SOS: Sensoriamento Overlay Seguro em Redes de Sensores Sem Fio Hierárquicas

Leonardo B. Oliveira[1], Antonio A. F. Loureiro[2], Ricardo Dahab[1], Hao Chi Wong[3]

[1] Instituto de Computação – UNICAMP
Campinas, SP

[2]Departamento de Ciência da Computação – UFMG
Belo Horizonte, MG

[3]Palo Alto Research Center (PARC)
Palo Alto, CA, EUA

{leob,rdahab}@ic.unicamp.br,loureiro@dcc.ufmg.br,hcwong@parc.com

Resumo. *Este artigo apresenta o Sensoriamento Overlay Seguro (SOS). O SOS constrói uma Rede Overlay (RO) sobre uma Rede de Sensor Sem Fio (RSSF). Ao estabelecer e monitorar rotas alternativas, o SOS é capaz de encontrar rotas mais seguras que as fornecidas pelo protocolo de roteamento padrão. Os resultados indicam que o SOS é: 1) eficaz no aumento da taxa de entrega de mensagens em redes sob ataques de negação de serviço e 2) eficiente em termos de consumo de energia. Até onde sabemos, o SOS é o primeiro mecanismo de segurança baseado em ROs para RSSFs.*

Abstract. *In this paper, we present Secure Overlay Sensornets (SOS). SOS builds an Overlay Network (ON) over a sensornet, and it establishes and monitors alternative overlay routes. By doing so, SOS is able to find out routes more secure than routes provided by the default routing protocol. Our results indicate that SOS improves the delivery ratio in scenarios under DoS attacks and that it is efficient in terms of energy consumption. To our knowledge, SOS is the first security mechanism based on ONs for sensornets.*

1. Introdução

Redes *overlay* (ROs) são redes construídas sobre redes físicas convencionais, cuja função é migrar parte da complexidade de roteamento para a camada de aplicação [Andersen et al. 2001]. Baseadas em um dado critério, ROs são capazes de monitorar a rede e fornecer caminhos alternativos ao usuário. Tais caminhos são construídos através da atuação de nós *overlay* como intermediários no envio de dados.

Já Redes de Sensores Sem Fio (RSSFs) [Estrin et al. 1999] são redes *ad hoc* compostas basicamente por pequenos nós sensores de recursos extremamente limitados (pouca energia, largura de banda, capacidade computacional etc.) e uma ou mais estações rádio base (ERBs). Tais redes podem ser utilizadas para diferentes aplicações, tais como operações de resgate em áreas de conflito/desastre e detecção de exploração ilegal de recursos naturais.

Como qualquer outro tipo de rede *ad hoc* sem fio, RSSFs são vulneráveis a ataques [Karlof and Wagner 2003, Wood and Stankovic 2002]. Porém, além das vulnerabilidades já existentes na comunicação sem fio de redes *ad hoc* em geral, RSSFs enfrentam problemas adicionais. Elas comumente são dispostas em ambientes abertos, muitas vezes

hostis, o que as torna fisicamente acessíveis a adversários. Não obstante, nós sensores são mais escassos de recursos que nós de redes *ad hoc* (o nó sensor Mica2 Motes [Hill and Culler 2002], por exemplo, possui um processador de 7.8 MHz e 4 KB de memória RAM), e as soluções convencionais não lhes são aplicáveis. Por exemplo, o fato de que nós sensores devem ser descartáveis e, por conseguinte, de baixo custo, torna pouco viável equipá-los com dispositivos contra violação (*tampering*). Além disso, o baixo poder computacional dos sensores torna inviável a utilização de algoritmos criptográficos assimétricos (RSA, por exemplo) em todos os cenários [1]. Logo, dotar RSSFs de segurança é uma tarefa especialmente desafiadora e essencial em aplicações que demandem sigilo, autenticação, privacidade etc.

O objetivo deste trabalho é avaliar o emprego de ROs para aumentar a segurança de RSSFs hierárquicas e heterogêneas. Mais detalhadamente, ao empregar redes ROs sobre RSSFs, esperamos aumentar a taxa de entrega de mensagens contendo informação sensível à ERB em ambientes sob ataques de negação de serviço (*Denial of Service* – DoS). Para tal, propomos o **S**ensoriamento **O**verlay **S**eguro (SOS). Até onde sabemos, o SOS é o primeiro mecanismo de segurança para RSSFs baseado em ROs. Em outras palavras, ele é o primeiro mecanismo que efetua uma monitoração efetiva da rede à procura de rotas alternativas mais seguras. Além disso, como iremos ver na Seção 5, este é o primeiro mecanismo que tira proveito do fato de que existem mensagens mais importantes que outras. Nosso especial interesse por redes hierárquicas e heterogêneas deve-se ao seu melhor custo-benefício em termos de energia [Melo and Liu 2002] e devido ao número reduzido de trabalhos de segurança especialmente voltados para tais organizações.

