo no sucesso da tecnologia das RSSF.

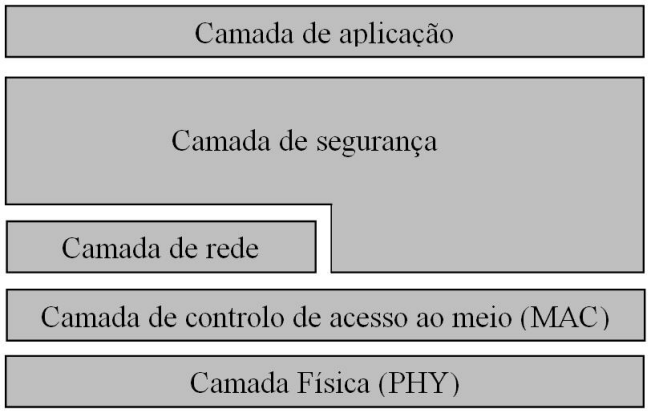


Figura 2.5: Pilha de serviços para RSSF

2.6 Segurança em RSSF

A investigação mais recente na área das redes de sensores mostra que, devido às características das mesmas e atendendo aos requisitos de segurança de algumas das aplicações perspectivadas para estas redes, a segurança é um critério chave.

Devido às características das RSSF, as técnicas, mecanismos e serviços de segurança utilizados vulgarmente nas redes convencionais, ou mesmo no âmbito das redes móveis *ad-hoc* em geral, não podem ser directamente aplicados às redes de sensores. Dadas as características físicas, a natureza das comunicações sem fios e os cenários de

aplicação destas redes, os mecanismos de segurança têm de estar presentes logo a um nível muito baixo na abordagem de uma pilha de protocolos e serviços para arquitecturas de redes de sensores seguras. As redes de sensores são muitas vezes distribuídas em locais inacessíveis e sem vigilância, constituindo um maior risco devido à eventualidade de ataques físicos. Finalmente, as RSSF interagem muito directamente tanto com o meio envolvente como com pessoas, aumentando o seu nível de vulnerabilidade.

2.7 Distribuição e estabelecimento de chaves em RSSF

No enquadramento das anteriores dimensões da segurança em RSSF, o âmbito da presente dissertação foca-se essencialmente na problemática da gestão e distribuição segura de chaves ou de segredos criptográficos. Como inicialmente focado no capítulo 1, o objectivo previsto para a dissertação tem em vista avaliar diversos esquemas de distribuição segura de chaves para RSSF propostos na investigação destas redes, sendo essa avaliação suportada pela concepção e implementação de um ambiente de simulação e emulação para RSSF. O ambiente de simulação a conceber utilizará como base um núcleo de simulações baseado no sistema TOSSIM, ao qual serão acrescentados módulos de simulação que permitirão fazer as medições pretendidas. A análise de um esquema de distribuição e estabelecimento de chaves para RSSF deve abarcar o estudo de cada solução face a diferentes critérios [41]:

- Consumo energético – avaliação do impacto causado por estes mecanismos no consumo energético da rede;
- Latência e tempo de estabilização – qual a latência introduzida pela utilização destes mecanismos;
- Taxas de cobertura – qual o número de nós capaz de comunicar com a rede após o processo de estabelecimento de chaves ter terminado;
- Taxas de entrega de pacotes – qual o nível de fiabilidade de uma rede após as ligações seguras estarem definidas.

Na verdade, os critérios acima referidos possuem uma incidência que só pode ser rigorosa e completamente avaliada com base em análises experimentais que sejam relevantes face a condições reais ou que possam antecipar essas condições reais. A abordagem da presente dissertação privilegiará o estudo experimental numa base de simulação dos critérios anteriores.

3

Trabalho relacionado

3.1 Modelo de adversário

Um modelo de adversário pretende identificar um conjunto de possíveis ataques a que um sistema poderá estar sujeito. Através dos modelos de adversário, pretende-se adoptar uma perspectiva defensiva ao invés de uma perspectiva reactiva. Isto é, em vez de se tentar encontrar falhas num sistema já implementado, pretende-se que na fase de desenho desse sistema seja tida em conta uma determinada tipologia de ataques, levando a que o sistema seja implementado de forma a estar protegido das ameaças identificadas nesse modelo.

À medida que novas tecnologias vão sendo criadas novas vulnerabilidades vão surgindo, pelo que os modelos de adversário definidos anteriormente poderão já não ser suficientes para proteger determinado tipo de sistemas. Desta forma, um sistema poderá ficar vulnerável a um determinado tipo de ataques que não está previsto pelo modelo de adversário definido previamente. Assim, novas tecnologias (e consequentemente novas vulnerabilidades) requerem, por norma, novos modelos de adversário. Apesar de no âmbito desta dissertação não serem considerados ataques às redes formadas pelos protocolos em estudo, neste capítulo serão identificados alguns dos possíveis ataques às redes de sensores sem fios e será também definido um modelo de adversário adequado às características da presente dissertação.

Recordando a organização por camadas utilizada pelas redes 802.15.4, é possível constatar que os ataques podem ocorrer a diversos níveis:

- Ao nível físico, o ataque mais comum consiste num atacante emitir sinais com o objectivo de perturbar quaisquer comunicações que se tentem estabelecer (esta técnica é conhecida como *jamming*). É também necessário considerar um outro tipo de ataques que actuam por intrusão, baseando-se na captura e *cracking* de nós;

- Ao nível MAC, o ataque que mais se verifica consiste em perturbar/impedir o acesso de outros nós ao meio de comunicação, levando a uma quebra no fluxo de informação trocada entre os nós, resultando num DoS (*Denial of Service*);

- Ao nível da rede, tipicamente, um ataque passa por: falsificar, alterar ou reenviar informação de encaminhamento; reencaminhar apenas alguns pacotes; atrair grande parte do tráfego para uma zona onde se encontre um nó que tenha sido corrompido do ponto de vista da segurança e que permita ao atacante manusear toda essa informação que foi atraída até ele da maneira que mais lhe convier; ataques *sybil*, nos quais um nó cuja segurança tenha sido comprometida assume diversas identidades, ignora informação de encaminhamento, injecta pacotes corruptos na rede e reencaminha pacotes por caminhos errados; entre outras variantes;

- Nos níveis superiores ao nível da rede (e.g. nível de aplicação) os ataques podem ser orientados para o processamento interno da rede ou para processos de agregação de dados de acordo com os requisitos das aplicações.

3.1.1 Ataques ao nível MAC

Os ataques ao nível MAC podem ser divididos em duas categorias: os ataques que seguem as regras do standard 802.15.4, independentemente de as seguirem à risca ou parcialmente, e os ataques que não seguem as regras especificadas para as redes 802.15.4 [28].

Apesar de um nó seguir as regras do protocolo MAC, não significa que esse nó não possa fazer ataques à rede. Um nó que cumpra as especificações poderá, por exemplo, enviar um número elevado de pacotes de grande dimensão, reduzindo drasticamente o desempenho da rede. Outro ataque passível de ser feito passa por enviar pacotes para um determinado nó com o objectivo de esgotar os seus recursos energéticos. Um nó malicioso pode também alterar o valor de aMacBattLifeExt para *true*, fingindo correr em modo de bateria de longa duração, provocando alterações no algoritmo CSMA/CA, que irá escolher o valor inicial para o expoente de *backoff* de 2 em vez

de 3, dando maior probabilidade de acesso ao meio a esse nó malicioso. É, também, necessário considerar os nós que utilizem um protocolo MAC modificado. Um exemplo disso é um nó que após uma tentativa de transmissão sem sucesso não aumente o seu expoente de *backoff*. O gerador de números aleatórios pode ser modificado para dar preferência a baixos expoentes o que dará preferência a estes nós maliciosos, dando-lhes acesso à rede num maior número de casos e deixando os nós regulares com um menor acesso ao meio.

Outra situação pode ser identificada pelos nós que não respeitam o protocolo MAC. Um adversário com os recursos apropriados poderá desenvolver hardware dedicado que seja compatível com as especificações 802.15.4 mas que não esteja conforme as mesmas. Os tipos de ataques podem ser dos mais variados e o seu nível de sofisticação pode variar bastante. Um exemplo de um ataque consiste num adversário injectar pacotes na rede com o objectivo de perturbar comunicações, destruindo informações ou obrigando a retransmissões.

3.1.2 Ataques ao encaminhamento (nível rede)

Muitos dos protocolos de encaminhamento assumem que os nós não terão um comportamento incorrecto. Isto permite que nós maliciosos possam atacar a rede e perturbar o encaminhamento de pacotes. Os ataques podem ser classificados como passivos e activos.

Um ataque passivo envolve, geralmente, obter o conteúdo de pacotes com informação de encaminhamento que permita ao atacante descobrir informações que lhe possam ser úteis. Estes ataques são de difícil detecção pois eles não destroem os protocolos de encaminhamento, simplesmente escutam o tráfego que passa na rede (*eavesdropping*).

Um ataque activo tem como objectivo destruir total ou parcialmente o protocolo de encaminhamento, alterar os pacotes de encaminhamento trocados na rede, ganhar o controlo da rede através da injecção de pacotes falsos na rede ou através da captura de pacotes trocados entre outros nós. Estes ataques podem, portanto, ser divididos em duas outras categorias: ataques externos, se forem realizados por dispositivos que não pertencem à rede, ou ataques internos caso o nó malicioso faça parte da rede.

Desta forma, os ataques ao nível de encaminhamento estão organizados em três classes distintas, que serão explicadas de seguida [39].

3.1.2.1 Ataques à descoberta de rotas

Este tipo de ataques tem o objectivo de impedir que nós legítimos estabeleçam rotas, enviando para o efeito falsa informação de encaminhamento. É possível identificar três ataques distintos:

- *Fake Routing Information* – Ataque que consiste em anunciar rotas falsas. Na categoria de protocolos proactivos o objectivo é criar entradas com rotas incorrectas nas tabelas de encaminhamento dos nós. Em protocolos reactivos o ataque pode basear-se em responder a pedidos quando não deve ou alterar o endereço dos pacotes de resposta.
- *Rushing Attacks* – Tipo de ataque que consiste em suprimir um pedido de estabelecimento de rota e enviar um pedido alterado de forma a impedir o estabelecimento de uma rota válida ou de obrigar essa rota a incluir um nó sob controlo do atacante. Este tipo de ataques aplica-se à categoria de protocolos reactivos.
- *RREQ Flood Attacks* – Este tipo de ataques baseia-se na noção de negação de serviço efectuada através da inundação na rede com pacotes de pedido de estabelecimento de rota. Este ataque irá consumir os recursos da rede e poderá também encher por completo as tabelas de encaminhamento dos nós, impedindo o estabelecimento de rotas válidas.

3.1.2.2 Ataques à selecção de rotas

Este tipo de ataques pretende aumentar a hipótese de um nó malicioso ser seleccionado por outros nós para fazer parte das rotas de encaminhamento, com o objectivo de escutar as mensagens que passam nessas rotas ou de desencadear outro tipo de ataques.

- *Hello Flood Attacks* – Em alguns protocolos é necessário os nós anunciarem-se a eles próprios através do envio de mensagens HELLO. Um nó, ao receber esta mensagem, assume sempre que foi enviada por um nó que está a um salto (hop) de distância dele. No entanto, um atacante com acesso a maiores recursos de comunicação (e.g. através de um computador portátil) poderá enviar mensagens HELLO direccionadas para um raio de acção maior, fazendo com que muito provavelmente os nós dentro desse raio de acção o seleccionem para composição de rotas.
- *Sinkhole Attacks* – O objectivo deste tipo de ataques é atrair o tráfego da rede para um dos nós que esteja sob controlo do atacante, tal como acontecia no ataque

anterior (*Hello Flood*). No entanto, enquanto no *Hello Flood* o atacante teria de possuir maior capacidade de comunicação, no *Sinkhole* ele poderá ter os mesmos recursos que os restantes nós na rede, pois o ataque é feito a partir de modificação de mensagens.

- *Wormhole Attacks* – Neste tipo de ataques o atacante possui normalmente dois nós e uma ligação física entre eles independente da rede e à qual só ele tem acesso. Isto permitirá criar um túnel de comunicação entre estes dois nós, o que tipicamente terá uma latência mais baixa do que através da rota normal por *multi-hop*. Isto resultará numa falsa aparência da topologia da rede, dando a ideia de que esses dois nós controlados pelo atacante utilizam uma rota multi-hop melhor do que a existente, levando os nós vizinhos a defini-los como parte integrante das suas rotas.

- *Sybil Attacks* – Num ataque *sybil*, um nó com comportamento malicioso irá anunciar-se na rede com várias identidades diferentes, aumentando a sua probabilidade de ser seleccionado pelos nós vizinhos para fazer parte das suas rotas. Este tipo de ataques é também muito forte contra protocolos tolerantes a falhas, que estabelecem várias rotas para resistir a ataques (essas rotas poderão ser diferentes do ponto de vista do nó legítimo e na realidade passarem todas pelo nó com comportamento malicioso devido às múltiplas identidades utilizadas pelo mesmo).

3.1.2.3 Ataques após o estabelecimento de rotas

Após um nó ter estabelecido uma rota através de um nó com comportamento malicioso, este poderá descartar pacotes ou modificar o conteúdo destes se a informação não estiver cifrada. É possível identificar dois tipos de ataques.

- *Blackhole Attacks* – Este tipo de ataques consiste em ter um nó controlado pelo atacante inserido em algumas rotas da rede e posteriormente descartar os pacotes que passem por ele, interrompendo as comunicações. Poderá, alternativamente, descartar selectivamente alguns pacotes, evitando que os emissores se apercebam de que através da rota em questão os pacotes não estão a ser entregues ao destino.

- *Spam Attacks* – Estes ataques baseiam-se em gerar um número enorme de mensagens inúteis com o único objectivo de consumir os recursos da rede, especialmente a capacidade energética dos sensores que irão reencaminhar mensagens sem qualquer interesse.

3.1.3 Sumário

O modelo de adversário permite caracterizar a tipologia de ataques que se pretende considerar para as RSSF. Para que seja possível obter uma definição realista desse modelo, é necessário ter consciência das ameaças e fragilidades a que este tipo de redes está sujeito. Para tal, é imprescindível fazer uma análise cuidada aos vários níveis da pilha de suporte das redes de sensores e identificar as possibilidades de ataque em cada um desses níveis. Após a obtenção do resultado final dessa análise, será possível definir um modelo de adversário adequado aos objectivos da implementação pretendida, o que será feito na secção seguinte.

3.2 Definição do modelo de adversário

Para caracterizar um adversário podemos recorrer a dois critérios que se revelam essenciais para obter o perfil do atacante: quais os recursos que tem à sua disposição e que tipo de ataques consegue efectuar [41, 21].

Perceber quais são os recursos que um adversário pode utilizar para atacar um sistema tem uma importância particularmente relevante no âmbito das redes de sensores, dadas as características inerentes às mesmas (mais concretamente, as suas limitações a nível energético). Desta forma, podemos fazer uma distinção entre duas classes: *mote* e *laptop*. Na primeira, o adversário possui equipamento semelhante ao que é utilizado na rede, enquanto na segunda tem à sua disposição dispositivos mais potentes do ponto de vista de capacidade de processamento, capacidade energética e capacidade de comunicação. Um adversário que se enquadre nesta segunda classe terá obviamente maior facilidade em atacar a rede (e.g. dada a sua maior capacidade de comunicação, ele poderá produzir um ataque em qualquer nó da rede ou qualquer mensagem).

Quanto ao tipo de ataques existentes, eles também podem ser divididos em duas classes: ataques externos e ataques internos. Entende-se por ataques externos todos os ataques compreendidos no modelo de Dolev-Yao [13], ou seja, ataques que se baseiem no comprometimento de mensagens a qualquer nível, inserindo-se assim na classe de ataques às comunicações. Os ataques internos envolvem rapto ou intrusão de um nó na rede com o objectivo de obter informações críticas como por exemplo chaves ou segredos criptográficos, estando normalmente associados à classe de ataques por intrusão.

Os ataques externos consistem em ataques do tipo *man-in-the-middle*, no qual um atacante poderá ler, modificar, reordenar, forjar, reproduzir ou inserir mensagens em qualquer ponto da rede. Poderá também tentar actuar como um nó legítimo na rede

imitando o comportamento desses nós, porém nunca conseguirá obter chaves criptográficas de nenhum principal da rede. Tal como referido anteriormente, estes tipos de ataques são considerados no modelo de Dolev-Yao [13], estando também definidos em *frameworks* de referência para uniformização de bases de conceitos, de terminologia, de notação, de definição e caracterização de propriedades de segurança, como por exemplo a norma ISO X.800 [20] e sua documentação ou no RFC2828 estabelecido pelo IETF [4] como referência para serviços de segurança na Internet. No caso das RSSF, os ataques externos podem ser, geralmente, protegidos por mecanismos criptográficos e técnicas bem conhecidas, com excepção para os métodos de criptografia assimétrica ou outros protocolos de segurança que se revelem desadequados por precisarem de suporte para comunicação síncrona e transporte fiável orientado à conexão. Também são desadequados todos os mecanismos e serviços de segurança que exijam um elevado custo ou complexidade de processamento, dadas as limitações dos sensores. Por último, mas não menos importante, devem evitar-se as soluções associadas a protocolos de segurança cujas exigências ao nível da complexidade de comunicação obrigue a custos energéticos insuportáveis para as condições de autonomia e limitação energética dos sensores.

Os ataques internos distinguem-se dos externos por, essencialmente, permitirem ao atacante obter chaves ou segredos criptográficos, o que irá reduzir ou tornar nulas as propriedades de segurança estabelecidas pelos mecanismos que dependem desses segredos. Com a produção de um ataque deste género é possível a um atacante replicar um comportamento incorrecto ou malicioso do nó comprometido e produzir ataques que alterem e manipulem a rede. Neste caso, um atacante pode comprometer a rede a partir de hipóteses mais fortes de desencadeamento dos anteriores ataques ao encaminhamento, mesmo que os protocolos de encaminhamento estejam protegidos primariamente por mecanismos associados à defesa de ataques externos.

3.2.1 Sumário

O modelo de adversário a ter em conta compreende essencialmente dois tipos de ataques: os ataques externos e os ataques internos.

Os ataques externos (associados à classe de ataques às comunicações) consistem em ataques do tipo *man-in-the-middle* nos quais um atacante poderá ler, modificar, reordenar, forjar, reproduzir ou inserir mensagens em qualquer ponto da rede. Este tipo de ataques é considerado no modelo Dolev-Yao.

Os ataques internos (associados à classe de ataques por intrusão) consistem em ataques nos quais o adversário poderá ter acesso a chaves ou segredos criptográficos, o que lhe permitirá desencadear ataques de maiores dimensões a partir desta premissa.

Este tipo de ataques enquadra-se no modelo de ataques bizantinos [10].

3.3 Esquemas de pré-distribuição de chaves

Dada a natureza das redes de sensores sem fios e a necessidade de introduzir segurança nas suas comunicações, revela-se necessário recorrer a métodos criptográficos capazes de proteger a rede perante determinado tipo de ataques. Estes métodos criptográficos utilizarão chaves ou segredos que, de alguma forma, terão de ser distribuídos por todos os sensores para que estes possam estabelecer ligações com nós vizinhos numa base segura. Actualmente é possível identificar três esquemas de gestão de chaves distintos: i) esquema de distribuição de chaves; ii) esquema de acordo de chaves; e iii) esquema de pré-distribuição de chaves.

Um olhar mais atento aos dois primeiros esquemas permite rapidamente concluir que não se adequam às RSSF. O primeiro esquema baseia-se numa terceira entidade para distribuir chaves por todos os nós da rede, o que se revela inadequado dada a natureza das comunicações deste tipo de redes. No segundo esquema, os nós combinam chaves entre si apenas quando pretendem estabelecer uma comunicação, o que se torna igualmente inadequado pois tal técnica requer a utilização de criptografia assimétrica, o que não é aplicável nestes dispositivos dadas as suas limitações a nível energético.

Assim sendo, o esquema de pré-distribuição de chaves apresenta-se como o mais apropriado ao oferecer soluções práticas e eficientes para a gestão de chaves em redes de sensores.

Seguidamente serão identificados alguns esquemas de pré-distribuição de chaves que serão alvo de estudos de análises comparativas, o que constitui uma das contribuições da presente dissertação.

3.3.1 Esquema de pré-distribuição aleatória de chaves (R-KPS)

O esquema de pré-distribuição aleatória de chaves, inicialmente proposto por Eschenauer e Gligor [16], é baseado numa partilha probabilística de chaves e num protocolo de descoberta de chaves comuns entre dois nós. Este esquema pode assim ser dividido em três fases distintas: a fase de pré-distribuição de chaves, a fase de descoberta de chaves partilhadas e a fase de estabelecimento de caminhos.

A primeira fase toma lugar ainda antes do *deployment* da rede onde, perante uma *pool* de chaves geradas de forma aleatória, cada um dos nós irá seleccionar aleatoriamente um subconjunto de chaves que irá formar o seu anel de chaves. Esta fase

pretende garantir à partida uma determinada probabilidade de dois nós partilharem pelo menos uma chave.

Nesta fase, o tamanho da *pool* de chaves e o tamanho dos anéis de chaves são determinados a partir da probabilidade pretendida de dois nós partilharem pelo menos uma chave. Essa probabilidade pode ser calculada através da seguinte fórmula:

$$p = 1 - \frac{((P-k)!)^2}{(P-2k)!P!},$$

onde p representa a probabilidade de dois nós partilharem pelo menos uma chave.

Na segunda fase, cada um dos nós irá descobrir quais os vizinhos com quem partilha pelo menos uma chave, de forma a ser possível estabelecer comunicações seguras entre eles.

Finalmente, na terceira fase, se um nó detectar que existe outro nó no seu raio de comunicação com quem não conseguiu emparelhar nenhuma chave, tentará fazer uso das ligações seguras resultantes da segunda fase para estabelecer uma chave entre estes dois nós.

3.3.2 Esquema de pré-distribuição partilhada aleatória Q-composta

O esquema baseado em pré-distribuição partilhada aleatória Q-composta [11] é uma variante do esquema básico apresentado por Eschenauer [16], diferindo apenas no tamanho do anel de chaves, utilizando múltiplas chaves em vez de utilizar apenas uma para estabelecer uma comunicação. Ao aumentar o número de chaves necessárias, a resiliência dos nós face a ataques por captura é também fortalecida.

O seu funcionamento é semelhante ao do R-KPS. Inicialmente, um conjunto de chaves aleatório S é seleccionado de todo o universo de chaves. Deste conjunto S, são seleccionadas m chaves de forma aleatória, para cada nó na rede, que irão formar o seu anel de chaves. Para que seja possível dois nós estabelecerem uma comunicação, eles deverão partilhar entre si q ou mais chaves, sendo que a chave a utilizar nas comunicações será composta pelo *hash* das chaves em comum entre eles (i.e. $K = hash(k_1 \parallel k_2 \parallel k_3 \parallel \ldots \parallel k_n)$).

De forma a calcular o tamanho da *pool* de chaves e o tamanho dos anéis de chaves a utilizar afigura-se necessário, em primeiro lugar, definir com que probabilidade se pretende que dois nós partilhem q ou mais chaves. Tal probabilidade poderá ser calculada a partir da seguinte fórmula:

$$p = 1 - \sum_{i=1}^{q-1} \frac{\binom{|S|}{i}\binom{|S|-i}{2(m-i)}\binom{2(m-i)}{m-i}}{\binom{|S|}{m}^2}.$$

Este esquema oferece uma boa resiliência contra ataques por captura de nós quando o número de nós capturados é pequeno. Se este número for elevado, este esquema tende a revelar grande parte da rede. Desta forma, aumentar o valor de q chaves necessárias para estabelecer uma comunicação dificulta ataques por captura de nós se o número desses nós for pequeno. Por outro lado, a rede fica mais vulnerável a ataques por captura em grande escala. Isto poderá ser desejável, visto que os ataques de captura de nós em pequena escala são mais fáceis de efectuar e de difícil detecção quando comparados com os ataques em larga escala.

3.3.3 Pré-distribuição com noção de localização

Em determinado tipo de aplicações é possível conhecer *a priori* a localização dos sensores, o que poderá ser útil para o processo de pré-distribuição de chaves. Nas RSSF, o objectivo é estabelecer comunicações com os nós vizinhos e se, para cada nó, se tiver conhecimento prévio de quais os seus vizinhos prováveis, o processo de pré-distribuição de chaves fica facilitado.

O modelo de Liu e Ning [24] utiliza esta noção e propõe dois novos R-KPS: *closest* R-KPS e *location-based* R-KPS.

O *closest* R-KPS pré-distribui pares de chaves para os pares de sensores nos quais um sensor apareça, com alta probabilidade, no raio de comunicação do outro (e vice-versa). Para tal, na fase de pré-distribuição de chaves, o servidor responsável pela atribuição de chaves irá basear-se na posição estimada dos sensores para fazer essa distribuição. Assim, para cada sensor a, será obtido o conjunto de sensores S, composto pelos sensores que tenham uma alta probabilidade de ficar numa localização perto de a (i.e. dentro do seu raio de comunicação). Para cada sensor b em S, será gerada uma chave $K_{a,b}$, que será enviada ao par de nós em questão. As fases de descoberta de chaves partilhadas e de estabelecimento de caminhos são similares às do R-KPS.

Quanto ao *location-based* R-KPS, ele baseia-se, tal como o *closest* R-KPS, na informação da localização dos sensores, combinando esta noção com a utilização de polinómios de duas variáveis. Para tal, a zona onde a rede será implementada é dividida em sectores, denominados por células. A cada uma das células será associado um polinómio de duas variáveis a partir do qual será possível determinar a chave gerada para essa célula. Posteriormente, para cada nó, ser-lhe-á atribuído um conjunto de polinómios correspondente à célula referente à sua localização estimada após a fase de criação da rede e a outros quatro polinómios referentes às quatro células circundantes, para que seja possível estabelecer comunicações com os nós das células vizinhas.

3.3.4 Esquemas de pré-distribuição de chaves com chaveiros estruturados

Os esquemas de pré-distribuição de chaves com base em chaveiros estruturados, essencialmente usados no estabelecimento de chaves par-a-par, oferecem boas garantias de segurança face a ataques por captura de nós. É possível fazer uma distinção entre esquemas deterministas e esquemas não deterministas.

Nos esquemas deterministas, cada nó a irá armazenar informação pública (P_a) e informação privada (S_a). Durante a fase de descoberta de chaves, os nós trocarão entre si as suas informações públicas de forma a poder calcular a chave a utilizar nas comunicações entre ambos, o que é conseguido através de uma função f, tal que $f(P_\mathrm{b}, S_\mathrm{a}) = f(P_\mathrm{a}, S_\mathrm{b})$, tendo assim ambos os nós gerado uma chave igual.

Os esquemas não deterministas consistem em combinar o esquema básico apresentado por Eschenauer com os esquemas deterministas apresentados no parágrafo anterior. Desta forma, o servidor de distribuição de chaves gera aleatoriamente m espaços de chaves em que cada um deles tem associada informação privada. Para cada sensor será atribuído um conjunto de k espaços de nomes de entre os m espaços de nomes gerados. Assim, se dois nós tiverem pelo menos um espaço de nomes em comum, eles conseguirão calcular uma chave secreta utilizando para o efeito o esquema determinista definido nessa rede.

Seguidamente serão apresentados três modelos de pré-distribuição de chaves com base em chaveiros estruturados (dois deterministas e um não determinista).

3.3.4.1 Modelo determinista de Blom

O modelo proposto por Blom [7] permite que qualquer par de nós da rede consiga calcular uma chave secreta. Este mecanismo diz-se λ-seguro, ou seja, na eventualidade de existirem até λ nós comprometidos, não será possível deduzir nenhuma chave dos nós não comprometidos.

O funcionamento deste modelo baseia-se em ter o servidor, na fase de pré-distribuição de chaves, a gerar uma matriz G que será de domínio público (i.e. todos os nós poderão ter acesso à sua informação). De seguida, será gerada de forma aleatória uma matriz simétrica D que será utilizada para gerar a matriz $A = (G \cdot D)^T$. Posteriormente, será possível calcular uma matriz simétrica $K = A \cdot G$. Dado a matriz ser simétrica, $K_{ij} = K_{ji}$, sendo este valor a chave calculada para uso nas comunicações entre os nós i e j. Para que ambos os nós tenham acesso a essa chave, a fase de pré-distribuição consiste em, para cada nó k, introduzir a k^a linha da matriz A e a k^a coluna da matriz G. Desta forma, quando dois nós i e j pretenderem calcular a chave par-a-par, bastará que cada

um envie a sua respectiva coluna de G ao outro nó, o que em conjunto com a sua linha privada da matriz A permitirá obter a chave K_{ij} comum entre ambos. É de salientar que a matriz D terá de ser mantida em segredo, caso contrário as chaves de toda a rede estariam comprometidas.

3.3.4.2 Modelo determinista de Blundo

O modelo proposto por Blundo [8] baseia-se na utilização de polinómios. Para tal, o servidor gera de forma aleatória um polinómio de duas variáveis que garanta uma propriedade de simetria (i.e. $f(x,y) = f(y,x)$). Assume-se que cada sensor tem um identificador único. Assim, de seguida, para cada sensor i, o servidor calcula o polinómio correspondente a $f(i,y)$ e associa-o a i. Desta forma, se o nó i pretender saber a chave a utilizar para o nó j basta calcular $f(i,j)$. Da mesma forma, um nó j que conheça o polinómio $f(j,x)$ conseguirá determinar $f(j,i)$, o que devido às propriedades de simetria do polinómio, será igual a $f(i,j)$, tendo assim ambos os nós chegado a uma chave comum. Desta forma, este modelo permite estabelecer chaves entre pares de nós sem que para isso seja necessário efectuar comunicações adicionais (bastará apenas conhecer o ID do nó com o qual se deseja contactar).

O modelo de Blundo apresenta boas propriedades de segurança, no entanto o custo de armazenar um polinómio num sensor cresce exponencialmente com o tamanho da rede, tornando impossível a utilização deste modelo em redes com sensores de fraca capacidade de armazenamento.

3.3.4.3 Modelo não determinista de Du

De forma a obter melhor resiliência face a ataques por captura de nós, Du [14] propôs um esquema não determinista de pré-distribuição de chaves que melhora determinadas propriedades de segurança do modelo proposto por Blom. A ideia centra-se em utilizar múltiplos espaços de chaves, resultando numa combinação do R-KPS proposto por Eschenauer com o modelo de Blom. Poder-se-ão, assim, identificar duas fases: pré-distribuição e acordo de chaves.

Na primeira fase, à semelhança do modelo de Blom, gera-se uma matriz G associando a cada nó i a i^{a} coluna de G. De seguida, serão geradas w matrizes simétricas $D1$, $D2$, ..., Dw. Cada tuplo constituído por $Si = (Di, G), i = 1, \ldots, w$, forma um espaço de chaves e $A_i = (D_i \cdot G)^T$. Finalmente, serão seleccionados para cada nó, de entre os w espaços de chaves criados, t espaços de chaves. Para cada espaço de chaves seleccionado para um nó j será armazenado nesse nó a j^{a} linha de A_i. De acordo com o modelo de Blom, dois nós conseguirão assim calcular uma chave secreta se ambos tiverem um espaço de nomes em comum.

Na segunda fase, cada nó irá enviar uma mensagem *broadcast* contendo o seu identificador, os índices dos espaços de chaves que lhe foram atribuídos e a coluna da matriz G correspondente a esse nó. Um nó que receba essa mensagem, poderá comparar os seus índices dos espaços de chaves com os índices que recebeu e se houver pelo menos um em comum, será possível calcular uma chave comum entre esses dois nós. Depois de todos os nós terem calculado as chaves a utilizar com os seus vizinhos, daí resultará um grafo $G(V,E)$, onde V é o conjunto de todos os sensores pertencentes à rede e E é o conjunto de arcos existentes na rede. Para existir um arco entre dois nós estes terão de ter em comum pelo menos um espaço de chaves e terão de estar no raio de comunicação um do outro. Desta forma, no modelo de Du, é possível a quaisquer dois nós estabelecerem uma comunicação segura entre eles mesmo que não partilhem um espaço de chaves. Para tal, terão de utilizar as ligações seguras já existentes para formar um caminho entre eles a partir do qual um dos nós enviará uma chave secreta que após chegar ao destino passará a ser utilizada entre ambos nas suas comunicações.

3.3.5 Esquemas de distribuição aleatória para cobertura com topologia em *cluster* (modelo *SecLEACH*)

As redes de sensores, para além de uma topologia plana utilizada frequentemente como topologia base nos esquemas apresentados nas secções anteriores, podem também utilizar uma topologia hierárquica orientada para grupos. A comunicação nas redes com topologias hierárquicas baseia-se na eleição de um líder de grupo *cluster head* (CH), tipicamente eleito de um modo probabilístico, que fica encarregue da tarefa de encaminhar pacotes para dentro e para fora do grupo, bem como da tarefa de agregação de dados. Estes nós tornam-se assim mais susceptíveis a ataques por desempenharem funções de importância relevante no funcionamento da rede.

A formação de grupos passa pela disseminação, por parte do líder do grupo, do seu estatuto. Os nós que interceptarem essa informação poderão juntar-se a esse grupo ou a outro de que tenham conhecimento, com base numa determinada métrica (e.g. força do sinal, indicando qual o grupo que se encontra mais próximo).

Como o cargo de CH é muito exigente do ponto de vista energético, existe um mecanismo de reeleição de líderes de grupo com o objectivo de distribuir essa carga pelos diferentes nós da rede. O protocolo LEACH [30] é um protocolo de encaminhamento orientado para redes de topologia hierárquica que implementa as características descritas anteriormente.

O modelo SecLEACH consiste em introduzir propriedades de autenticidade, integridade, confidencialidade e frescura de mensagens ao protocolo LEACH [30], através da utilização do esquema básico de pré-distribuição de chaves proposto por Eschenauer

[16]. O seu funcionamento consiste em gerar um anel de chaves e respectivos identificadores, associando a cada nó m chaves. Cada nó terá um identificador (i.e. o seu endereço) que será então utilizado como semente para um gerador pseudoaleatório de números, gerando uma sequência de m números que representa o conjunto de identificadores das chaves existentes no anel de chaves desse nó. Para além disso, será associada, para cada nó, uma chave par-a-par com a estação base (*base station*).

3.3.6 Sumário

O estudo de esquemas e protocolos de distribuição e estabelecimento de chaves abarca normalmente análises de carácter teórico, não tendo em conta certos aspectos que se poderão revelar decisivos: factores do meio ambiente; impacto do funcionamento dos protocolos em relação às condições de energia da rede; o impacto de uma reconfiguração ou rejuvenescimento da rede; e, finalmente, o impacto da eficácia destes protocolos quando aplicados a redes de diferentes topologias (condições de adaptabilidade a organizações planas ou hierárquicas).

Assim, torna-se necessário complementar essas análises teóricas com uma análise experimental que se aproxime mais das diferentes premissas de funcionamento real da rede, o que se conseguirá através de uma aproximação por simulação e emulação da mesma. Para tal, na abordagem do estudo de protocolos de distribuição e estabelecimento de chaves com base em análise experimental por simulação, devem avaliar-se os critérios de: consumo energético, taxa de cobertura, taxa de fiabilidade e tempo de estabilização, através de uma avaliação sistemática "vis-à-vis" dos esquemas apresentados neste capítulo.

Atendendo à necessidade de dotar as RSSF de características de resiliência face a eventuais ataques por intrusão, os mecanismos de estabelecimento de chaves devem abarcar a possibilidade de suportarem reorganizações topológicas da rede e refrescamento dinâmico de chaves e segredos criptográficos, que poderão, por exemplo, ser alcançados mediante a adição de novos nós. Este aspecto nem sempre é tratado na proposta e formulação teórica dos esquemas de distribuição de chaves. Nos casos em que este tipo de mecanismos é apresentado, nem sempre se aborda a problemática do seu impacto na operação real da rede, pelo que o estudo e observação das várias soluções com base em análises experimentais e simulação do funcionamento real da rede se revelam uma direcção de trabalho relevante.

3.4 Ambientes de simulação/emulação

Para atingir os objectivos previstos para a presente dissertação será necessário recorrer a um ambiente de simulação/emulação que possibilite efectuar uma análise experimental dos vários protocolos de gestão e distribuição de chaves. Esse ambiente de simulação terá que possuir determinadas características que se revelarão essenciais para uma análise fiável desses mesmos protocolos. Para tal, serão analisados de seguida três simuladores distintos: Freemote, JProwler e TOSSIM.

3.4.1 Freemote

O Freemote [26] é um simulador/emulador implementado em Java que se foca em credibilizar o comportamento dos nós emulados, ao possibilitar a interacção destes com outros nós reais. Esta ferramenta suporta não só *motes* Java que são baseados em JVMs[1] optimizadas (e.g. Squawk, Sentilla Point) como também outros tipos de plataformas (e.g. Java cards, SunSPOT).