O restante deste documento está organizado da seguinte maneira. Na Seção 2 discutimos os trabalhos relacionados. Em seguida, na Seção 3, discutimos a organização e a segurança de RSSFs hierárquicas. Descrevemos o modelo de rede considerado na Seção 4. Na Seção 5 apresentamos nossa solução. Na Seção 6 descrevemos como foi conduzida a simulação e a forma com a qual analisamos os resultados, os quais são apresentamos em seguida na Seção 7. Finalmente, na Seção 8 concluímos o trabalho.

2. Trabalhos Relacionados

Propostas de roteamento seguro para redes *ad hoc* em geral ([Zhou and Haas 1999, Papadimitratos and Haas 2002, Capkun and Hubaux 2003], por exemplo) não são adequadas para RSSFs, pois ou contam com criptografia de chave pública, ou não levam em consideração o roteamento assimétrico de "muitos pra um" de uma RSSF. Isso fez com que surgissem um número considerável de propostas voltadas exclusivamente para RSSFs (por exemplo, [Eschenauer and Gligor 2002, Zhu et al. 2003, Liu et al. 2005, Du et al. 2005, de Oliveira et al. 2005]), a maior parte delas sem se ater a algum tipo particular de organização.

Staddon *et al.*, por exemplo, propuseram um algoritmo eficiente para identificar nós que falharam [Staddon et al. 2002] e desviar a informação para rotas alternativas. Algumas das idéias do trabalho, como a descoberta da vizinhança por nós e o envio desta informação à ERB, foram também empregadas pelo SOS. Já Perrig *et al.* [Perrig et al. 2002] apresentaram SPINS, um conjunto de protocolos baseados em chaves simétricas para fornecer primitivas de segurança (sigilo, autenticação, integridade etc.) durante o roteamento.

[1] Note-se que recentemente surgiram implementações eficientes de algoritmos criptográficos assimétricos baseados em curvas elípticas [Gura et al. 2004], mas eles ainda são ordens de grandeza mais caros que algoritmos simétricos

Um subconjunto dos trabalhos busca o aumento de segurança e/ou confiabilidade através do emprego de rotas múltiplas. Ganesan *et al.* [Ganesan et al. 2001] propuseram um versão de rotas múltiplas para o protocolo de roteamento Directed Diffusion [Intanagonwiwat et al. 2000]. Eventualmente, mensagens redundantes são enviadas através de rotas alternativas para verificar se essas rotas continuam operacionais e, ante uma falha na rota padrão, elas são utilizadas. A idéia do trabalho é aumentar a tolerância a falhas do Directed Diffusion e ele não trata o problema de nós comprometidos. Deng *et al.* [Deng et al. 2003] propuseram o INSENS, em que um nó sempre envia uma mesma mensagem por mais de uma rota . Boukerche *et al.* [Boukerche et al. 2004] propuseram PEQ, um protocolo de detecção e reparo de falhas em rotas baseado em ACKs. Diferentemente de outros protocolos que utilizam três transmissões (*three way protocols*), empregando o PEQ, um nó é capaz de escolher uma nova rota apenas com informações a respeito do caminho mínimo entre seus vizinhos e a ERB. Lou *et al.* [Lou et al. 2004] transformam um mensagem em múltiplos compartilhamentos (*shares*) através de técnicas de compartilhamento de segredos [Shamir 1979]. Cada compartilhamento é enviado via rotas independentes, de forma que se um pequeno número de nós ao longo das rotas estiverem comprometidos, o segredo da mensagem como um todo permanece confidencial. Note-se que todos esses trabalhos empregam em algum grau redundância para entrega de mensagens. Apesar de aumentar a chance da mensagem alcançar o destinatário, a redundância ocasiona aumento no consumo de energia. Além disso, nenhum deles efetua uma monitoração efetiva das rotas alternativas, como é feito em uma RO. Finalmente, técnicas de classificação de mensagens não são utilizadas e uma mensagem contendo informação sensível possui a mesma chance de ser comprometida que uma mensagem comum.

Há também trabalhos exclusivamente voltados para RSSFs hierárquicas. Ambos os trabalhos de Kong *et al.* [Kong et al. 2002] e de Bohge e Trappe [Bohge and Trappe 2003] apresentam soluções para redes hierárquicas e heterogêneas. Contudo, eles presumem nós muito poderosos, capazes de executar algoritmos de chave pública. Mas, recentemente, surgiram protocolos baseados exclusivamente em esquemas de chaves simétricas [Oliveira et al. 2005b,Ferreira et al. 2005]. No trabalho [Oliveira et al. 2005b], batizado de LHA-SP, um conjunto de protocolos de segurança para RSSFs hierárquicas é apresentado. O LHA-SP mescla chaves de grupo e par-a-par para fornecer autenticação e sigilo durante a configuração, operação e manutenção da rede. O mesmo grupo de pesquisa também propôs duas extensões de segurança para o protocolo LEACH [Heinzelman et al. 2000], são elas SLEACH [Ferreira et al. 2005] e SecLEACH [Oliveira et al. 2005a, Oliveira et al. 2006]. A primeira utiliza μTESLA [Perrig et al. 2002] e chaves compartilhadas par-a-par entre os nós e a ERB para realizar o agrupamento e a fusão de dados de forma autenticada. Já SecLEACH emprega chaves aleatórias para configurar e operar a rede de forma autenticada e sigilosa.

3. RSSFs Hierárquicas: Organização e Segurança

RSSFs podem ser organizadas de diferentes maneiras. Em RSSFs *planas* [Akyildiz et al. 2002], todos os nós possuem papéis semelhantes no sensoriamento, processamento de dados e roteamento. Em particular, todos os nós operam com raio de transmissão limitado para poupar energia e a comunicação nós→ERB, em função disto, é *multi-hop*, com nós exercendo o papel de roteadores uns para os outros. Em RSSFs *hierárquicas* [Estrin et al. 1999], por outro lado, a rede é em geral organizada em grupos (*clusters*), em que líderes (CHs – *clusters-heads*) e membros comuns de grupos exercem diferentes papéis. Enquanto membros comuns são responsáveis pelo sensoriamento, CHs são responsáveis por tarefas adicionais, tais como reunir e processar o dado sensoriado pelos demais membros do grupo, e repassar (*forward*) os resultados para a ERB.

Como qualquer RSSFs, as hierárquicas são vulneráveis a um grande número de ataques [Karlof and Wagner 2003, Wood and Stankovic 2002], incluindo interferência (*jamming*), personificação (*spoofing*), retransmissão (*replay*) e violação (*tampering*). Nessas redes, ataques envolvendo CHs são os mais devastadores, já que eles são responsáveis pelas funções mais importantes, como fusão de dados e roteamento.

Caso um adversário planeje se tornar um CH, ele pode efetuar ataques como o buraco negro (*blackhole* [Karlof and Wagner 2003]) e o repasse seletivo (*selective forwarding* [Marti et al. 2000]) e, potencialmente, causar danos a grandes frações da rede. Outra opção, obviamente, é não interferir no roteamento e tentar prejudicar o resultado do sensoriamento injetando dados espúrios na rede. O adversário pode também bisbilhotar (*eavesdrop*) a comunicação entre nós legítimos a fim de obter a leitura dos dados efetuada pelos demais sensores.

4. Modelo de Rede

Neste trabalho consideramos redes hierárquicas e heterogêneas em que os nós são estáticos.

A comunicação funciona da seguinte forma. Entre filhos e CHs é *single-hop*, e entre CHs em direção à ERB *multi-hop* – ou seja, de CHs para CHs até que se alcance a ERB. Já a ERB alcança todos os nós diretamente e a comunicação ERB em direção aos nós é *single-hop*.

O sensoriamento é dirigido a relógio, ou seja, nós reportam dados coletados do ambiente após intervalos fixos de tempo.

A densidade da rede é tal que cada CH possui vizinhos que possam ser usados como rotas alternativas para o envio e repasse de mensagens.

Existem duas classes de mensagens: as prioritárias e as não prioritárias. Entende-se por mensagens não prioritárias aquelas que, caso sejam perdidas, não representem grandes danos ao funcionamento da rede. Já mensagens prioritárias carregam consigo informação sensível. Essa classificação ocorre entre a coleta de um dado e seu envio à ERB. Ela é efetuada pelo próprio nó sensor que a coletou, o qual se baseia na importância do dado coletado.

Finalmente, ERB é confiável e a rede é susceptível aos seguintes ataques de DoS: buraco negro e repasse seletivo.

5. SOS

Nesta seção apresentamos o SOS, um mecanismo de **S**ensoriamento **O**verlay **S**eguro para RSSFs. Nosso objetivo é fazer as mensagens prioritárias trafegarem por rotas mais seguras e, por sua vez, aumentar sua taxa de entrega à ERB. Primeiramente, damos uma visão geral do protocolo (Seção 5.1) e, logo em seguida, o apresentamos com mais detalhes (Seção 5.2).

5.1. Visão Geral

O SOS constrói uma RO sobre uma RSSF da seguinte forma. Alguns ou a totalidade do nós da RSSF subjacente são escolhidos para integrar a RO. A cada um desses nós, chamados de nós *overlay*, uma lista de vizinhos lógicos lhes é atribuída. Esses vizinhos, chamados de vizinhos *overlay*, são usados como intermediários em rotas alternativas à rota fornecida pelo protocolo padrão de roteamento (rota padrão). Os nós *overlay* então alternam igualmente o envio de mensagens não prioritárias por essas rotas, incluindo

a padrão. Além do usual objetivo de carregar dados sensoriados, essas mensagens são usadas para monitorar a taxa de entrega das rotas. A ERB então contabiliza as taxas de entrega baseada nas últimas t unidades de tempo. As rotas com taxas de entrega maiores são, por motivos óbvios, consideradas *melhores* (mais seguras) e a ERB informa aos nós *overlay* quais são elas. Assim, na hora de enviar mensagens prioritárias, os nós *overlay* utilizam suas respectivas melhores rotas a fim de potencializar a chance de entrega das mesmas.