Esta aplicação divide a arquitectura de um *mote* em três camadas independentes: *Application*, *Routing* e *Data Link and Physical*. O código utilizado nas camadas de *Application* e *Routing* corre tanto em nós reais como em nós emulados, sem ser necessária qualquer adaptação. A camada de *Data Link and Physical* dos nós reais é simulada nos nós emulados. Os nós reais podem ser representados por qualquer dispositivo baseado na interface de comunicação standard IEEE802.15.4 (e.g. MICAz, JMotes, Tmote Sky). Quanto aos nós emulados, estes consistem em *threads* independentes com a possibilidade de estarem distribuídos por vários computadores diferentes, ligados em rede. Assim, podem ser configurados independentemente, o que permite que cada um deles assuma um papel distinto (e.g. *aggregator*, *bridge*, *execution*). A capacidade de emular nós em várias máquinas ligadas em rede possibilita a criação de redes de larga escala (i.e. 10 000 nós) que poderá ser simultaneamente constituída por nós emulados e nós reais.

O Freemote foi concebido com uma orientação virada mais para a análise do comportamento de rede do que para a análise da performance de aplicações com base na sua duração (i.e. quanto "custa" energeticamente uma determinada aplicação), pelo que não disponibiliza qualquer modelo de energia. O modelo de rádio disponível é de baixo nível e foi implementado de uma forma muito simples. Assim, é possível haver dois nós a enviarem mensagens simultaneamente sem que isso gere um erro (i.e. não existe modelo de colisões). Para além disso também não simula o atraso das propagações das ondas rádio.

[1] JVM - Java Virtual Machines

3.4.2 JProwler

O JProwler, desenvolvido no ISIS[2], é um simulador probabilístico implementado em Java e baseado em eventos discretos. Este ambiente permite prototipar, verificar e analisar protocolos de comunicação de redes de sensores *wireless*. O simulador contempla suporte de modelação para um sensor do tipo *MicaMote* e *Mica2*, actuando de forma estática ou em condições de mobilidade. São suportados dois modelos de rádio: *Gaussian Model* (para nós estáticos) e *Rayleigh Model* (mais complexo, para nós móveis), e um protocolo MAC para *MICA2*, sem *acknowledgment*. Tal facto permite testar condições experimentais de simulação de comunicação (propagação e recepção de ondas rádio) com base no modelo de colisões. Este ambiente é escalável, na medida em que permite simular redes com um número arbitrário de sensores que executam um determinado tipo de aplicação. Possibilita também alterações dinâmicas na topologia da rede.

3.4.3 TOSSIM

O TOSSIM, desenvolvido pela Universidade de Berkeley, foi idealizado para simular o funcionamento de sensores equipados com TinyOS. Tem como principal vantagem a possibilidade de utilizar no simulador o código que foi desenvolvido para os sensores. Esse código é escrito em nesC [18], um dialecto da linguagem de programação C.

Esta ferramenta é baseada em eventos discretos e consegue emular um número limitado de tipos de hardware (e.g. conversores A/D, sensores). O seu funcionamento consiste em substituir componentes por implementações de simulações e, durante a sua execução, vão sendo removidos eventos da pilha de eventos (organizada por ordem cronológica) que são de seguida executados. Esses eventos podem pertencer a vários níveis (i.e. podem ser interrupções de *hardware* - baixo nível - ou podem ser recepções de pacotes - alto nível). A interacção com este simulador é feita através de um interpretador que poderá também ser utilizado para efectuar processos de *debug* a uma determinada aplicação. Essa interacção é feita com recurso a duas linguagens de programação: Python e C++. Através deste interpretador é possível definir o número de nós que compõem a rede, os ganhos das ligações entre eles e controlar o avanço da simulação com uma precisão ao nível dos eventos.

Para simular as comunicações, este ambiente de simulação utiliza um modelo de rádio baseado em *signal-to-noise ratio*. Este modelo permite simular colisões e interferências externas. Para tal, é definido para cada um dos nós quais os ganhos das ligações entre esse próprio nó e os outros nós da rede.

[2]ISIS - Institute for Software Integrated Systems

Não existe qualquer modelo de consumo energético disponibilizado por esta ferramenta, no entanto existe uma extensão PowerTOSSIM-Z que acrescenta a possibilidade de realizar estimativas relativas à energia dispendida por uma determinada aplicação.

Todavia, existem dois aspectos menos bons que convém salientar. O primeiro é que todos os nós têm as mesmas características físicas, pelo que é impossível efectuar eleições de nós com base nessas mesmas características. O segundo, relacionado com o modelo de rádio, prende-se com o facto de que o ganho associado a uma ligação entre dois nós é constante ao longo da simulação, mas numa rede real esse dado variaria com o tempo (i.e. devido à humidade, temperatura).

3.4.4 Selecção do simulador base

Tendo em conta o teor do trabalho e a temática que se pretende abordar, pensa-se que o simulador/emulador mais indicado seja o TOSSIM. Apesar de não ser possível haver diferentes nós a desempenharem tarefas distintas, este ambiente suporta um modelo de energia (PowerTOSSIM-Z) e suporta também um modelo de rádio um pouco mais elaborado do que o ambiente escolhido para segunda opção (Freemote), já que permite introduzir um modelo de colisões baseado em *signal-to-noise ratio*. Estes modelos permitem que sejam tidos em consideração determinados aspectos que se revelam essenciais para o desenho de protocolos de distribuição de chaves. Para além disso, apresenta ser também um simulador escalável, característica de relevada importância para o estudo em causa. Na tabela 3.1 é feita uma comparação entre os 3 simuladores analisados.

	Freemote	**JProwler**	**TOSSIM**
Linguagem	Java	Java	Python/C++/nesC
Simulação/Emulação	Ambos	Simulação	Ambos
Fiabilidade	Baixa	Alta	Alta
Escalabilidade	Alta	Alta	Média
Baseado em eventos	Sim	Sim	Sim
Modelos de rádio	Simples	*Gaussian* e *Rayleigh*	*Signal-to-noise ratio*
Heterogeneidade de nós	Sim	Sim	Não

Tabela 3.1: Grelha de critérios de comparação entre os vários simuladores

3.4.5 Sumário

A utilização de um ambiente de simulação/emulação é da mais extrema importância para a concretização dos objectivos desta dissertação, que visa a realização de uma

análise e estudo de vários esquemas de distribuição de chaves com base em avaliações experimentais. Para o efeito, foram considerados três ambientes de simulação distintos e após uma análise cuidada de cada um deles, foi feita uma análise comparativa entre os mesmos de modo a distinguir aquele que mais se enquadra com os propósitos desta dissertação. O TOSSIM revelou-se o ambiente mais adequado ao teor desta dissertação pelos seguintes aspectos:

- Possibilita a realização de simulações oferecendo suporte para emulação aproximada da operação real de sensores TinyOS;
- Utiliza um modelo de rádio que permite simular comunicações com um comportamento muito próximo do real;
- Tem disponível um módulo adicional que permite controlar o consumo energético;

3.5 Sumário

Neste capítulo pretendeu-se fazer uma apresentação do estado de arte, tentando cobrir os aspectos de trabalho relacionado mais directamente associados aos objectivos da dissertação.

Inicialmente foi feita uma apresentação de tipologias de ataques que pudessem ser utilizadas para obter uma definição válida de um modelo de adversário. Para tal, foram identificados os ataques presentes em cada um dos níveis da pilha de serviços para suporte de aplicações em RSSF.

Com base nestas informações, no capítulo seguinte procedeu-se à definição do modelo de adversário, que se baseia na tipologia de ataques inspirados na Framework X.800 [20], complementada pela visão de ataques à disseminação e encaminhamento de dados em [39].

Na terceira secção foram apresentados os vários esquemas de pré-distribuição de chaves que serão objecto de análise na presente dissertação. No estudo desses esquemas será feita uma avaliação sistemática entre eles com base numa análise experimental por simulação com o objectivo de avaliar os critérios de consumo energético, tempo de estabilização, taxa de cobertura e taxa de fiabilidade.

Na quarta e última secção foi feita uma análise comparativa de ambientes de simulação ou de emulação para redes de sensores, cujos serviços base pudessem ser utilizados para alavancar um núcleo de simulação e emulação. O simulador TOSSIM foi classificado como o ambiente mais adequado face aos objectivos da dissertação, por oferecer suporte para emulação aproximada da operação real de sensores TinyOS,

por se encontrar já implementado um módulo de gestão de consumo energético, entre outras razões.

4

Arquitectura

Uma das contribuições da presente dissertação consiste em conceber um modelo que sirva como base de trabalho para conseguir alcançar as restantes contribuições esperadas. Esse modelo deverá integrar um conjunto de funcionalidades que permitam medir e avaliar o desempenho de diferentes esquemas de pré-distribuição de chaves em RSSF nas dimensões e critérios anteriormente estipulados.

4.1 Modelo

Tal como referido no capítulo 1, o modelo a implementar pode ser dividido em três grandes categorias/secções:

- Um núcleo de simulação que compreenda o simulador/emulador propriamente dito (TOSSIM), uma camada que ofereça suporte à comunicação segura, um módulo de cálculo de consumo energético, um módulo de visualização e configuração de topologias de redes e um módulo de gestão de simulações que permita definir determinados parâmetros (número de nós da rede a simular, topologia, esquema de pré-distribuição de chaves a utilizar, etc.);
- Uma colecção de módulos que compõem os diferentes esquemas de pré-distribuição de chaves ou segredos criptográficos a estudar, mais concretamente:
  - esquema baseado em cobertura aleatória a partir de pré-distribuição de chaves (Gligor e Eschenauer) [16];

  - esquema baseado em pré-distribuição partilhada aleatória Q-composta (Chan, Perrig e Song) [11];
  - esquema aleatório para arquitecturas em *cluster* (SecLEACH) [30].

- Um conjunto de módulos adicionais que permitam inferir indicadores sobre as prestações efectuadas pela simulação dos esquemas identificados no ponto anterior, face aos seguintes critérios:

  - Avaliação do custo energético associado ao processo de auto-organização e cobertura da rede a partir da distribuição e estabelecimento de chaves par-a-par;
  - Avaliação de critérios de latência associada ao processo de auto-organização através do estabelecimento de chaves par-a-par;
  - Avaliação de critérios de escala e cobertura da rede;
  - Avaliação de taxas de entrega de pacotes de dados.

O modelo idealizado pode assim ser sintetizado através da arquitectura identificada na figura 4.1. O objectivo será, a partir deste modelo, extrair indicadores que avaliem o desmpenho dos vários esquemas de pré-distribuição segura de chaves, o que permitirá posteriormente fazer uma comparação justa entre esses diferentes esquemas.

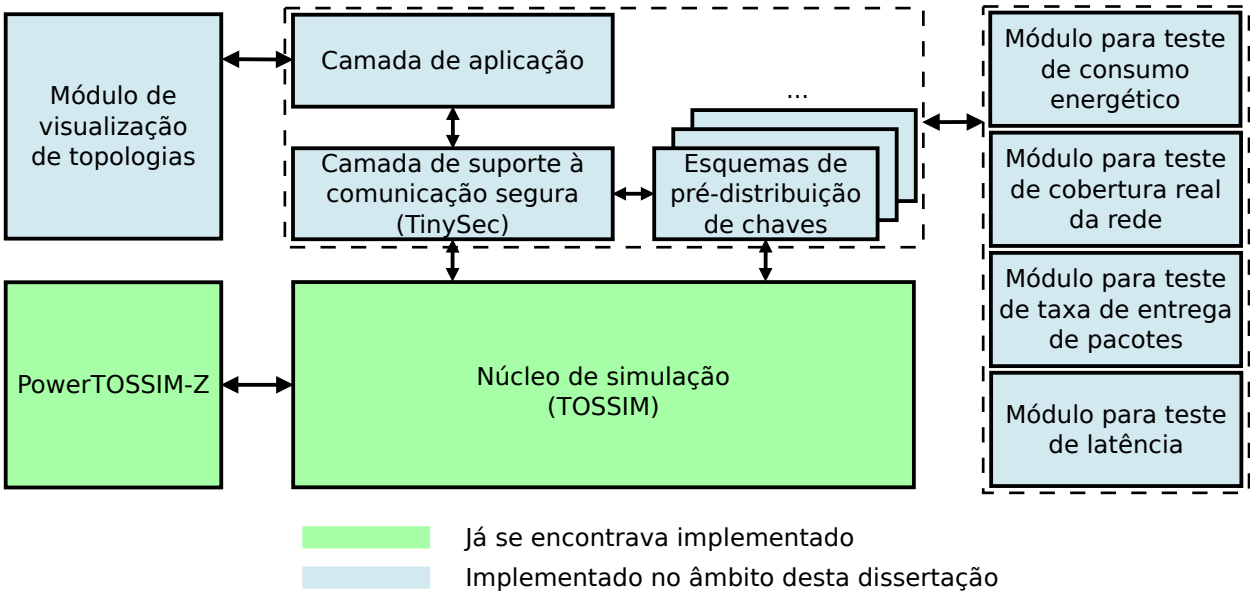


Figura 4.1: Arquitectura da plataforma base

4.1.1 TOSSIM

O TOSSIM, tal como apresentado na secção 3.4.3, é um simulador baseado em eventos, cujo funcionamento se baseia na inserção de eventos numa fila e posterior processamento desses mesmos eventos por ordem cronológica. Este simulador explora o modelo hierárquico do TinyOS substituindo componentes de hardware de baixo nível por emulações desses mesmos componentes, o que permite obter um nível de fiabilidade bastante elevado. Assim sendo, as características que mais realçam este simulador são: i) a sua fiabilidade; ii) a utilização de um modelo de rádio baseado em *signal-to-noise ratio* que permite obter um comportamento muito próximo do real; e iii) a possibilidade de realizar simulações utilizando as mesmas implementações que seriam utilizadas em sensores reais. Estas características permitem concluir que o TOSSIM se trata de um simulador bastante realista, o que aumenta a credibilidade dos resultados obtidos através das simulações, podendo utilizá-los para efectuar comparações de desempenho a vários níveis entre diversos esquemas de pré-distribuição de chaves.

4.1.2 PowerTOSSIM-Z

PowerTOSSIM-Z [31] é uma extensão que permite modelar o consumo energético de aplicações em RSSF. Para determinar o impacto energético causado por essas aplicações é necessário monitorar o comportamento dos componentes de um sensor (e.g. microcontrolador, chip de rádio). Para tal, determinado tipo de acontecimentos (e.g. enviar uma mensagem, receber uma mensagem, ligar LEDs) são capturados num registo que será posteriormente processado com base nos valores extraídos do modelo de energia dos sensores MICAz, obtendo o custo energético imposto por esse conjunto de actividades capturadas.

Para modelar o comportamento de uma bateria é necessário integrar determinadas características que permitam obter um modelo não-linear que se aproxime mais do comportamento real da mesma:

- A voltagem de uma célula diminui gradualmente com o tempo (esta relação é não-linear). Eventualmente, a bateria há-de chegar a um ponto em que a voltagem das suas células atinge um nível demasiado baixo para que a bateria seja usada e a carga restante não poderá ser utilizada;

- Existe um fenómeno conhecido por efeito de taxa de capacidade (*Rate Capacity Effect*) que demonstra que a capacidade de uma bateria diminui com o aumento da corrente de descarga (relação não-linear);

- No estado inactivo, as baterias recuperam parte da sua energia (conhecido como efeito de capacidade de recuperação (*Recovery Capacity Effect*)), que à semelhança das características apresentadas anteriormente também revela propriedades de não linearidade.

Todos estes conceitos estão convenientemente integrados no PowerTOSSIM-Z garantindo que o modelo energético utilizado irá produzir resultados o mais realistas possível. De relembrar que este módulo permite capturar apenas o consumo energético ao nível do núcleo de simulação no seu formato original, excluindo os custos impostos pela utilização da camada de suporte à comunicação segura introduzida no sistema.

Os custos considerados pelo PowerTOSSIM-Z estão indicados na tabela 4.1.

Acção	**Valor de corrente (mA)**
CPU Activo	8.93
CPU Idle	4.93
CPU ADC Noise Reduction	0.32
CPU Power down	0.0003
CPU Power save	0.009
CPU Standby	1
CPU Extended Standby	0.25
Transmissão de uma mensagem	17.4
Recepção de uma mensagem	19.7
Radio Idle	0.02
Radio Sleep	0.001
LED	2.2

Tabela 4.1: Tabela de custos considerados pelo PowerTOSSIM-Z

4.1.3 Módulo de visualização de topologias

O objectivo desta extensão consiste em fornecer uma interface gráfica que tem como finalidade permitir a visualização de topologias de redes. Através desta ferramenta será possível visualizar a disposição dos nós numa área previamente definida. Será também possível visualizar as ligações seguras estabelecidas entre os nós após a simulação de um esquema de pré-distribuição de chaves.

Para tal, foi implementada em Java uma ferramenta que carrega a topologia e a dimensão da área simulada a partir dum ficheiro de texto. Depois de carregada a topologia, ela é apresentada tal como ilustrado na figura 4.2.