A Figura 1 apresenta um diagrama de uma RO sobre uma RSSF. Observe que o nó SRC conta com rotas *overlay* – além da rota padrão – para o envio de dados.

5.2. Descrição do Protocolo

A priori, considera-se que os nós sensores foram lançados, que a granularidade de envio de mensagens foi especificada e que o protocolo de roteamento padrão foi estabelecido.

O primeiro passo para a construção da RO é a ERB identificar o grafo de conectividade da rede. Para tal, os nós podem, por exemplo, enviar um *broadcast* a procura de vizinhos [Staddon et al. 2002]. Ao receber a respostas dos vizinhos, o nó então repassa essa informação para a ERB. Esta última, de posse da lista de vizinhos de cada nó, constrói o grafo.

A ERB então determina quais nós farão parte da RO. Para cada um deles, ela também atribui uma lista de vizinhos *overlay*, os quais são escolhidos dentre os demais integrantes da RO levando-se em conta informações do grafo de conectividade. Esses vizinhos são utilizados como intermediários no envio de dados por rotas alternativas, chamadas de rotas *overlay*. Em outras palavras, cada vizinho *overlay* corresponde ao primeiro *hop* de uma rota *alternativa*. Na Figura 1, vizinhos *overlay* são aqueles que compartilham enlaces virtuais. Note-se que o critério de escolha dos vizinhos *overlay* pode variar dependendo da aplicação de sensoriamento. Contudo, visto que na maioria das vezes RSSFs tentam minimizar o consumo de energia – e este aumenta de forma quadrática com o raio de comunicação [Akyildiz et al. 2002] –, acredita-se que o vizinho *overlay* de um dado nó deve ser escolhido dentre os nós em sua vizinhança.

Em seguida, o sensoriamento é iniciado e nós sensores passam a enviar mensagem prioritárias e não prioritárias, de acordo com a relevância da informação coletada.

As mensagens não prioritárias são empregadas tanto para enviar dados, como para monitorar as rotas. Os nós *overlay* então alternam o envio dessas mensagens entre a rota padrão e as demais rotas *overlay* de maneira justa (*round robin*). Já as mensagens prioritárias, enquanto não se conhece as melhores rotas (ou seja, os primeiros resultados da monitoração ainda não foram divulgados), são enviadas e repassadas pela rota padrão. Dois *flags* são adicionados a cada mensagem pelo nó remetente: 1) um que indica a classe da mensagem, isto é, se ela é ou não prioritária e 2) outro que indica por qual rota (padrão, *overlay* #1, *overlay* #2,...) a mensagem foi enviada. Tais *flags* são necessários para que, posteriormente, a ERB possa calcular a taxa de entrega das rotas.

Assim que a ERB recebe as primeiras mensagens ela cria um registro. O registro é organizado por nó e por rota e, para cada rota de um determinado nó, existe a taxa de entrega de mensagens não prioritárias. A taxa de entrega é calculada levando em consideração o número de mensagens enviadas e recebidas nas últimas t unidades de tempo. A ERB infere o número de mensagens enviadas baseada na granularidade de envio de dados e no fato de que este envio é alternado igualmente por cada rota. O cálculo do número de mensagens recebidas é, claramente, mais fácil e basta a ERB identificar o nó

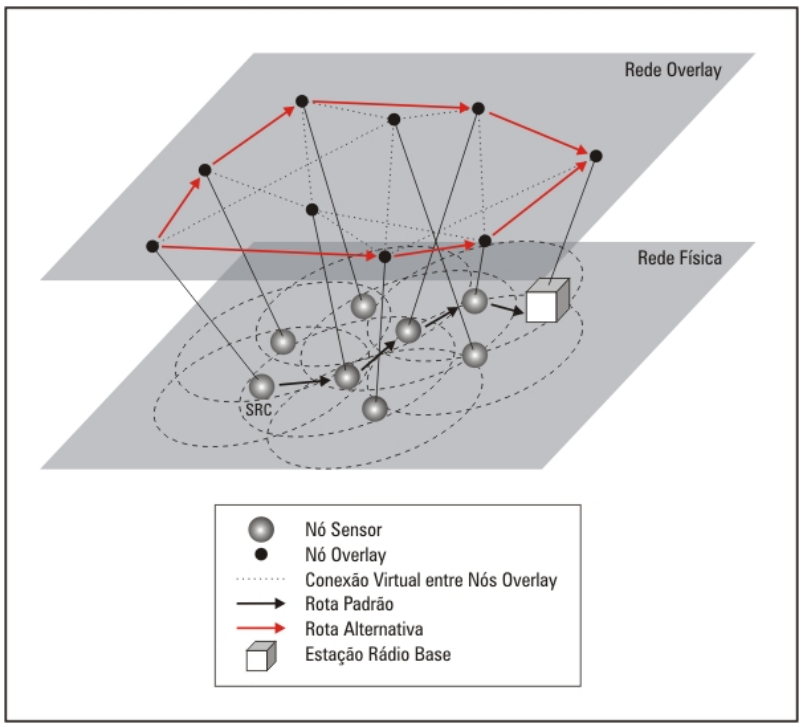


Figura 1: Diagrama de uma RO sobre uma RSSF

nó sensor	taxa de entrega de rotas		
	padrão	overlay #1	overlay #2
1	76%	54%	65%
2	70%	78%	73%
3	31%	50%	39%
4	55%	82%	97%
5	83%	89%	42%

Tabela 1: Registro para uma rede de 5 nós e 2 rotas overlay pra cada um. É mostrado a taxa de entrega das rotas e as melhores rotas estão em destaque.

remetente (campo *source* da mensagem) e a rota utilizada (*flag* da mensagem). Note-se que mensagens prioritárias não são levadas em conta nessa contabilidade, já que não são enviadas alternadamente entre as rotas. A Tabela 1 mostra um registro para uma rede fictícia de 5 nós em que 2 rotas *overlay* por nó são utilizadas. As rotas em destaque são as melhores rotas.

A ERB então informa aos nós *overlay* quais as melhores rotas. Essas rotas são então usadas pelos nós para o envio e repasse de futuras mensagens prioritárias, tendo em vista seu aumento de taxa de entrega. O procedimento quanto as mensagens não prioritárias, contudo, não muda (Tabela 2), em função do objetivo contínuo de monitoração das rotas.

Note-se que com o decorrer do tempo, pode ser que para alguns nós a melhor rota mude. Neste caso, a ERB re-informa aos nós em questão sobre suas respectivas novas melhores rotas. Note-se também que para nenhum tipo de mensagem ocorre replicação, ou seja, o SOS não emprega redundância para garantir entrega de mensagens.

É importante salientar que em RSSFs as soluções de segurança são bem específicas [Wood and Stankovic 2002] e não existe uma panacéia para todos os problemas. No caso

classe da mensagem	Inicialmente		Após divulgação de melhor rota	
	envio	repasse	envio	repasse
prioritária	padrão	padrão	melhor rota	melhor rota
não prioritária	alternado	padrão	alternado	padrão

Tabela 2: Resumo de políticas de envio e repasse no SOS

do SOS, em particular, existem essencialmente dois aspectos de segurança que devem ser discutidos. Em primeiro lugar, adversários podem utilizar a informação de qual a melhor rota para inferir por onde futuras mensagens prioritárias trafegarão e, assim, tê-las como principal foco de comprometimento. Segundo, eles podem se beneficiar do *flag* que identifica a classe das mensagens. Neste caso, por exemplo, um ataque de repasse seletivo poderia optar por descartar apenas mensagens prioritárias e repassar somente as não. Esses problemas são decorrentes principalmente da comunicação não sigilosa e não autenticada entre os nós. Todavia, uma grande gama de trabalhos ([Eschenauer and Gligor 2002, Zhu et al. 2003, Liu et al. 2005, Du et al. 2005, de Oliveira et al. 2005], por exemplo) oferecem soluções criptográficas efetivas para sanar tais vulnerabilidades e o SOS deve ser empregado em conjunto com tais soluções.

Outra coisa a ser dita é que o envio de mensagens não prioritárias através de diferentes rotas pode causar uma quebra no sequenciamento das mensagens recebidas pela ERB. Vale lembrar, contudo, que: 1) essas mensagens não são prioritárias; 2) em RSSFs os dados coletados são simples e geralmente enviados em uma única mensagem, o que torna o sequenciamento menos importante. Se ainda assim o sequenciamento for necessário, pode-se embutir identificadores nas mensagens e reordená-las na ERB antes que seus dados sejam processados.

6. Metodologia de Avaliação

Para avaliar o SOS, comparamos uma mesma aplicação de sensoriamento em RSSFs com e sem o SOS. Para isso, implementamos a aplicação e o SOS no simulador de redes ns-2 (*Network Simulator*) [Fall and Varadhan 2001]. A escolha pelo simulador deve-se ao fato do mesmo ser amplamente utilizado pela comunidade científica. Nesta seção, descrevemos como as simulações foram conduzidas e apresentamos os parâmetros e métricas considerados na avaliação.

Em nossa aplicação, os nós foram lançados de maneira aleatória na área a ser sensoriada. Para o SOS, particularmente, consideramos que mecanismos para troca de chaves simétricas entre vizinhos (LEAP [Zhu et al. 2003], por exemplo) e para agrupamento de nós comuns em torno dos CHs próximos já haviam sido efetuados (LEACH [Heinzelman et al. 2000], por exemplo).