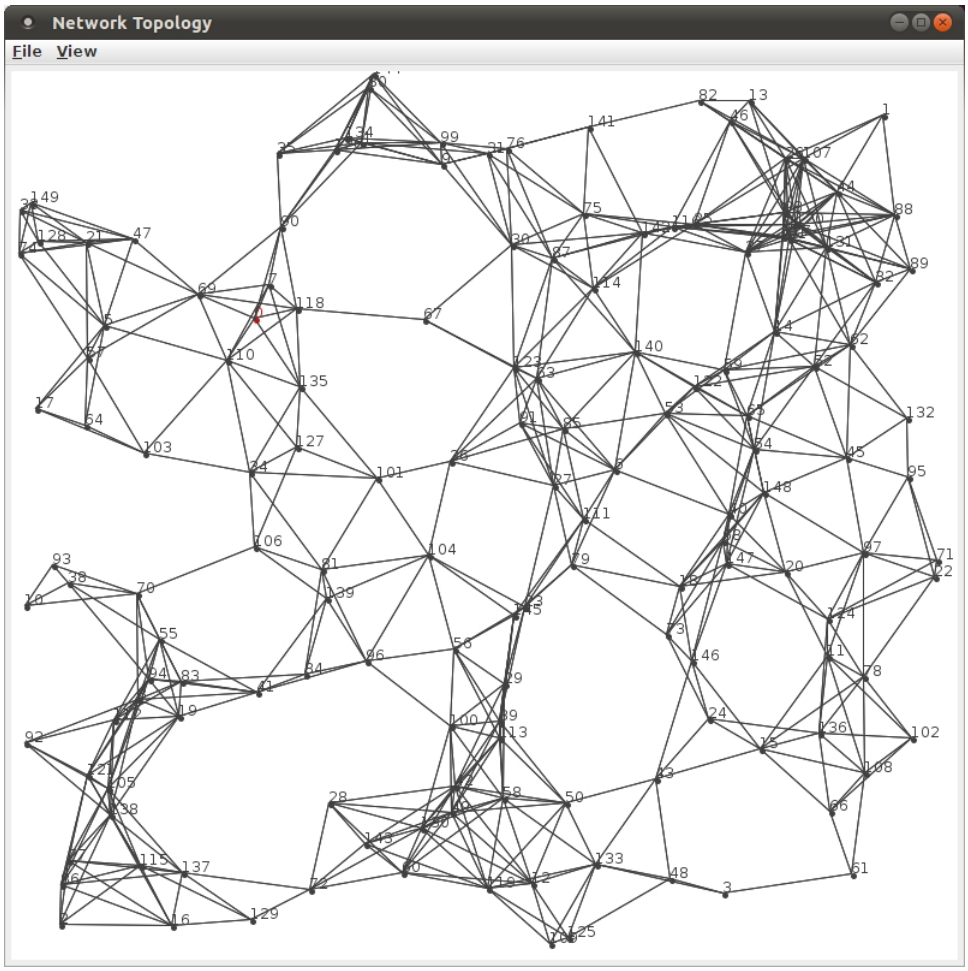


Figura 4.2: Módulo de visualização de topologias

Já dentro da ferramenta, será possível carregar um outro ficheiro de texto que contenha as definições referentes às ligações seguras estabelecidas pelo mecanismo de distribuição de chaves utilizado durante a simulação, que serão assinalados a verde. Desta forma, será possível obter uma imagem global do aspecto da rede após a execução do processo de estabelecimento de chaves.

Convém salientar que esta ferramenta não possibilita uma visualização em tempo real da simulação. Permite apenas visualizar de que forma os nós estão dispostos numa determinada área e, após uma simulação, permite visualizar que ligações seguras foram estabelecidas, tal como apresentado na figura 4.3.

4.1.4 Camada de suporte à comunicação segura (TinySec)

Tal como referido na secção 2.6, devido às características das redes de sensores sem fios e aos requisitos das aplicações perspectivadas para este tipo de redes, a segurança

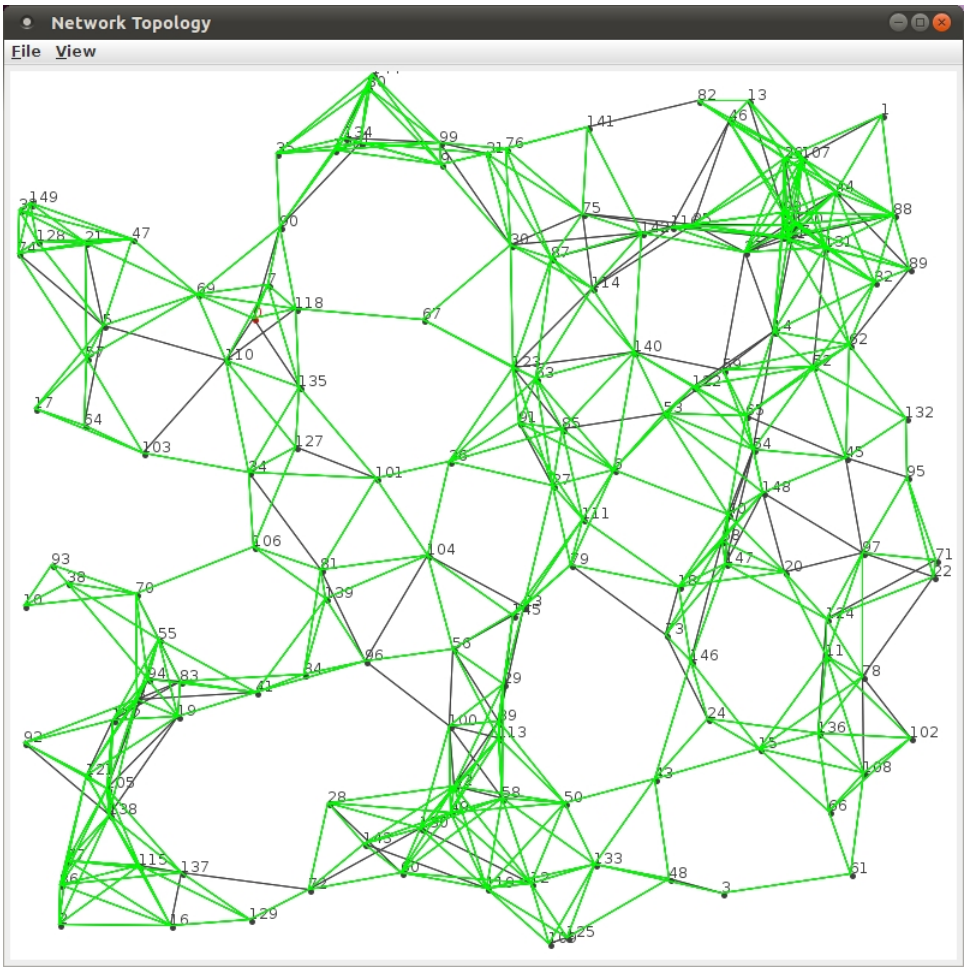


Figura 4.3: Módulo de visualização de topologias com ligações seguras carregadas

desempenha um papel preponderante.

A implementação de uma camada que ofereça suporte à comunicação segura apresenta assim uma importância relevante, tendo como objectivo fornecer quatro grandes propriedades de segurança [22]:

- Autenticidade – através desta propriedade é possível a um nó certificar-se de que a mensagem provém realmente do endereço de onde afirma ter sido enviada. É assim possível evitar ataques em que um nó assuma a identidade de outro nó. Para garantir esta propriedade de segurança são utilizados MAC (*Message Authentication Codes*).

- Integridade – alguns tipos de ataques baseiam-se em capturar mensagens em trânsito, modificá-las e colocá-las de novo na rede. A integridade de mensagens pretende garantir que uma mensagem não sofreu qualquer alteração desde o momento do seu envio. À semelhança da autenticidade, a integridade de mensagens também pode ser garantida se se utilizarem MAC (*Message Authentication Codes*).

- Confidencialidade – esta propriedade garante que um nó não autorizado nunca conseguirá obter os dados de mensagens trocadas na rede. A confidencialidade poderá ser conseguida através de operações de cifra, levando a que dois nós troquem mensagens cifradas em vez de mensagens em claro.
- Protecção de ataques por *replay* – existe um outro tipo de ataques nos quais um nó captura mensagens para as retransmitir mais tarde, o que poderá levantar falhas de segurança pois a mensagem continua a ser válida do ponto de vista das propriedades apresentadas anteriormente e o nó receptor irá aceitá-la. Uma defesa comum passa por incluir um contador nas mensagens. Assim, sempre que um nó receber uma mensagem de um emissor *X*, cujo último contador conhecido é *c*, e o contador da mensagem é inferior a *c*, essa mensagem será descartada.

Figura 4.4: Formato original de uma mensagem

De forma a introduzir as propriedades de segurança acima descritas, é necessário alterar o formato das mensagens. Para tal, em vez do formato originalmente utilizado pelo TOSSIM (identificado na Figura 4.4), passarão a ser utilizados três tipos distintos de mensagens: *plain*, *authenticated* e *encrypted*. Nestes três tipos, o campo *Group* é substituído por um campo *SecType* que identifica o tipo de segurança da mensagem em questão. Esse tipo pode assumir três valores: 1 para mensagens do tipo *plain*, 2 para mensagens do tipo *authenticated* e 3 para mensagens do tipo *encrypted*.

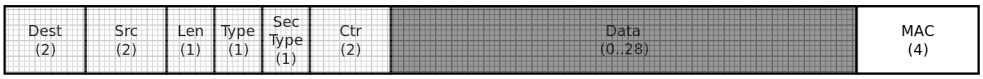
As mensagens *plain*, cuja representação se encontra na Figura 4.5(a), são mensagens que não são cifradas nem autenticadas, substituindo assim o formato das mensagens originais do TOSSIM, não garantindo qualquer propriedade de segurança.

As mensagens *authenticated*, cujo formato se encontra representado na Figura 4.5(b), introduzem dois novos campos na mensagem: i) *Counter* (Ctr) – será um valor inteiro, que irá crescer a cada envio de mensagem e que terá como principal função evitar ataques por *replay*; ii) MAC (*Message Authentication Code*) – que substituirá o campo CRC (*Cyclic Redundancy Check*) e que permitirá garantir as propriedades de autenticidade e integridade. Apesar deste tipo de mensagens ser autenticado, os dados continuam a ser enviados em claro, o que permitirá a um atacante ler o conteúdo da mensagem. No entanto, esse atacante não conseguirá alterar nenhum campo da mensagem, pois o MAC é calculado sobre o cabeçalho e o campo de dados, identificados na Figura 4.5(b) por um padrão quadriculado, protegendo assim a integridade desses campos.

Finalmente, as mensagens *encrypted* são mensagens autenticadas e cifradas, garantindo propriedades de integridade, autenticidade e confidencialidade. Este tipo de mensagens tem um formato idêntico ao das mensagens *authenticated*, possuindo igualmente os dois campos adicionais (*Counter* e MAC). A diferença entre as mensagens do tipo *authenticated* e *encrypted* reside no facto de que as primeiras têm o campo de dados em claro, enquanto que no segundo tipo o campo de dados se encontra cifrado. Na Figura 4.5(c) é possível visualizar a representação deste tipo de mensagens, onde os campos protegidos pelo MAC se encontram preenchidos com uma área quadriculada e o campo de dados se encontra preenchido com um fundo cinzento, indicando que esse campo se encontra cifrado.

(a) Formato de uma mensagem *plain*

(b) Formato de uma mensagem *authenticated*



(c) Formato de uma mensagem *encrypted*

Figura 4.5: Tipos de mensagens do TinySec

De um ponto de vista arquitectural, a aplicação que anteriormente interagia com a camada de rede passa assim a interagir com a nova camada de segurança, que por sua vez tem um papel de intermediário entre a camada de aplicação e a camada de rede, tal como representado na Figura 4.6.

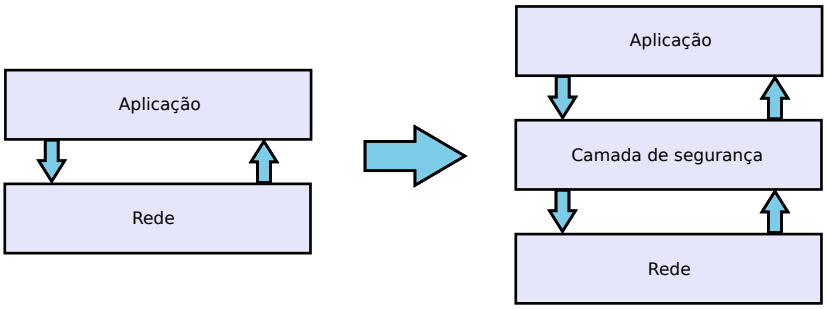


Figura 4.6: Transformação ao nível da pilha de serviços resultante da introdução de uma nova camada que ofereça suporte à comunicação segura

Esta nova camada tem assim como principal fundamento oferecer suporte à comunicação segura, fornecendo três tipos de envio para a camada de aplicação: *sendPlain*, *sendAuth* e *sendEncrypted*, correspondendo aos formatos de mensagem apresentados anteriormente. Em relação à recepção de mensagens, fornece apenas um evento *receive* que será invocado quando existir recepção de mensagens. Ao nível da camada de segurança, será feita uma triagem de acordo com o campo *SecType* e, dependendo do tipo de segurança com que a mensagem foi construída, será verificada a sua autenticidade e integridade e serão decifrados os campos de dados, quando aplicável. Se uma mensagem falhar o processo de verificação do MAC, ela não será passada à camada superior (aplicação) e será descartada. Quando o MAC é verificado e caso se trate de uma mensagem do tipo *encrypted*, ela é passada para a camada de aplicação com o campo de dados já decifrado, garantindo que o impacto sofrido ao nível da aplicação relacionado com a inserção desta nova camada de segurança seja mínimo.

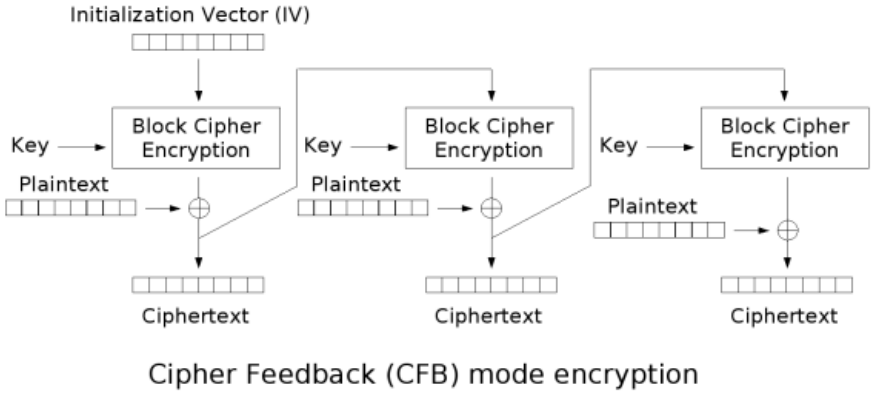
Para introduzir estas noções de cálculo de MAC e cifra de dados é necessário recorrer a algoritmos criptográficos. Para tal, é utilizada uma implementação do algoritmo AES (*Advanced Encryption Standard*) [29] que pode ser encontrada em [3].

No entanto, esta implementação não possuía nenhum modo de operação de cifra de bloco, pelo que para adaptar o algoritmo criptográfico às necessidades energéticas dos sensores foi necessário implementar o modo CFB, tal como representado na figura 4.7.

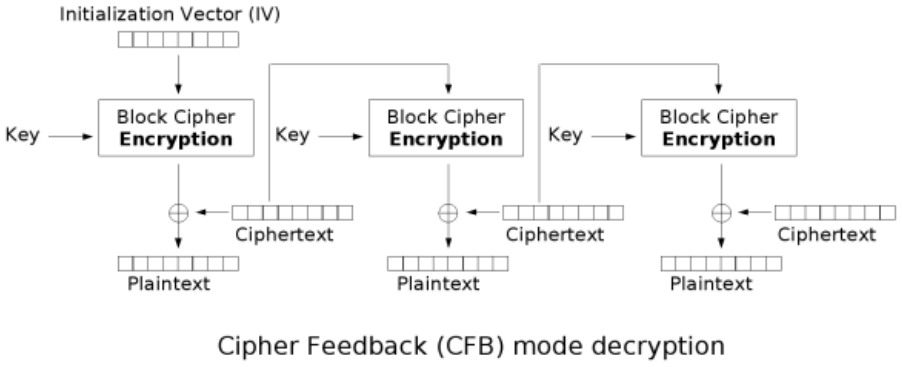
4.1.5 Esquemas de pré-distribuição de chaves

Este componente da arquitectura representa os diversos esquemas de pré-distribuição de chaves que serão alvo de análises experimentais de forma a corresponder aos objectivos desta dissertação. Estes esquemas serão implementados de raiz no âmbito deste trabalho e posteriormente utilizados na fase de avaliações experimentais, obtendo indicadores que permitam comparar os diferentes protocolos.

Os esquemas de pré-distribuição de chaves foram implementados de forma totalmente independente do resto da plataforma, garantindo assim uma boa modularização de código. Assim, se futuramente se desejar avaliar outros esquemas de pré-distribuição de chaves, esta plataforma poderá facilmente servir esse propósito, pois as alterações a efectuar para adaptar os novos esquemas à plataforma já implementada são mínimas.



(a) Operação de cifra



(b) Operação de decifra

Figura 4.7: Modo de operação CFB

4.1.6 Módulos de extracção de indicadores

Através deste componente da arquitectura será possível extrair indicadores que meçam o desempenho de cada um dos protocolos. Serão analisados indicadores de consumo energético, cobertura, fiabilidade e latência.

Para obter dados sobre o consumo energético, bastará modificar o módulo de consumo energético já existente (PowerTOSSIM-Z) de forma a que este considere os novos custos introduzidos pela adição da camada de segurança (custos de cifra, de decifra e de *hashing*). Os restantes módulos de extracção de indicadores serão implementados no âmbito desta dissertação, cujo funcionamento se baseará em pós-processamento de *logs*.

4.1.7 Módulo de interface gráfica de gestão de simulações

Apesar de não estar identificado na visão geral da arquitectura do ambiente de simulação, ilustrada na figura 4.1, foi ainda implementado um módulo de interface gráfica de gestão de simulações que permite definir parâmetros da simulação a efectuar, tais como: o número de nós que compõem a rede, a topologia, a duração da simulação, o esquema de pré-distribuição de chaves a utilizar, etc.

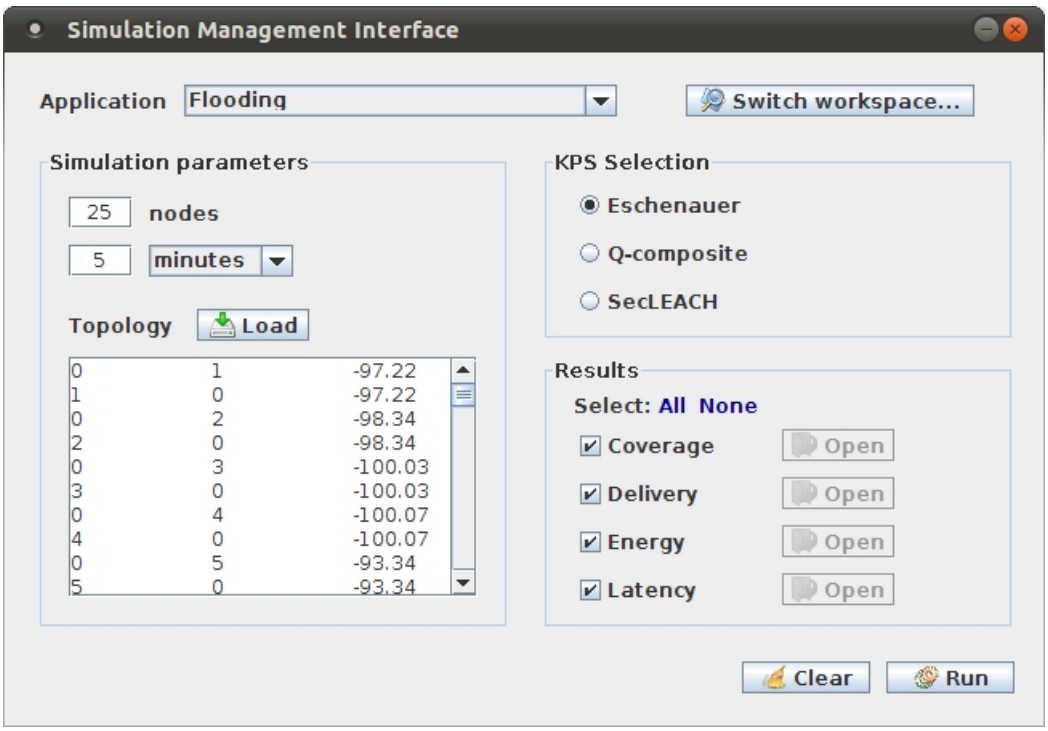


Figura 4.8: Módulo de interface gráfica de gestão de simulações

Esta interface facilita, assim, a utilização da plataforma agilizando o processo de avaliações experimentais.