Para cada nó sensor, a rota padrão que o liga à ERB foi computada através de um protocolo similar ao TinyOS beaconing [Levis et al. 2004], que constrói uma árvore com raiz na ERB a partir de uma busca em largura.

Nesta instância do SOS, com exceção dos CHs que alcançavam diretamente a ERB, todos os CHs faziam parte da RO. Os nós comuns não foram incluídos na RO por motivos de eficiência, ou seja, para poupar a ERB do ônus de disseminar o estado das rotas para toda a rede e poupar os nós comuns de receber mensagens sobre o estado das rotas. Dois vizinhos alternativos foram atribuídos para cada nó overlay X. Esses vizinhos foram escolhidos dentre os vizinhos físicos de cada nó, com exceção dos que já participavam da rota padrão. Além disso, para evitar *loops*, o vizinho escolhido não

poderia ter como primeiro *hop* da rota padrão o próprio nó X. Note-se, contudo, que ainda sim poderiam haver *loops*. Logo, para evitar-se que uma mensagem fosse tratada indefinidamente, implementamos um esquema em que os nós não repassavam mensagens que já haviam sido tratadas recentemente.

O raio de transmissão dos nós foi configurado para $100m$ e o envio de mensagens de um nó comum para seu CH era feito a cada $10s$. Este último então agregava a informação dos diferentes filhos em uma só mensagem e a repassava para a ERB através dos vizinhos.

Os tamanhos de mensagens para as redes com e sem o SOS foram de 36 (tamanho padrão do TinyOS [Levis et al. 2004]) e 30 bytes, respectivamente. Essa diferença deve-se à introdução de um código de autenticação de mensagem (*Message Authentication Code* – MAC). Na verdade, RSSFs em geral utilizam chaves de 64 bits e MACs, portanto, são de 8 bytes [Perrig et al. 2002]. Contudo, sua introdução dispensa a utilização de código de redundância cíclico (*Cycle Redundancy Check* – CRC), de 2 bytes em RSSFs [Levis et al. 2004]. Note-se que por motivos práticos, não levamos em consideração os *flags*, já que seriam necessários apenas 3 bits (dois para identificação das rotas e um para classificação de mensagens) no cenário com 2 vizinhos alternativos por nó *overlay*. Todavia, este custo deve ser levado em consideração caso o SOS seja empregado com um número maior de rotas alternativas.

Para estimar-se o consumo de energia, empregou-se o mesmo modelo utilizado em [Heinzelman et al. 2000]. Neste modelo, o rádio dissipa ϵ_r = $50\mathrm{nJ/bit}$ para executar o transmissor ou receptor, e ϵ_a = $100\mathrm{pJ/bit/m^2}$ para o amplificador do transmissor. Além disso, o rádio gasta o mínimo de energia necessário para alcançar os destinatários e são desligados para evitar recebimento de transmissões a eles não destinadas. Finalmente, a perda de energia nas transmissões é proporcional ao quadrado da distância. Assim, os custos para transmitir ($\mathcal{E}_T$) e receber ($\mathcal{E}_R$) uma mensagem de β bits a uma distância δ são dados, respectivamente, por:

$$\mathcal{E}_T(\beta, \delta) = \beta\,\epsilon_r + \beta\,\delta^2\,\epsilon_a$$
$$\mathcal{E}_R(\beta) = \beta\,\epsilon_r$$

Para todos os parâmetros foram considerados 2 tipos de ataques de DoS. São eles o buraco negro e o repasse seletivo. No primeiro ataque, o nó comprometido simplesmente some com toda e qualquer mensagem enviada a ele. No segundo, repasse seletivo, o nó deixa de repassar apenas um percentual das mensagens. Em nossas simulações, sob quaisquer ataques os nós também paravam de gerar dados de sensoriamento.

Os ataques eram disparados logo após os nós serem lançados e se organizarem em grupos. Os nós atacados foram escolhidos de forma aleatória entre os CHs, já que, como foi discutido na Seção 3, ataques a esses nós resultam em maiores estragos ao funcionamento da rede. É verdade que um ataque concentrado perto da ERB também seria interessante para adversários. Entretanto, optamos por deixá-los com a mesma probabilidade de ocorrer que os demais, pois, justamente por focarem localidades próximas à ERB, são mais difíceis de ser executados [2]. Em relação aos ataques de repasse seletivo, em particular, a taxa de descarte empregada foi de 50%.

[2]Para violar um nó próximo a ERB, um adversário teria que chegar muito próximo a ela, e poderia ser facilmente flagrado, até mesmo a olho nu.

Finalmente, cada simulação durou $1000s$ e foi executada 33 vezes, sendo que cada execução alimentou o gerador de números aleatórios do simulador com uma semente distinta das demais. Os resultados obtidos representam a média das 33 execuções.