5

Implementação

Ao longo deste capítulo serão identificadas várias particularidades relacionadas com a implementação dos componentes que constituem o ambiente de simulação desenvolvido no âmbito desta dissertação.

Numa primeira secção, serão abordados os serviços básicos do ambiente de simulação, compreendidos como a camada de suporte à comunicação segura e o algoritmo de encaminhamento utilizado durante a fase de testes.

Posteriormente, serão apresentados os detalhes mais relevantes da implementação dos esquemas de pré-distribuição de chaves.

Finalmente, serão descritas as técnicas utilizadas para a extracção de indicadores relacionados com os critérios em análise nos diferentes protocolos.

5.1 Serviços básicos do ambiente de simulação

No que concerne aos serviços básicos do ambiente de simulação, importa referir qual a táctica utilizada para a implementação da camada de suporte à comunicação segura e do algoritmo de encaminhamento.

5.1.1 Camada de suporte à comunicação segura

Tal como apresentado na secção 4.1.4, quando observada de um ponto de vista arquitectural, esta camada de suporte à comunicação segura encontra-se entre a camada de rede e a camada aplicacional. Desempenha assim um papel de intermediário entre

estas duas camadas, oferecendo um conjunto de serviços que garantem propriedades mínimas de segurança (i.e. confidencialidade, autenticidade, integridade, protecção contra ataques por *replay*).

Para tal, esta camada fornece três chamadas à camada aplicacional: *sendPlain*, *sendAuthenticated* e *sendEncrypted*.

Quando uma aplicação envia uma mensagem através da função *sendPlain*, a camada de segurança preenche os cabeçalhos adicionais introduzidos por esta camada (e.g. campo *SecType*) e envia a mensagem para a rede. Esta função não garante qualquer propriedade de segurança e tem como finalidade enviar mensagens em claro.

Ao enviar uma mensagem através das funções *sendAuthenticated* e *sendEncrypted*, a camada de segurança preencherá igualmente os cabeçalhos adicionais e enviará tantas mensagens quanto o número de vizinhos com os quais tem chaves emparelhadas. Para tal, esta camada irá aceder a um repositório de chaves que foi resultado do processo de estabelecimento de ligações seguras. Nesse repositório, é feita uma associação entre o endereço de cada vizinho e o identificador da chave partilhada com ele. Assim, para cada vizinho, será obtida a chave partilhada com este, e será enviada uma mensagem assinada, no caso da função *sendAuthenticated*, ou uma mensagem cifrada e assinada, no caso da utilização da função *sendEncrypted*.

5.1.2 Algoritmo de encaminhamento utilizado

Para que os nós de uma rede possam comunicar entre si e difundir mensagens por essa mesma rede, é necessário estar definido um algoritmo de encaminhamento, o qual dita o comportamento de um nó ao receber uma mensagem (i.e. para quem a deve reencaminhar).

No âmbito desta dissertação, o algoritmo de encaminhamento utilizado foi o de *flooding* com filtragem de duplicados. Para tal, cada nó mantém uma lista de assinaturas de mensagens que já recebeu até um determinado momento. Ao receber uma nova mensagem, este poderá assim verificar se se trata de um duplicado.

5.2 Esquemas de pré-distribuição de chaves

Nesta secção serão analisadas as aproximações utilizadas para a implementação dos esquemas de pré-distribuição de chaves que constituem o alvo deste estudo.

5.2.1 Esquema de Eschenauer

O esquema básico criado por Eschenauer e Gligor pode ser descrito por 3 fases distintas, tal como referido na secção 3.3.1.

A primeira fase consiste no carregamento de anéis de chaves na memória dos sensores. De forma a simular este comportamento, em cada nó foi incluído um ficheiro *header* que continha o seu anel de chaves.

Durante a segunda fase, é suposto os vários nós da rede descobrirem que chaves partilham entre si. Para tal, foi adoptado o seguinte comportamento:

$$A \xrightarrow{lista\ de\ ids} B$$
$$A \xleftarrow{id,\ \{challenge\}_{K_{id}}} B$$
$$A \xrightarrow{\{challenge+1\}_{K_{id}}} B$$

Durante esta fase, os nós irão enviar mensagens, por *broadcast*, com os identificadores das chaves que constituem o seu anel de chaves. Os nós que recebam estas mensagens, irão confrontar o conteúdo das mesmas com o seu próprio anel de chaves. Se existir uma chave em comum, será enviada uma resposta que contém o identificador da chave em comum e um *challenge* cifrado com essa mesma chave. O primeiro nó, ao receber esta mensagem, terá que a decifrar e enviar uma resposta constituída por esse *challenge* incrementado em 1 valor, igualmente cifrada com a chave agora partilhada entre ambos.

Numa terceira fase, os nós que tenham detectado a existência de outros nós no seu raio de comunicação com os quais não partilhem nenhuma chave, irão tentar estabelecer uma chave com esses nós, recorrendo às ligações seguras já definidas pela fase 2. Para tal, o nó que desencadear esta acção, seleccionará uma chave do seu anel de chaves que ainda não esteja a ser utilizada para nenhuma ligação. Esta chave será enviada através das ligações seguras já definidas pela fase anterior, na esperança de que exista caminho até ao nó com o qual se pretende estabelecer ligação.

Dado que o número de colisões detectadas após a implementação deste protocolo era bastante elevado, tornou-se necessário introduzir atrasos de duração aleatória no envio de mensagens.

5.2.2 Esquema Q-Composta

A implementação do protocolo Q-Composta é bastante semelhante à do esquema básico de Eschenauer, sendo que em vez de se descobrir apenas uma chave em comum entre os diferentes nós, será necessário descobrir q chaves em comum.

Assim, a primeira fase deste protocolo é idêntica à do esquema de Eschenauer, onde os anéis de chaves são carregados para a memória dos sensores. Tal como nesse esquema, para efeitos de simulação, os anéis foram gerados para um ficheiro *header*, o qual foi introduzido no código dos sensores.

Na segunda fase, os nós irão descobrir quais as chaves comuns que partilham entre si através do comportamento descrito no diagrama abaixo apresentado:

$$
\begin{array}{lcll}
A & \xrightarrow{grupo\ 1\ de\ ids} & B & \\
A & \xleftarrow{id_1, id_2} & B & \\
A & \xrightarrow{grupo\ 2\ de\ ids} & B & \\
A & \xleftarrow{id_1, id_2, id_3} & B & \\
A & \xleftarrow{\{challenge_B\}_{K_{ab}}} & B & ab = hash(K_{id_1}||K_{id_2}||K_{id_3}) \\
A & \xrightarrow{\{challenge_A\}_{K_{ab}}} & B & \\
A & \xrightarrow{\{challenge_B+1\}_{K_{ab}}} & B & \\
A & \xleftarrow{\{challenge_A+1\}_{K_{ab}}} & B &
\end{array}
$$

Um nó começa por enviar a lista de identificadores que compõe o seu anel de chaves. Os outros nós que recebem esta mensagem, enviam uma resposta com os identificadores em comum descobertos até então. Quando o número de identificadores em comum é maior ou igual a q, ambos calculam uma chave que é resultado do *hash* das várias chaves que eles partilham. Depois de terem calculado a chave, cada um deles irá enviar um *challenge* cifrado pela nova chave agora partilhada entre eles. Ao receberem esse *challenge*, irão enviar uma resposta constituída por *challenge+1*, igualmente cifrado pela chave calculada entre ambos.

Este protocolo requer um maior número de mensagens trocadas de forma a estabelecer ligações seguras entre os nós. Como tal, à semelhança do que aconteceu no protocolo de Eschenauer, revelou-se necessário introduzir atrasos de duração aleatória no envio de mensagens. Estes atrasos foram parametrizados com um limite inferior e um limite superior, que tiveram de ser maiores que os utilizados no esquema de Eschenauer, de forma a conseguir resultados equivalentes.

O algoritmo de *hashing* utilizado neste protocolo foi o SHA-1, cuja implementação em C se poderá encontrar no RFC 3174 [15]. Esta implementação foi adaptada para nesC e incluída no ambiente de simulação.

5.2.3 Esquema SecLEACH

No protocolo SecLEACH teremos dois tipos de nós: os *Cluster Heads*, que serão os coordenadores dos *clusters*, e os nós comuns, que tentarão juntar-se a estes *clusters*.

Na especificação do protocolo é referido que todos os nós da rede conseguem comunicar com a *base station* se utilizarem uma intensidade de sinal suficientemente forte. Para a implementação deste protocolo, seguiu-se uma aproximação onde, em vez de todos os nós terem a capacidade de comunicar directamente com a *base station*, as mensagens entre estes são difundidas pela rede através de um algoritmo simples de *flooding*, descrito na secção 5.1.2.

Durante o funcionamento deste protocolo, uma percentagem dos nós da rede irá auto-eleger-se CH e irá enviar uma mensagem onde se anuncia como tal, contendo o seu identificador e um *nonce*. Os nós que escutem esta mensagem, irão determinar o anel de chaves do CH em questão através duma função de geração de números pseudo-aleatórios que receberá como semente o identificador desse CH. Caso o nó partilhe uma chave com este CH, ele irá enviar um pedido de junção ao *cluster*, assinado com a chave partilhada entre ambos. O CH, ao receber este pedido, irá enviar uma resposta de confirmação igualmente assinada com a chave partilhada entre ambos.

O funcionamento deste protocolo é descrito pelo diagrama abaixo apresentado:

$$
\begin{array}{lcl}
CH & \xrightarrow{id_{CH},\, nonce} & A \\
CH & \xleftarrow{id_A,\, id_{CH},\, r,\, mac_{K_r}(id_A,\, id_{CH},\, r,\, nonce)} & A \\
CH & \xrightarrow{\{nonce\}_{K_r}} & A
\end{array}
$$

Devido ao facto de este protocolo necessitar de trocar poucas mensagens entre os nós para que a rede se auto-organize, os atrasos de duração aleatória no envio das mensagens foram parametrizados com valores muito inferiores aos utilizados nos esquemas de Eschenauer e Q-Composta.

5.3 Módulos de extracção de indicadores

De forma a poder avaliar o desempenho e a eficácia de cada um dos protocolos, foi necessário introduzir mecanismos que permitissem obter métricas referentes a vários critérios de avaliação, tais como: consumo energético, latência, taxa de cobertura e taxa de fiabilidade. O modo como estes indicadores foram extraídos do ambiente de simulação será descrito nas secções seguintes.

5.3.1 Módulo de cálculo de consumo energético

Para calcular o consumo energético de uma simulação recorreu-se ao módulo PowerTOSSIM-Z já existente. No entanto, este módulo não contempla os custos introduzidos pela camada de suporte à comunicação segura, pelo que foi necessário complementar a sua acção com uma ferramenta que considere estes custos adicionais.

O funcionamento do PowerTOSSIM-Z consiste em processar registos resultantes de uma simulação, o que lhe irá permitir calcular o consumo energético total de cada nó. De forma a incluir nestes cálculos o consumo energético introduzido pelas operações criptográficas e funções de *hashing*, foi necessário criar uma ferramenta adicional que fizesse o processamento de um outro ficheiro de texto onde estas operações eram registadas.

Os cálculos foram feitos com base nos valores indicados na tabela 5.1, encontrados em [38].

Algoritmo	**Energia**
SHA-1	5.9 μJ/byte
AES 128 Bytes, cifra	1.62 μJ/byte
AES 128 Bytes, decifra	2.49 μJ/byte

Tabela 5.1: Tabela de custos associados às operações criptográficas e função de *hashing*

5.3.2 Módulo de latência

Existem dois tipos de latência que interessam medir: o tempo de estabilização de um protocolo e o tempo que uma mensagem leva da origem até ao destino.

Para medir a primeira noção de latência apresentada, bastou criar uma ferramenta que analisasse os registos da simulação e registasse o instante temporal em que foi trocada a última mensagem originada pelo funcionamento do processo de estabelecimento de chaves.

Para medir o tempo que uma mensagem leva desde a origem até ao destino, foi adicionado um campo à estrutura metadata de cada mensagem. Este campo é preenchido com o instante temporal em que a mensagem foi enviada pelo nó inicial. Quando a mensagem chega à *base station* é calculada a diferença entre esse instante e o instante registado na estrutura de metadata da mensagem.

No final da simulação é apresentada a latência média de todas as mensagens recebidas pela *base station*.

5.3.3 Módulo de cobertura

Neste trabalho foram considerados dois tipos de cobertura: cobertura potencial e cobertura efectiva.

Para determinar a primeira, foi criada uma ferramenta de pós-processamento de registos de simulação que analisa o progresso do funcionamento do esquema de pré-distribuição de chaves e vai registando quais os nós que partilham pelo menos uma chave com um vizinho. No final dessa análise, devolve o número total de nós que partilham pelo menos uma chave com um dos seus vizinhos.

De forma a calcular a taxa de cobertura efectiva de uma rede, criou-se uma aplicação de teste que faz com que cada um dos nós dessa rede envie, à vez, 5 mensagens espaçadas por 5 segundos cada uma. Os nós que consigam fazer chegar mensagens à *base station* serão considerados nós efectivamente cobertos.

5.3.4 Módulo de fiabilidade

Para determinar a taxa de fiabilidade de uma simulação, recorreu-se a uma técnica muito semelhante à utilizada no cálculo da taxa de cobertura efectiva – foi desenvolvida uma aplicação de teste, na qual os nós enviam 5 mensagens espaçadas por 5 segundos cada uma. A *base station* conta o número total de mensagens que recebeu e comparando-o com o número total de mensagens enviadas, devolve a taxa de fiabilidade conseguida por aquela rede.

Para a realização deste teste foi também interessante averiguar qual o tipo de emissor das mensagens no caso do SecLEACH. Para tal, foi adicionado um campo booleano à estrutura metadata das mensagens que tomou um valor verdadeiro quando estas foram enviadas por um *Cluster Head* e um valor falso quando as mensagens foram enviadas por nós órfãos.

6

Avaliação experimental

Como referido anteriormente, a análise de desempenho dos vários esquemas de pré-distribuição de chaves disponível na literatura actual é meramente teórica. Torna-se, por isso, fundamental analisar esses mesmos esquemas através de avaliações experimentais que levem em linha de conta factores comuns ao funcionamento normal duma rede de sensores sem fios, tais como ruído nas comunicações, colisões, entre outros. Assim sendo, é possível identificar duas fases distintas durante o tempo de vida útil duma RSSF: a fase de configuração e a fase de operação. A fase de configuração consiste na execução de um esquema de pré-distribuição de chaves, cujo processo terá como resultado final uma topologia que será fornecida à camada de encaminhamento. Esta fase permitirá, portanto, estabelecer ligações seguras que garantam determinadas propriedades de segurança (confidencialidade, integridade e autenticidade). Desta forma, interessa analisar o desempenho de cada um dos protocolos relativamente aos seguintes critérios:

- Consumo energético
- Cobertura potencial
- Cobertura efectiva
- Tempo de estabilização

Após a estabilização da rede segue-se a fase de operação onde esta desenvolverá a tarefa para a qual foi programada. Nesta fase, serão utilizadas as ligações seguras

definidas pela fase de configuração. Assim sendo, o desempenho de cada um dos protocolos terá ainda repercussões na fase de operação, pelo que será interessante analisar as seguintes dimensões:

- Consumo energético
- Fiabilidade
- Latência

Para estes testes foram criadas várias topologias com recurso à ferramenta de geração de topologias fornecida pelo próprio simulador, na qual foi definido que o raio de comunicação de um sensor seria de aproximadamente 50 metros. Através destas definições, foram encontradas áreas óptimas para diferentes tamanhos de rede, que vão desde os 25 nós até aos 150 nós, conforme indicado na seguinte tabela:

Número de nós	**Dimensões**	**Área**
25	180x180	32 400 m^2
50	230x230	52 900 m^2
75	280x280	78 400 m^2
100	330x330	108 900 m^2
125	380x380	144 400 m^2
150	430x430	184 900 m^2

Tabela 6.1: Relação entre a dimensão da rede e a área utilizada

De forma a garantir condições de igualdade entre os vários testes efectuados, foram definidos determinados parâmetros e configurações.

Os tamanhos dos anéis de chaves utilizados foram determinados tendo em conta que a probabilidade de dois nós conseguirem estabelecer uma ligação seria de 0.99. Assim sendo, a partir de *pools* de 1000 chaves, foram gerados anéis de 75, 100 e 75 chaves para os protocolos de Eschenauer, Q-Composta e SecLEACH, respectivamente. Dado que cada chave do anel terá 16 bytes, o espaço total necessário para armazenar um anel de 75 chaves será de 1200 bytes e a memória necessária para armazenar um anel de 100 chaves será de 1600 bytes. Se tomarmos como exemplo os sensores MICAz que têm 128 KBytes de memória para aplicações, um anel com 75 chaves irá ocupar 0.9375% da memória total disponível e um anel de 100 chaves ocupará 1.25% do espaço total de memória, o que se consideram ser valores bastante realistas.

Em relação ao algoritmo de encaminhamento utilizado, foi aplicada uma versão simples do algoritmo de *flooding* com filtragem de duplicados.

No caso do protocolo Q-Composta, o valor de Q utilizado nos testes foi de 3, ou seja, para dois nós poderem definir uma ligação segura entre si, eles terão de partilhar 3 ou mais chaves.

Para o protocolo SecLEACH, a probabilidade de um nó se auto-eleger *Cluster Head* é de 20%.

Para a construção dos gráficos apresentados ao longo deste capítulo foram utilizados dados obtidos através da média de várias experiências, com o objectivo de mitigar grandes discrepâncias nos indicadores a medir.