6.1. Parâmetros

Durante a simulação, os seguintes parâmetros foram variados:

1. **Percentual de mensagens prioritárias**: variamos o percentual de mensagens prioritárias em relação ao número total de mensagens. São elas 25%, 50% e 75%. O valor padrão deste parâmetro também foi 25%, uma vez que acreditamos que mensagens prioritárias ocorrem com menor freqüência.
2. **Percentual de CHs comprometidos**: variamos o percentual de CHs comprometidos. A taxas de comprometimento foram de 25%, 50% e 75%. O valor padrão deste parâmetro foi 25%, já que acreditamos que percentuais menores de comprometimento são mais freqüentes.
3. **Tamanho da rede**: com o objetivo de avaliarmos a escalabilidade da solução geramos resultados com três tamanhos distintos de redes, são eles 1100 (100 CHs e 1000 nós comuns), 2200 (200 CHs e 2000 nós comuns) e 3300 (300 CHs e 3000 nós comuns) nós. A fim de isolarmos o efeito do tamanho da rede nos resultados optamos por manter a densidade da rede, ou seja, à medida que o número de nós aumentava, ajustávamos proporcionalmente o tamanho da região sensoriada. O valor padrão do tamanho da rede e da área em que os nós foram lançados foram 1100 nós e $330\times330m^2$, respectivamente. A escolha dos 1100 nós como padrão foi devido às restrições computacionais para se executar o simulador. Já o tamanho da região sensoriada foi calculada a partir do número de nós para que obtivéssemos um densidade próxima a do trabalho em [Heinzelman et al. 2000].

6.2. Métricas

Também escolhemos métricas para mensurar os ganhos, custos e eficiência do SOS. Tais métricas são discutidas abaixo.

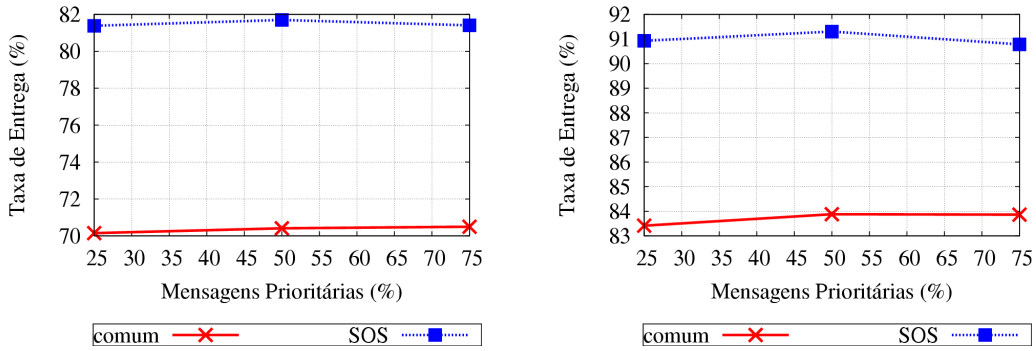
- **Taxa de entrega (ganho)**: como descrito anteriormente (Seção 6), o objetivo do SOS é aumentar a taxa de entrega de mensagens prioritárias à ERB. Com isso, o ganho da solução será dado em termos desta taxa de entrega. Observe que a taxa de entrega de mensagens não prioritárias não é melhorada com o emprego do SOS, pois seu envio é alternado de maneira justa dentre três possibilidades.
- **Energia (custo)**: a energia é o recurso mais escasso de um nó sensor e, por conseguinte, é geralmente escolhida como medida de custo de uma proposta. A diferença em termos de consumo de energia de uma rede com e sem o SOS pode ser oriunda de 5 fatores. Em primeiro lugar está o fato do SOS possibilitar que um maior número de mensagens cheguem até a ERB, o que causa um aumento de tráfego e, conseqüentemente, um aumento natural do consumo de energia. Segundo, no SOS de tempos em tempos CHs recebem notificações da ERB sobre qual a melhor rota. Este recebimento por parte destes nós também acarreta em consumo de energia. Note-se, contudo, que CHs são um percentual pequeno da rede, e isso não deve influenciar muito o consumo total. Terceiro, apesar das rotas alternativas serem escolhidas a partir de vizinhos, pode ser que o vizinho escolhido esteja um *hop* mais distante da ERB. Com isso, o caminho médio das rotas aumenta e com ele o consumo de energia. Em quarto lugar, está o aumento do tamanho da mensagem, devido principalmente a adição de um MAC, como visto acima, nesta mesma seção. Em último lugar, estão gastos com processamento de algoritmos criptográficos. Porém, não os levamos em consideração, uma vez que

já foi mostrado que este custo, para alguns algoritmos simétricos, é ínfimo [Perrig et al. 2002].

- **Gasto energético por mensagem entregue (eficiência)**: Uma vez descritas as métricas de ganho e custo da solução é possível estabelecer uma para a eficiência. No SOS, a eficiência é dada pelo custo energético por mensagem entregue.