6.1 Fase de configuração

Nesta secção serão avaliados os diferentes protocolos de acordo com os critérios anteriormente citados, tendo em conta a seguinte organização: na secção 6.1.1 irá determinar-se qual a energia consumida por cada um dos protocolos; na secção 6.1.2 será calculada a cobertura potencial garantida por cada um dos esquemas; na secção 6.1.3 será determinada a cobertura efectiva oferecida pelos vários protocolos; e, finalmente, na secção 6.1.4 será medido o tempo necessário para a execução de cada protocolo.

6.1.1 Análise de consumo energético

Dada a limitação de recursos energéticos existente nas RSSF torna-se imperativo averiguar qual o protocolo mais pesado do ponto de vista da energia dispendida e qual a evolução desse consumo com o aumento do tamanho da rede. Para isso é necessário medir qual a energia dispendida por cada um desses protocolos em condições de igualdade de cirunstâncias. Para tal, foram geradas várias topologias de diferentes tamanhos de rede e para cada uma delas foram testados os protocolos em estudo, capturando a energia total consumida pela rede.

Os resultados obtidos estão apresentados na figura 6.1 e da análise deste gráfico interessam realçar essencialmente dois pontos: i) por um lado, o baixo consumo energético do esquema SecLEACH mesmo quando se verifica um aumento do tamanho da rede (em alguns testes para redes pequenas chegou a apresentar valores inferiores a 1 mJ); ii) por outro lado, o consumo energético dos esquemas Eschenauer e Q-Composta que demonstra ser proporcional ao número de nós na rede.

O baixo consumo energético do esquema SecLEACH deve-se ao facto de este protocolo utilizar poucas mensagens para organizar a rede. Um nó que se eleja *Cluster Head* terá apenas que enviar $n+1$ mensagens, onde n é o número de nós que constituem esse

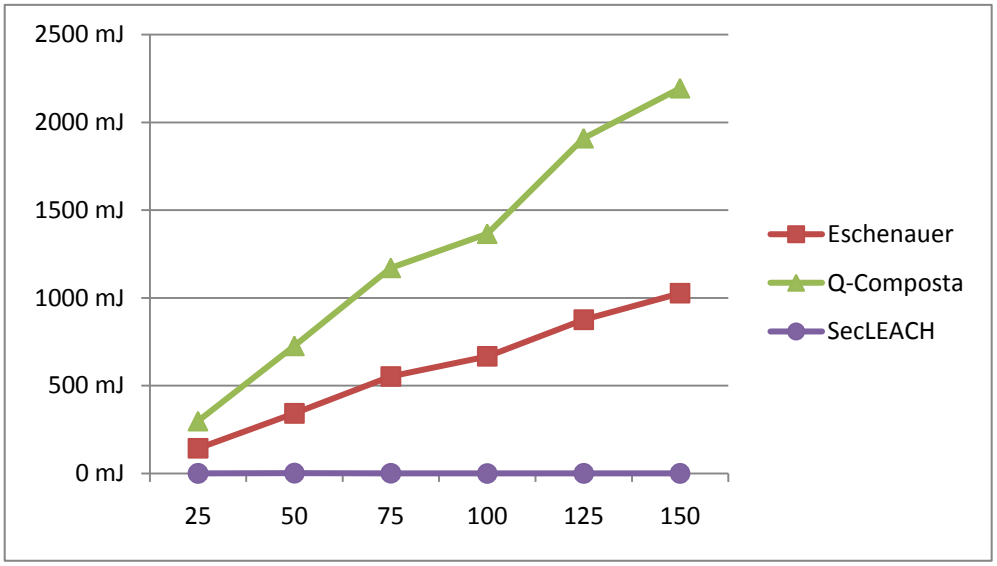


Figura 6.1: Análise de consumo energético, resultante da fase de configuração, para redes de dimensões de 25 até 150 nós

cluster, ao passo que um nó comum terá apenas de escutar o anúncio de um CH, enviar uma única mensagem de pedido de junção ao *cluster* e aguardar pela resposta de confirmação. Os nós mais sobrecarregados do ponto de vista de consumo energético são os CH mas que, ainda assim, apresentam um consumo bastante baixo, mesmo quando a dimensão da rede aumenta.

No caso dos esquemas de Eschenauer e Q-Composta, o número de mensagens trocadas é extremamente superior ao número de mensagens trocadas no SecLEACH, o que se reflecte na energia total consumida pela rede. Nestes esquemas, um nó tentará emparelhar chaves com todos os seus vizinhos, processo esse que requer um número elevado de mensagens trocadas, com especial incidência no Q-Composta por duas principais razões: i) o número de chaves que dois nós precisam de descobrir entre si é maior que no esquema de Eschenauer, o que implica maior troca de mensagens; ii) os anéis de chaves são tipicamente maiores que no esquema de Eschenauer, o que também se traduz numa maior troca de mensagens.

6.1.2 Análise de cobertura potencial

O segundo critério analisado está relacionado com as questões de cobertura da rede. No caso dos esquemas de Eschenauer e Q-Composta, um nó diz-se potencialmente coberto quando partilha uma ou mais chaves com um ou mais nós. No caso da variante sem propriedades de segurança e no caso do SecLEACH, para um nó se considerar potencialmente coberto bastará ter um vizinho. Para este último protocolo, isto só se verifica porque cada um dos nós partilha uma chave única com a *base station*, o que

permite difundir mensagens pela rede sem ser necessário recorrer a ligações seguras. Este indicador pode, assim, dar-nos uma primeira noção sobre a eficácia de cada um dos protocolos no que diz respeito ao estabelecimento de ligações seguras. Para medir este critério foram geradas várias topologias, de várias dimensões de rede, e para cada uma delas foram testados os três esquemas em análise e ainda uma outra variante que não utiliza qualquer suporte de segurança (i.e. não é feito qualquer estabelecimento de ligações seguras, pelo que bastará a um nó ter pelo menos um vizinho para se considerar potencialmente coberto). Esta variante permitirá assim ter uma base de comparabilidade face à utilização, ou não, de topologias seguras.

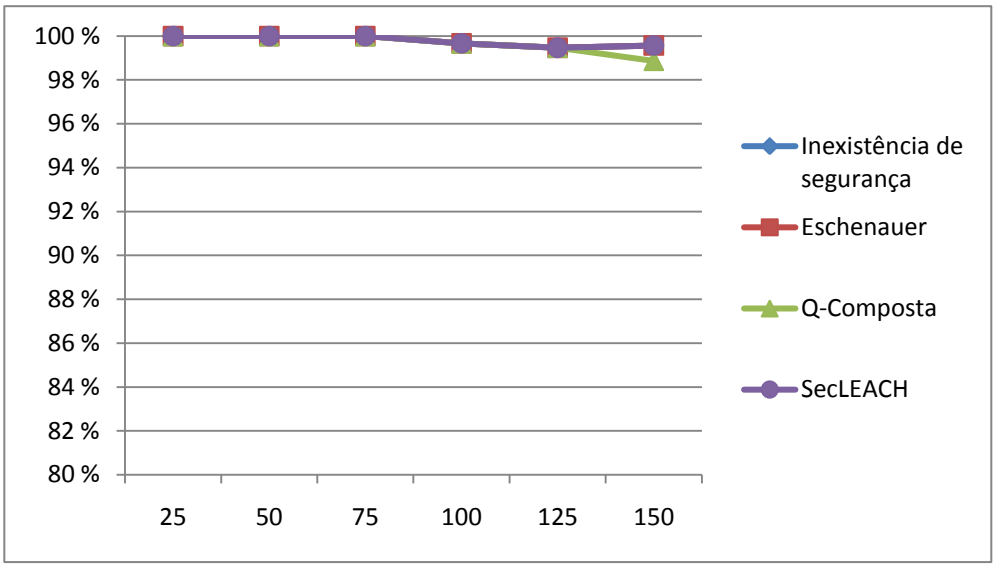


Figura 6.2: Análise de cobertura potencial, após fase de configuração, para redes de dimensões de 25 até 150 nós

A figura 6.2 apresenta os resultados obtidos para a cobertura potencial. Neste gráfico, convém salientar que as linhas correspondentes à variante de inexistência de segurança, ao esquema de Eschenauer e ao SecLEACH estão sobrepostas. No caso do Eschenauer, isto significa que para todos os nós com pelo menos um vizinho foi possível emparelhar chaves com pelo menos um deles, resultando assim em indicadores de cobertura potencial idênticos aos indicadores obtidos na inexistência de segurança, onde basta existir um vizinho para um nó se considerar potencialmente coberto. No caso do SecLEACH, dado que cada nó possui uma chave partilhada com a *base station*, os indicadores obtidos foram também idênticos àqueles conseguidos na variante sem segurança, pois mesmo que um nó não se consiga juntar a nenhum *cluster*, ele continua a poder enviar mensagens até à *base station*, desde que exista um caminho até à mesma, o que resulta em indicadores idênticos aos obtidos na variante sem qualquer tipo de segurança.

Importa, ainda, salientar que nas redes de 25 até 75 nós se obteve uma cobertura potencial de 100%, o que significa que não havia nós isolados da rede. Com o aumento da dimensão da rede (i.e. dos 100 até aos 150 nós), a probabilidade de existerem nós desconectados aumentou ligeiramente, verificando-se a existência de algumas "ilhas".

Finalmente, verificou-se ainda uma cobertura potencial ligeiramente inferior nas redes de maior escala (150 nós) para o protocolo Q-Composta. Ou seja, dos nós que se encontravam potencialmente cobertos na variante sem segurança, houve ainda alguns que, por não terem conseguido emparelhar chaves com pelo menos um vizinho, ficaram desconectados, o que resultou numa cobertura potencial um pouco abaixo da obtida com os outros protocolos. No entanto, e dado que os anéis de chaves utilizados garantem uma probabilidade de partilha de chave entre dois nós de 0.99, os resultados obtidos vão de encontro ao esperado nos estudos teóricos, dado que as coberturas potenciais obtidas estão próximas deste valor.

6.1.3 Análise de cobertura efectiva

Existe outro tipo de cobertura merecedor de observação atenta: a cobertura efectiva, que define o aproveitamento obtido a partir dos sensores lançados numa rede. Considera-se que um nó está efectivamente coberto quando existe um caminho entre este e a *base station*. A noção de caminho altera-se entre a variante sem suporte de segurança e os esquemas de pré-distribuição de chaves em estudo. No caso da variante sem suporte de segurança e no caso do SecLEACH, para um nó se considerar efectivamente coberto basta que exista um caminho entre este e a *base station* (note-se que no SecLEACH isto só é possível porque cada um dos nós partilha uma chave única com a *base station*). No caso dos esquemas de Eschenauer e Q-Composta, para que um nó esteja efectivamente coberto, é necessário que exista um caminho formado unicamente por ligações seguras entre esse nó e a *base station*.

Por forma a determinar a existência de tal caminho, cada um dos nós da rede irá enviar à vez 5 mensagens espaçadas no tempo por intervalos de 5 segundos. As mensagens, ao chegarem à *base station*, servirão como prova de que o nó que as enviou se encontra efectivamente coberto. No final da simulação, a *base station* irá devolver o número de nós cobertos, que será objecto de análise.

De forma a avaliar este critério foram criadas diferentes topologias de várias dimensões. Para cada uma destas topologias foram testados os três esquemas que são alvo de estudo e ainda a variante que não oferece qualquer garantia de segurança, que servirá apenas como termo de comparação.

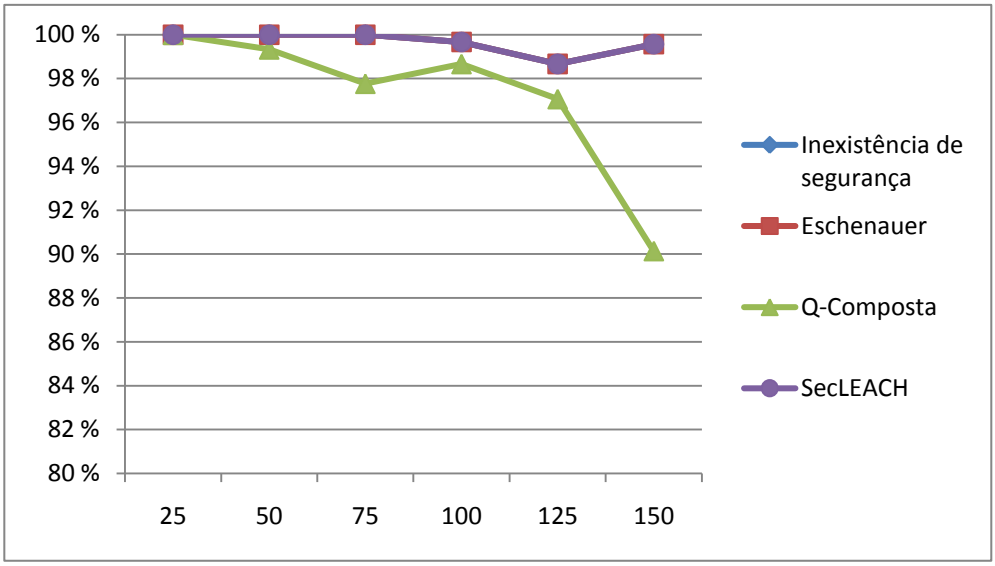


Figura 6.3: Análise de cobertura efectiva, após fase de configuração, para redes de dimensões de 25 até 150 nós

A figura 6.3 dá-nos conta dos resultados obtidos para a cobertura efectiva. À semelhança do que aconteceu com a cobertura potencial, os indicadores obtidos para a variante sem suporte de segurança e para os esquemas Eschenauer e SecLEACH foram idênticos. Isto significa que os nós que garantiam propriedades de cobertura efectiva na variante sem suporte de segurança, conseguiram também garantir essas mesmas propriedades nos esquemas acima referidos. No caso do protocolo Q-Composta, os resultados divergiram ligeiramente e foi possível verificar que com o aumento da dimensão da rede, a cobertura efectiva ia diminuindo gradualmente, sendo que para uma rede de 150 nós, apenas 90% dos nós se encontravam efectivamente cobertos (i.e. conseguiam fazer chegar mensagens até à *base station*).

6.1.4 Análise de tempo de estabilização

O último critério analisado na fase de configuração diz respeito ao tempo de estabilização de cada protocolo. Entende-se por tempo de estabilização o tempo que cada protocolo leva até terminar a sua execução. Para capturar estes dados, foram criadas várias topologias de diferentes dimensões e para cada uma delas foram testados os vários protocolos em estudo. De cada simulação efectuada obteve-se um registo que continha todos os instantes temporais onde foram trocadas mensagens resultantes do funcionamento dos esquemas de pré-distribuição de chaves. Para determinar o tempo total que um esquema necessitou para terminar a sua tarefa bastou identificar a última entrada desse registo criado, correspondente à última mensagem trocada. Calculando a diferença entre esse último instante temporal e o instante em que o protocolo iniciou

o seu funcionamento, é possível determinar qual a duração necessária para cada um dos protocolos, face a diferentes dimensões de rede.

Realizar este tipo de estudo é importante pois uma rede só poderá dar início à fase de operação depois de ter terminado a fase de configuração. Torna-se, pois, essencial identificar de que forma cada um dos protocolos responde a este requisito.

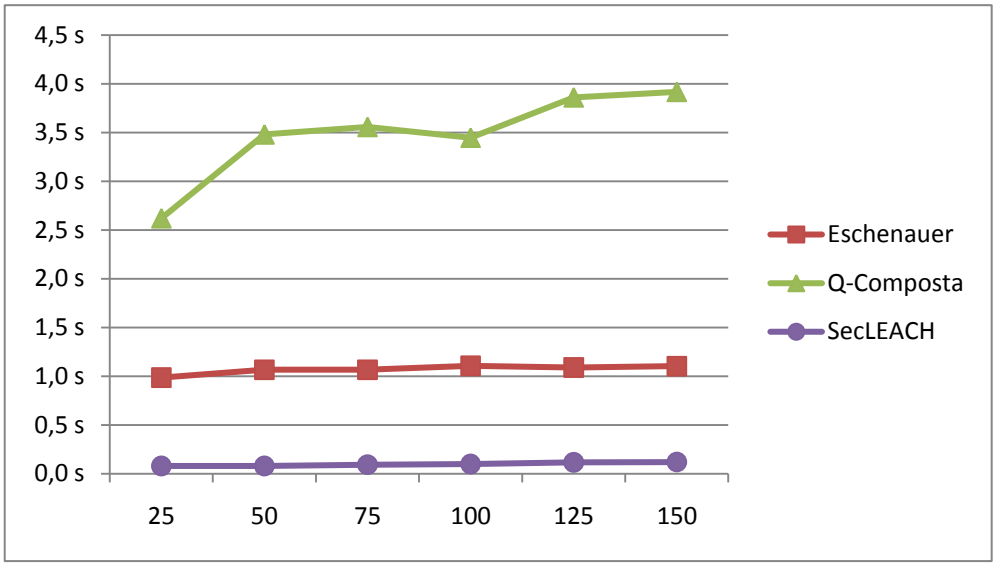


Figura 6.4: Análise de tempo de estabilização da fase de configuração para redes de dimensões de 25 até 150 nós

No gráfico da figura 6.4 estão patentes os resultados conseguidos através das avaliações experimentais efectuadas.

Na fase de implementação dos protocolos constatou-se que existiam demasiadas colisões que impediam o correcto funcionamento destes. Para resolver este problema foram introduzidos pequenos atrasos de duração aleatória no envio de mensagens, de forma a evitar que houvesse vários nós a tentar responder em simultâneo. Estes atrasos de duração aleatória foram parametrizados com valores mínimos e máximos, de acordo com as necessidades de cada protocolo. No caso do SecLEACH, os atrasos introduzidos foram mínimos pois o número de mensagens trocadas é relativamente baixo quando comparado com os protocolos de Eschenauer e Q-Composta, pelo que o SecLEACH conclui o seu funcionamento num período de tempo de aproximadamente uma décima de segundo. Em redes de pequena escala (de 25 até 75 nós) o protocolo terminou a sua execução em períodos inferiores a uma décima de segundo. Para redes maiores (até 150 nós), o protocolo apresentou resultados um pouco acima da décima de segundo (entre 0,10 e 0,13 segundos).

No esquema de Eschenauer, dado que o número de mensagens trocadas é superior

ao número de mensagens utilizadas no SecLEACH, foi necessário introduzir mais atrasos. Este comportamento reflecte-se no tempo total de execução do protocolo, pelo que os valores observados rondam 1 segundo: para redes de 25 nós, obtiveram-se resultados inferiores a 1 segundo; para redes de 50 ou mais nós, os dados obtidos situam-se entre 1 segundo e 1,10 segundos.

Finalmente, o esquema Q-Composta, pelas características que lhe são inerentes, revelou ser o mais demorado de entre os três protocolos estudados. Dado que o número de mensagens trocadas neste protocolo é superior comparativamente ao número de mensagens trocadas nos outros dois protocolos, os atrasos introduzidos durante o seu funcionamento acabam por penalizar o resultado final. Os valores encontrados situam-se entre 2,62 segundos, para uma rede de 25 nós, e 3,92 segundos, para uma rede de 150 nós. Este protocolo é também o que é mais penalizado com o aumento da dimensão da rede, tendo-se verificado uma variação de aproximadamente 1,5 segundos entre o tempo de estabilização obtido para uma rede de 25 nós e o tempo de estabilização para uma rede de 150 nós.

6.2 Fase de operação

Após a fase de configuração estar concluída, encontram-se definidas as ligações seguras entre os nós. A rede encontra-se assim num estado em que os nós conseguem comunicar entre si garantindo determinadas propriedades de segurança, dando início à fase de operação. É nesta fase que a rede irá cumprir o objectivo para o qual foi desenhada (e.g. medir temperatura, humidade, luminosidade, etc.). Durante esta fase, serão avaliados três critérios distintos: na secção 6.2.1 será medida a energia consumida pela rede durante o seu funcionamento normal; na secção 6.2.2 será determinado o nível de fiabilidade da rede quando aplicado cada um dos protocolos; e, por fim, na secção 6.2.3 será analisada a latência média das mensagens enviadas pelos nós para a *base station*.

Para a recolha de dados das análises elaboradas nesta secção, definiu-se como fase de operação a selecção aleatória de 20% de nós da rede como nós geradores de eventos. Estes nós irão enviar uma mensagem de 5 em 5 segundos para a *base station* até atingirem um limite de 5 mensagens enviadas.

6.2.1 Análise de consumo energético

O teste elaborado para este indicador pretende apurar qual o impacto energético que cada um dos protocolos tem na fase de operação. Para este efeito, foram criadas diversas topologias de várias dimensões. Para cada uma delas, foram testados os protocolos

em estudo e calculado o valor total da energia consumida pela rede, necessário para pôr em prática a fase de operação anteriormente descrita.

O gráfico da figura 6.5 apresenta os resultados obtidos na elaboração das experiências realizadas para este indicador.

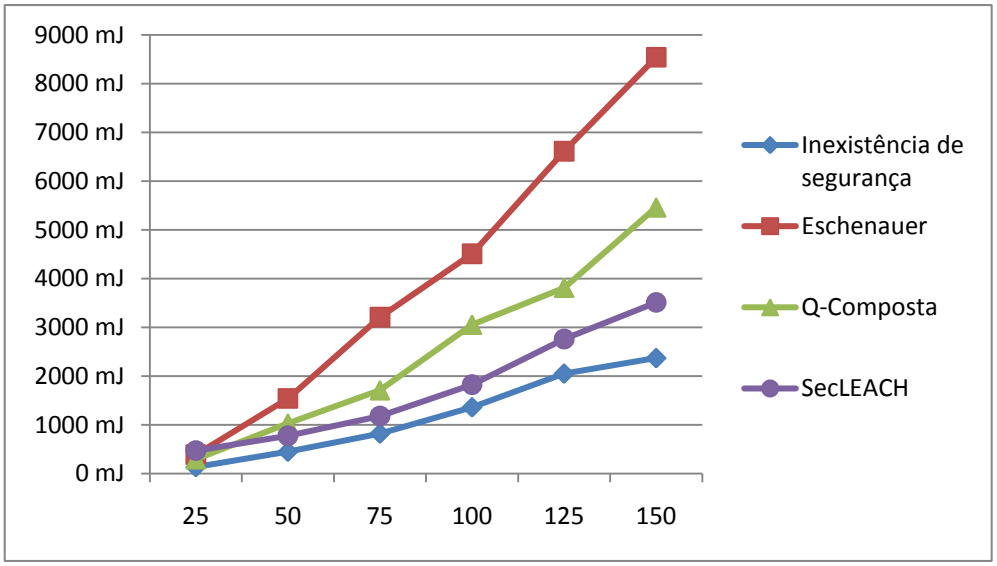


Figura 6.5: Análise de consumo energético, na fase de operação, para redes de dimensões de 25 até 150 nós

Neste gráfico, é possível observar que a variante sem suporte de segurança é a que consome menos energia. Em relação aos três protocolos em estudo, verifica-se que o que consome menos energia é o SecLEACH, que obtém uma variação de valores entre 450 e 3500 mJ. A variação de valores para o protocolo Q-Composta situa-se entre 300 e 4600 mJ e para o esquema de Eschenauer situa-se entre 400 e 8500 mJ.

No caso do SecLEACH, o baixo consumo energético (quando comparado com o consumo do protocolo de Eschenauer ou com o protocolo Q-Composta) justifica-se pela especificidade do próprio protocolo, principalmente no que diz respeito aos dois pontos seguintes:

i) devido à criação de *clusters* o número de mensagens a circular na rede irá obrigatoriamente diminuir, pois em vez de cada um dos nós enviar a sua própria mensagem e esta ser difundida por toda a rede, ele irá enviar a mensagem ao *Cluster Head* que irá aglutinar toda a informação do *cluster* numa só mensagem, sendo apenas essa mensagem a ser difundida pela rede;

ii) em virtude do comportamento da própria camada de segurança, uma mensagem enviada por um nó órfão (i.e. um nó que não se conseguiu juntar a nenhum *cluster*) não irá originar tantas mensagens difundidas pela rede como nos protocolos de Eschenauer e Q-Composta, onde existe emparelhamento de chaves entre os nós.

No protocolo Q-Composta, devido ao emparelhamento de chaves entre os vários nós da rede, por cada mensagem difundida serão geradas tantas mensagens quanto o número de vizinhos de cada nó, comportamento introduzido pela própria camada de segurança. Um nó que tenha três vizinhos, para enviar uma mensagem para a rede, terá de enviar três mensagens, cada uma delas assinada/cifrada com a chave que partilha com cada um desses vizinhos. Essa atitude irá propiciar um maior consumo energético por parte dos nós dado que há um maior número de mensagens a circular na rede.

Em relação ao Eschenauer, durante os testes efectuados foi possível verificar que o número de ligações definidas era bastante superior ao conseguido no protocolo Q-Composta. Este facto irá originar um consumo energético ainda maior do que o obtido nesse esquema, devido ao comportamento da camada de segurança anteriormente descrito.

Convém ainda salientar que, com o aumento do tamanho da rede, o número médio de *hops* das mensagens que chegam à *base station* vai aumentando progressivamente, o que se reflecte num maior número de reencaminhamentos e, por sua vez, num maior consumo energético.

6.2.2 Análise de fiabilidade

A taxa de fiabilidade numa rede de sensores sem fios diz-nos qual a probabilidade de entrega de mensagens à *base station*. Para este efeito, foram criadas várias topologias de diferentes dimensões e, para cada uma delas, foram testados os esquemas que são alvo de estudo no âmbito desta dissertação. Tal como referido anteriormente, foram seleccionados, de forma aleatória, 20% de nós da rede para gerarem eventos. A taxa de fiabilidade obtida é dada através da verificação do número de mensagens recebidas pela *base station* face ao número total de mensagens enviadas.

No gráfico representado na figura 6.6 é possível observar os resultados obtidos com esta avaliação experimental.

Da análise deste gráfico infere-se que o protocolo com pior desempenho foi o Q-Composta, enquanto que o protocolo que apresentou melhor taxa de fiabilidade foi o SecLEACH. Verifica-se também que, independentemente do protocolo usado, a taxa de fiabilidade diminui à medida que o tamanho da rede aumenta. Este facto deve-se essencialmente às colisões ocorridas durante o reencaminhamento de mensagens entre os nós geradores de eventos e a *base station*.

Um fenómeno interessante e que não passou despercebido foi a fiabilidade apresentada pelo protocolo de Eschenauer ser superior àquela conseguida pela variante que

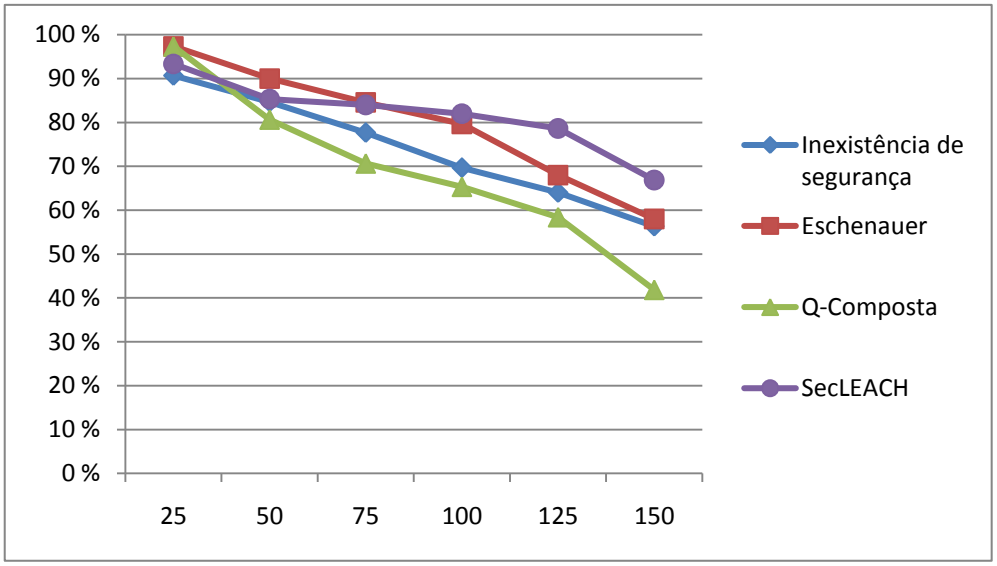


Figura 6.6: Análise de fiabilidade, na fase de operação, para redes de dimensões de 25 até 150 nós

não oferece suporte de segurança. Uma possível razão que pode justificar este acontecimento é o facto de no protocolo de Eschenauer, quando um nó envia uma mensagem, ele está na verdade a enviar n mensagens onde n é o número de vizinhos com quem esse nó conseguiu definir ligações seguras. Este comportamento, introduzido pela camada de segurança, acaba por funcionar como um mecanismo de retransmissão de mensagens, pelo que a probabilidade de uma mensagem se perder acaba por ser inferior à probabilidade encontrada na variante que não oferece segurança, onde cada mensagem é reencaminhada apenas uma única vez.

No caso do esquema Q-Composta, apesar de também existirem retransmissões introduzidas pelo comportamento inerente à camada de segurança, a taxa de fiabilidade foi inferior, o que se poderá justificar pela taxa de cobertura efectiva inferior analisada na secção 6.1.3.

Finalmente, a razão pela qual o SecLEACH apresenta a melhor fiabilidade de todas, prende-se com o facto de que um nó, mesmo que não se consiga juntar a nenhum *cluster*, poderá sempre continuar a enviar os seus eventos, bastando para isso assinar as mensagens com a chave que partilha com a *base station*. Assim sendo, torna-se interessante realizar uma análise específica a este protocolo, onde, de todas as mensagens que chegaram à *base station*, se averigue quais as que foram enviadas por *Cluster Heads* e quais as que foram enviadas por nós órfãos.

Importa ainda realçar que, com o aumento da dimensão da rede, o número médio de *hops* das mensagens que chegam à *base station* vai aumentando progressivamente, pelo que a probabilidade de uma mensagem se perder aumenta, o que se reflecte numa

menor taxa de fiabilidade.

6.2.2.1 Análise de origem de mensagens no SecLEACH

Na linha do que foi referido no parágrafo anterior, revelou-se interessante analisar a origem das mensagens chegadas até à *base station* e determinar quais as que foram enviadas por *Cluster Heads* e quais as que foram enviadas por nós órfãos (i.e. que não faziam parte de nenhum *cluster*). Para tal, foi inserida uma *flag* booleana no campo de metadata das mensagens. Caso a mensagem seja enviada por um CH, a *flag* terá um valor verdadeiro. Caso contrário, terá um valor falso.

Com esta aproximação, obteve-se o gráfico apresentado na figura 6.7.

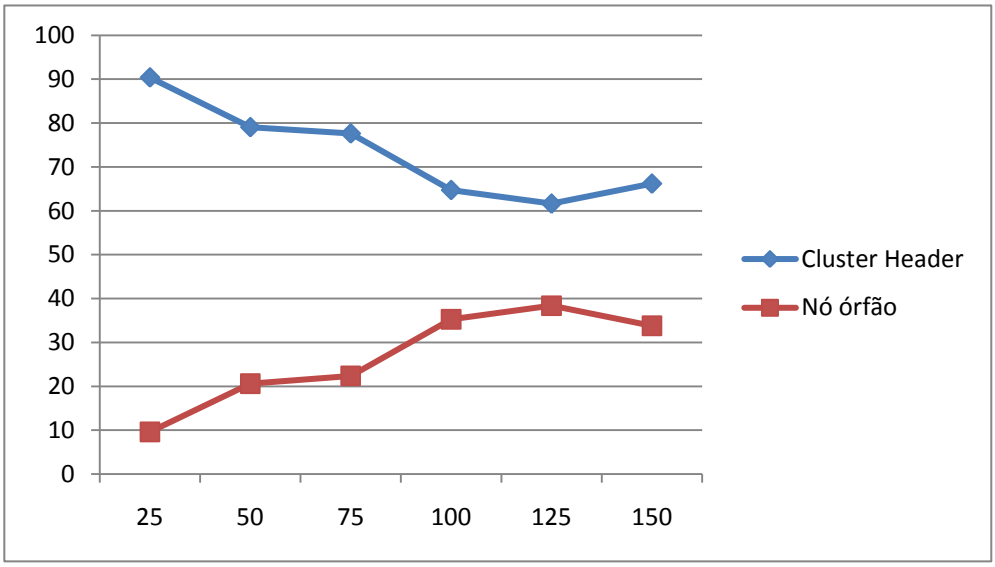


Figura 6.7: Análise de origem de mensagens, na fase de operação com utilização do SecLEACH, para redes de dimensões de 25 até 150 nós

A partir do gráfico acima exposto, é possível verificar que com o aumento do tamanho da rede, a taxa de mensagens enviadas pelos *Cluster Heads* vai diminuindo, dando lugar a cada vez mais mensagens enviadas por nós órfãos. Se um nó, para se dizer coberto, tivesse que pertencer obrigatoriamente a um *cluster*, iria verificar-se uma descida da taxa de cobertura potencial e da taxa de cobertura efectiva com o aumento do tamanho da rede.

6.2.3 Análise de latência

Tendo em conta que as ligações seguras definidas pelo esquema de pré-distribuição de chaves é que definirão a topologia a utilizar pela camada de encaminhamento (e, por

consequência, pela própria aplicação), torna-se igualmente interessante avaliar qual o impacto introduzido na latência do envio de mensagens quando utilizado determinado protocolo. Assim sendo, utilizando o módulo de teste de latências será determinado o tempo médio que uma mensagem leva a percorrer o caminho desde o nó emissor até à *base station*. Para tal, foram geradas topologias com dimensões entre 25 e 150 nós e foram medidas as latências médias obtidas com cada um dos protocolos.

Os resultados conseguidos estão apresentados no gráfico da figura 6.8.

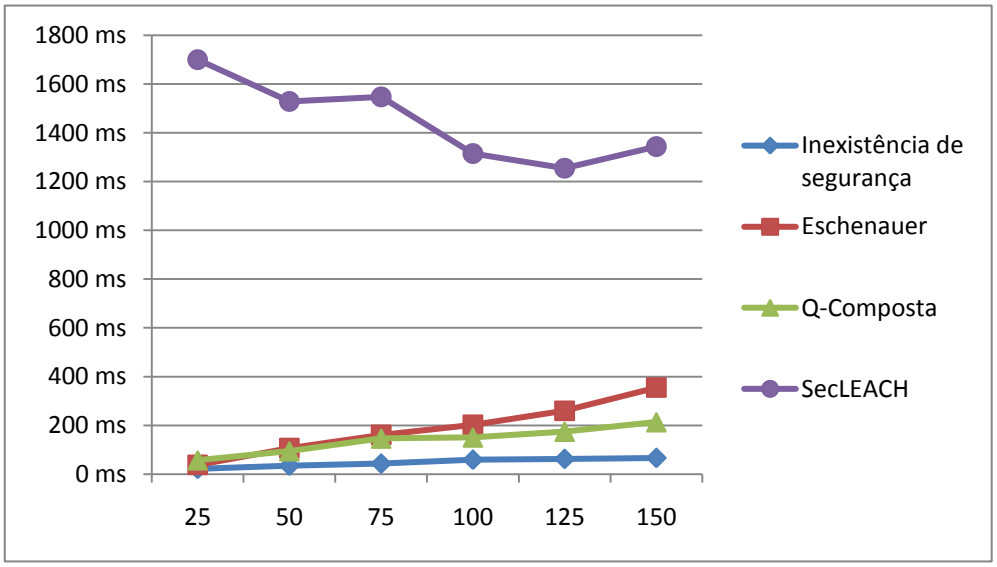


Figura 6.8: Análise de latência, na fase de operação, para redes de dimensões de 25 até 150 nós

Analisando este gráfico, é possível verificar que, como esperado, o resultado mais rápido coube à variante que não utiliza qualquer noção de segurança.

De entre os três esquemas que são alvo de estudo, o Q-Composta apresentou-se como o protocolo que oferece melhores latências. O esquema de Eschenauer apresentou latências ligeiramente superiores às do Q-Composta, sendo que a diferença entre as latências dos dois protocolos se vão distanciando com o aumento gradual do tamanho da rede. O protocolo SecLEACH registou latências bastante mais elevadas que as observadas nos outros dois protocolos.

A razão pela qual o Q-Composta apresenta latências mais baixas que o esquema de Eschenauer poderá dever-se ao facto de o número de ligações seguras definidas neste protocolo ser inferior ao número de ligações definidas no protocolo de Eschenauer. Assim sendo, as filas de mensagens não irão ficar tão cheias, pelo que as mensagens fluirão com maior facilidade pela rede, ao passo que no caso do Eschenauer, devido a um maior número de ligações seguras, um nó irá reter uma mensagem durante mais tempo até a conseguir reencaminhar.

É, ainda, de salientar que, com o aumento do tamanho da rede, o número de *hops* médio das mensagens que chegam à *base station* vai aumentando progressivamente, o que se traduz numa maior latência.

No caso do SecLEACH, os valores apresentados devem-se essencialmente aos *time outs* definidos para os *Cluster Heads* enviarem os eventos que conseguiram agregar até então. Tendo em conta que um CH não sabe ao certo quantos nós daquele *cluster* irão gerar mensagens, foi necessário implementar um temporizador que desse por terminada uma ronda e enviasse os resultados agregados pelo CH até então. Apesar de o valor definido para este temporizador ter sido de 2 segundos, os dados obtidos referentes à latência média apresentaram valores inferiores a esses 2 segundos devido às mensagens enviadas pelos nós órfãos, que não precisam de esperar por mensagens de outros nós e, portanto, assim que o evento é gerado, enviam imediatamente a mensagem, o que fará diminuir um pouco a latência média. Interessa também realçar que ao contrário do que acontece nos outros protocolos, no caso do SecLEACH a latência média diminui progressivamente com o aumento do tamanho da rede. Tal facto está relacionado com o descrito na secção 6.2.2.1 – com o aumento do tamanho da rede, o número de mensagens enviadas por nós órfãos aumenta o que faz com que a latência diminua. É igualmente interessante constatar que o desenho da curva da latência média do SecLEACH é bastante idêntico ao desenho da curva da taxa de mensagens enviadas por CHs, presente no gráfico da figura 6.7, o que confirma a relação entre estes dois indicadores (tipo de emissor e latência).

7

Conclusões

Neste capítulo serão apresentadas as ilacções conseguidas através do desenvolvimento deste trabalho, bem como possíveis propostas para trabalho futuro.

7.1 Conclusões

No âmbito desta dissertação, através de uma análise experimental por simulação, foram avaliados e comparados três esquemas de pré-distribuição de chaves: o esquema de Eschenauer, o esquema Q-Composta e o SecLEACH. Estes esquemas foram especificamente desenhados para redes de sensores sem fios devido às características específicas deste tipo de redes.

As RSSF são redes de comunicação por radiofrequência que utilizam a norma IEEE 802.15.4 ou outras possíveis variantes. São redes auto-organizadas, formadas por dispositivos (sensores) de baixo custo e de pequenas dimensões.

Os sensores são dotados de capacidades computacionais, energéticas e de comunicação muito limitadas, podendo ser distribuídos ao longo de uma determinada área geográfica formando redes de comunicação mais ou menos densas. Estes cooperam, assim, entre si, podendo monitorizar eventos que resultam da interacção entre os dispositivos e o meio ambiente, medindo valores associados a diferentes tipos de fenómenos físicos.

Dado que o funcionamente destas redes ocorre, frequentemente, em ambientes hostis e sem supervisão humana, revelou-se indispensável desenvolver mecanismos de

segurança que protejam este tipo de redes. Os requisitos de segurança podem estar associados a falhas ou ataques que podem ocorrer ao nível das comunicações sem fios, considerando-se as tipologias de ataques semelhantes às conceptualizadas pela Framework X.800 ou modelo de adversário de Dolev-Yao. As possibilidades de ataque alargam-se também a tipologias de ataques em que comportamentos maliciosos são induzidos no processamento dos nós e da rede no seu conjunto, partindo da captura e intrusões ao nível dos nós sensores ou da eventual replicação desses comportamentos através da adição de nós controlados pelo adversário. Por outro lado, os serviços de segurança para RSSF requerem técnicas que se mostrem adequadas às características dos dispositivos que são utilizados como nós dessas redes, atendendo às suas limitações em relação a recursos computacionais, de comunicação e de gestão e consumo energético.

Na literatura disponível, o estudo de esquemas e protocolos de distribuição e estabelecimento de chaves abarcam normalmente análises teóricas de condições de cobertura da rede ou análises de complexidade de custo de comunicação em condições teóricas (em relação ao número de mensagens ou rondas dos protocolos que estabelecem as chaves).

Por isso, revelou-se interessante realizar uma análise de carácter experimental que teve em linha de conta as condições reais de operação de uma RSSF subjacentes à pilha IEEE 802.15.4, mais concretamente, factores que influenciam a operação das redes que estão associados às características dos protocolos de acesso ao meio de comunicação por radiofrequência e a condições ambientais.

Assim, no âmbito desta dissertação, foi realizado um estudo sistemático e comparativo de métodos representativos de opções de distribuição, estabelecimento e refrescamento probabilístico de chaves com base em esquemas de emparelhamento aleatório a partir de pré-distribuição de chaves "setup". Os esquemas alvo de estudo no presente trabalho foram os seguintes: esquema baseado em cobertura aleatória a partir de pré-distribuição de chaves (*Random Key Predistribution Scheme* de Gligor e Eschenauer), esquema baseado em pré-distribuição partilhada aleatória Q-composta (*Shared-Key Threshold R-KPS* de Chan, Perrig e Song) e esquema aleatório para arquitecturas em *cluster* (ou modelo SecLEACH).

Os esquemas acima mencionados foram avaliados e comparados em duas fases distintas: fase de configuração e fase de operação. Nessas fases, foram avaliados os seguintes pontos: avaliação do consumo energético, avaliação de critérios de escala e cobertura da rede, avaliação de critérios de fiabilidade (qual a taxa de pacotes que, após o processo de organização e estabelecimento de chaves iniciais, foram entregues com sucesso) e avaliação de critérios de latência.

No caso do esquema de Eschenauer foi possível observar que a quantidade de ligações seguras estabelecidas foi bastante elevada, o que se revelou ser um factor positivo para garantir boas taxas de cobertura, mas na fase de operação penalizou a rede em termos de consumo energético e latência. Devido ao número elevado de ligações seguras estabelecidas, um nó demora mais tempo para reencaminhar as mensagens (o que se reflecte num aumento de latência). Para além disso, esse elevado número de ligações implica também um maior consumo energético pelo comportamento inerente à camada de suporte de comunicação segura (i.e. um nó, ao pretender enviar uma mensagem, terá de enviar tantas mensagens quanto o número de vizinhos com quem conseguiu estabelecer ligações seguras).

Durante a análise ao esquema Q-Composta foi possível averiguar que o consumo energético durante a fase de configuração foi bastante superior ao consumo energético do esquema de Eschenauer. Isso ficou a dever-se ao facto de haver um maior número de mensagens trocadas entre os nós durante esta fase, pois o número de chaves a descobrir entre eles é superior (no caso dos testes realizados era de três) ao do esquema de Eschenauer (onde basta partilharem apenas uma chave). Foi também possível observar que o número de ligações seguras estabelecidas por este protocolo era inferior ao número de ligações estabelecidas pelo esquema de Eschenauer, o que resultou em taxas de cobertura inferiores às taxas de cobertura conseguidas pelos outros protocolos. Durante a fase de operação foi possível observar que o consumo energético e a latência foram inferiores aos resultados obtidos pelo esquema de Eschenauer.

Finalmente, da análise do protocolo SecLEACH foi possível concluir que este apresenta taxas de cobertura óptimas aliadas a um baixo consumo energético, tanto na fase de configuração como na fase de operação. As boas taxas de cobertura devem-se ao facto de cada nó da rede partilhar uma chave par-a-par com a *base station*, o que garante que mesmo que um nó não se consiga juntar a nenhum *Cluster Head* ele possa continuar a enviar eventos para a rede. O baixo consumo energético apresentado por este protocolo é justificado pela acção dos CHs, que fazem diminuir significativamente o número total de mensagens a circular pela rede, ao agrupar os eventos do seu *cluster*, enviando uma única mensagem contendo toda a informação. No entanto, na fase de operação, a latência apresentada por este protocolo é bastante mais elevada que as latências observadas nos outros protocolos usados. Isso deve-se ao facto de os CHs terem um temporizador que define o período no qual ele espera receber eventos dos membros do *cluster*. Só quando esse período termina é que a mensagem que contém toda a informação agregada é enviada para a *base station*. Foi também interessante observar que, com o aumento do tamanho da rede, o número de mensagens enviadas por nós órfãos vai aumentando progressivamente, o que por sua vez faz com que a

latência diminua, pois um nó órfão envia a mensagem directamente para a *base station* sem sofrer quaisquer atrasos relacionados com os temporizadores dos CHs.

7.2 Aspectos em aberto e trabalho futuro

Como trabalho futuro propõe-se a implementação de uma interface gráfica que possibilite a visualização e interacção com as simulações, à semelhança do que era feito na ferramenta TinyViz, implementada apenas para o ambiente de simulação existente no TinyOS 1.x.

Propõe-se também realizar este tipo de estudos utilizando um outro algoritmo de encaminhamento que não se revele tão pesado para a rede como foi o caso do *flooding*, de forma a obter resultados ainda mais precisos e realistas.

Bibliografia

[1] "Crossbow Technology", Available at http://www.xbow.com/ (June 30, 2010).

[2] IEEE 802.15.4 Standard, Wireless MAC and PHY Specifications for Low Rate Wireless Personal Area Networks (WPANs), IEEE Standard, 802.15 IEEE Group, 2006.

[3] Pelissier, Sylvain (2010). "Cryptography algorithms for TinyOS". Available at http://tinyos.cvs.sourceforge.net/viewvc/tinyos/tinyos-2.x-contrib/crypto/index.html (June 30, 2010).

[4] I.E.T.F., "RFC2828 - Internet Security Glossary". RFC 2828 (Informational), May 2000.

[5] 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. *Comput. Commun.*, 30(7):1655–1695, 2007.

[6] Luis Bernardo, Rodolfo Oliveira, Miguel Pereira, Mário Macedo, and Paulo Pinto. A Wireless Sensor MAC Protocol for bursty data traffic. In *In IEEE PIMRC'07, 18th IEEE Annual International Symposium on Personal Indoor and Mobile Radio Communications*, 2007.

[7] R. Blom. An optimal class of symmetric key generation systems. In *Proc. of the EUROCRYPT 84 workshop on Advances in cryptology: theory and application of cryptographic techniques*, pages 335–338, New York, NY, USA, 1985. Springer-Verlag New York, Inc.

[8] Carlo Blundo, Alfredo De Santis, Amir Herzberg, Shay Kutten, Ugo Vaccaro, and Moti Yung. Perfectly-Secure Key Distribution for Dynamic Conferences. *Lecture Notes in Computer Science*, 740:471–486, 1993.

[9] Michael Buettner, Gary V. Yee, Eric Anderson, and Richard Han. X-MAC: a short preamble MAC protocol for duty-cycled wireless sensor networks. In *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 307–320, New York, NY, USA, 2006. ACM.

[10] 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.

[11] Haowen Chan, A. Perrig, and D. Song. Random key predistribution schemes for sensor networks. In *IN IEEE SYMPOSIUM ON SECURITY AND PRIVACY*, pages 197–213, 2003.

[12] K. Chintalapudi, Tat Fu, Jeongyeup Paek, N. Kothari, S. Rangwala, J. Caffrey, R. Govindan, E. Johnson, and S. Masri. Monitoring Civil Structures with a Wireless Sensor Network. *Internet Computing, IEEE*, 10(2):26–34, 2006.

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

[14] Wenliang Du, Jing Deng, Yunghsiang S. Han, and Pramod K. Varshney. A pairwise key pre-distribution scheme for wireless sensor networks. In *CCS '03: Proceedings of the 10th ACM conference on Computer and communications security*, pages 42–51, New York, NY, USA, 2003. ACM.

[15] D. Eastlake 3rd and P. Jones. US Secure Hash Algorithm 1 (SHA1). RFC 3174 (Informational), September 2001. Updated by RFC 4634.

[16] Laurent Eschenauer and Virgil D. Gligor. A Key-Management Scheme for Distributed Sensor Networks. In *In Proceedings of the 9th ACM Conference on Computer and Communications Security*, pages 41–47. ACM Press, 2002.

[17] L. Evers, M. J. J. Bijl, M. Marin-perianu, R. Marin-perianu, and P. J. M. Havinga. Wireless sensor networks and beyond: A case study on transport and logistics. In *In International Workshop on Wireless Ad-Hoc Networks (IWWAN 2005*, pages 1381–3625, 2005.

[18] David Gay, Philip Levis, Robert von Behren, Matt Welsh, Eric Brewer, and David Culler. The nesC language: A holistic approach to networked embedded systems. In *PLDI '03: Proceedings of the ACM SIGPLAN 2003 conference on Programming language design and implementation*, pages 1–11, New York, NY, USA, 2003. ACM.

[19] Tian He, Sudha Krishnamurthy, Liqian Luo, Ting Yan, Lin Gu, Radu Stoleru, Gang Zhou, Qing Cao, Pascal Vicaire, John A. Stankovic, Tarek F. Abdelzaher, Jonathan Hui, and Bruce Krogh. Vigilnet: An integrated sensor network system for energy-efficient surveillance. *ACM Trans. Sen. Netw.*, 2(1):1–38, February 2006.

[20] ITU. *Security architecture for Open Systems Interconnection for CCITT applications (ITU-T Recommendation X.800)*. International Telecommunications Union, March 1991.

[21] C. Karlof and D. Wagner. Secure routing in wireless sensor networks: attacks and countermeasures. In *Sensor Network Protocols and Applications, 2003. Proceedings of the First IEEE. 2003 IEEE International Workshop on*, pages 113–127, 2003.

[22] Chris Karlof, Naveen Sastry, and David Wagner. TinySec: a link layer security architecture for wireless sensor networks. In *SenSys '04: Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 162–175, New York, NY, USA, 2004. ACM.

[23] Mauri Kuorilehto, Marko Hännikäinen, and Timo D. Hämäläinen. A survey of application distribution in wireless sensor networks. *EURASIP J. Wirel. Commun. Netw.*, 2005(5):774–788, 2005.

[24] Donggang Liu and Peng Ning. Location-based pairwise key establishments for static sensor networks. In *SASN '03: Proceedings of the 1st ACM workshop on Security of ad hoc and sensor networks*, pages 72–82, New York, NY, USA, 2003. ACM.

[25] 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.

[26] Timothée Maret, Raphaël Kummer, Peter Kropf, and Jean-Frédéric Wagen. Freemote emulator: a lightweight and visual java emulator for WSN. In *WWIC'08: Proceedings of the 6th international conference on Wired/wireless internet communications*, pages 92–103, Berlin, Heidelberg, 2008. Springer-Verlag.

[27] Ar Milenkovic, Chris Otto, and Emil Jovanov. Wireless sensor networks for personal health monitoring: Issues and an implementation. *Computer Communications (Special issue: Wireless Sensor Networks: Performance, Reliability, Security, and Beyond*, 29:2521–2533, 2006.

[28] V. B. Misic, J. Fung, and J. Misic. MAC layer security of 802.15.4-compliant networks. In *Proc. 2005 International Workshop on Wireless and Sensor Networks Security WSNS'05*, 2005.

[29] NIST. *Advanced Encryption Standard (AES) (FIPS PUB 197)*. National Institute of Standards and Technology, November 2001.

[30] Leonardo B. Oliveira, Hao C. Wong, M. Bern, Ricardo Dahab, and A. A. F. Loureiro. SecLEACH - A Random Key Distribution Solution for Securing Clustered Sensor Networks. In *NCA '06: Proceedings of the Fifth IEEE International Symposium on Network Computing and Applications*, pages 145–154, Washington, DC, USA, 2006. IEEE Computer Society.

[31] Enrico Perla, Art Ó Catháin, Ricardo Simon Carbajo, Meriel Huggard, and Ciarán Mc Goldrick. PowerTOSSIM z: realistic energy modelling for wireless sensor network environments. In *PM2HW2N '08: Proceedings of the 3nd ACM workshop on Performance monitoring and measurement of heterogeneous wireless and wired networks*, pages 35–42, New York, NY, USA, 2008. ACM.

[32] 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.

[33] Injong Rhee, Ajit Warrier, Mahesh Aia, Jeongki Min, and Mihail L. Sichitiu. Z-MAC: a hybrid MAC for wireless sensor networks. *IEEE/ACM Trans. Netw.*, 16(3):511–524, 2008.

[34] Ana Rito. Redes de Sensores Sem Fios. Master's thesis, Departamento de Informática da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, Departamento de Informática da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, 2006.

[35] Tijs van Dam and Koen Langendoen. An adaptive energy-efficient MAC protocol for wireless sensor networks. In *SenSys '03: Proceedings of the 1st international conference on Embedded networked sensor systems*, pages 171–180, New York, NY, USA, 2003. ACM.

[36] Laura Vanzago. Overview on Wireless Sensor Networks. Master's thesis, Dipartiamento di Scienze del'Informazione, Università degli Studi di Milano, Dipartiamento di Scienze del'Informazione, Università degli Studi di Milano, March 2006.

[37] John Paul Walters, Zhengqiang Liang, Weisong Shi, and Vipin Chaudhary. Wireless sensor network security: A survey, in book chapter of security. In *Distributed, Grid, and Pervasive Computing, Yang Xiao*. CRC Press, 2007.

[38] Arvinderpal S. Wander, Nils Gura, Hans Eberle, Vipul Gupta, and Sheueling Chang Shantz. Energy analysis of public-key cryptography for wireless sensor networks. In *Proceedings of the Third IEEE International Conference on Pervasive Computing and Communications*, pages 324–328, Washington, DC, USA, 2005. IEEE Computer Society.

[39] Y.-C. Wang and Y.-C. Tseng. Attacks and Defenses of Routing Mechanisms in Ad Hoc and Sensor Networks. In *Security in Sensor Networks, Y. Xiao, Auerbach Publications*, 2006.

[40] Geoffrey Werner-Allen, Konrad Lorincz, Matt Welsh, Omar Marcillo, Jeff Johnson, Mario Ruiz, and Jonathan Lees. Deploying a Wireless Sensor Network on an Active Volcano. *IEEE Internet Computing*, 10(2):18–25, 2006.

[41] Y. Xiao. *Security in Sensor Networks*. Auerbach Publications, 2006.

[42] Wei Ye, John Heidemann, and Deborah Estrin. Medium access control with coordinated adaptive sleeping for wireless sensor networks. *IEEE/ACM Trans. Netw.*, 12(3):493–506, 2004.

[43] Wei Ye, Fabio Silva, and John Heidemann. Ultra-low duty cycle MAC with scheduled channel polling. In *SenSys '06: Proceedings of the 4th international conference on Embedded networked sensor systems*, pages 321–334, New York, NY, USA, 2006. ACM.

[44] H. Água. Protocolo MAC para acesso multi-modo em redes de sensores sem fios móveis. Master's thesis, Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, Departamento de Engenharia Electrotécnica e de Computadores da Faculdade de Ciências e Tecnologia da Universidade Nova de Lisboa, 2008.