7. Resultados

Nesta seção são apresentados os resultados de simulação. Como descrito na Seção 6, comparamos uma RSSF que executava apenas uma aplicação de sensoriamento, com outra que executava sensoriamento e o SOS juntos – para facilitar, as chamamos de redes *comum* e SOS, respectivamente. Na comparação foram considerados os 3 parâmetros: 1) percentual de mensagem prioritárias, 2) percentual de nós comprometidos e 3) tamanho da rede. Os valores de custos são apresentados em tabelas, sob forma de *overhead* de energia. Para cada ataque, esses valores são discriminados por CH (**CH**), pela rede como um todo (**geral** – inclui ambos os CHs e membros comuns de grupos), e por mensagem prioritária entregue (**por mensagem**). Note-se que este último também mede a eficiência do SOS. Além disso, como apenas os CHs faziam parte da RO (como vimos na Seção 6), o *overhead* de energia para os membros comuns de grupo foi essencialmente causado pela adição dos MACs e ficou em torno de 20% para todos os cenários. Por este motivo, optamos por omiti-lo nas tabelas. Finalmente, para o cálculo dos custos de energia foram considerados apenas os nós não comprometidos.



(a) **buraco negro** (b) **repasse seletivo**

Figura 2: Taxa de entrega de mensagens prioritárias em função do percentual de mensagens prioritárias

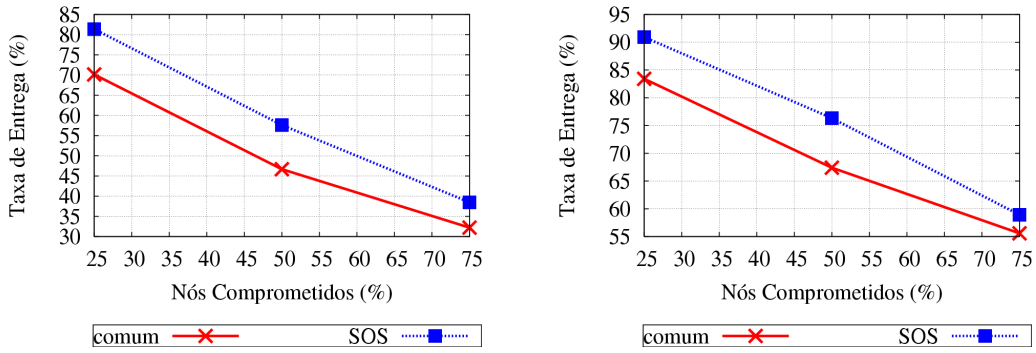
A Figura 2 apresenta a taxa de entrega de mensagens prioritárias em função do percentual delas nas redes comum e SOS. O ganho (diferença entre as taxas de entrega) do SOS, para 25%, 50% e 75% de mensagens prioritárias, foi, respectivamente, de 11,2%, 11,3% e 10,9%, para o ataque de buraco negro (Figura 2(a)), e de 7,5%, 7,4% e 6,9%, para o ataque de repasse seletivo (Figura 2(b)). Repare que à medida que o percentual de mensagens prioritárias cresceu, o ganho tendeu a cair. Isso porque quanto maior o número de mensagens prioritárias, menor o percentual das não prioritárias. Um número menor de mensagens prioritárias, por sua vez, diminui a freqüência de monitoramento das rotas. Como uma das idéias centrais das ROs é justamente o monitoramento das rotas, isso influenciou negativamente o desempenho do SOS.

A Tabela 3 apresenta o *overhead* de energia do SOS em relação à rede comum para os diferentes percentuais de mensagens prioritárias. O *overhead* para os CHs ficou em entre 43% e 44% para o ataque de buraco negro e entre 50% e 55% para o de repasse seletivo. Entretanto, o fato do *overhead* dos membros comuns de grupo ser menor

mensagens prioritárias	buraco negro			repasse seletivo		
	CH	geral	por mensagem	CH	geral	por mensagem
25%	43%	24%	7%	50%	26%	16%
50%	44%	25%	7%	54%	27%	16%
75%	44%	25%	8%	55%	27%	17%

Tabela 3: Overhead de energia em função do percentual de mensagens prioritárias

contribuiu para a diminuição do *overhead* geral. Veja, para o ataque de buraco negro o *overhead* geral ficou entre 24% e 25% e para o de repasse seletivo entre 25% e 27%. Veja também que, não importa o ataque, esse *overhead* aumentou à medida que o percentual de mensagem prioritárias cresceu. Isso porque quanto maior o percentual dessas mensagens, maior o valor absoluto de mensagens que são beneficiadas com o SOS. Isso, por sua vez, aumenta o tráfego de mensagens e o gasto de energia da rede. Em relação ao *overhead* por mensagem, ele ficou mais baixo que os demais (entre 7% e 8% para o ataque de buraco negro e entre 16% e 17% para o de repasse seletivo), já que a taxa de entrega do SOS foi consideravelmente maior que a da rede comum (Figuras 2(a) e 2(b)).



(a) **buraco negro** (b) **repasse seletivo**

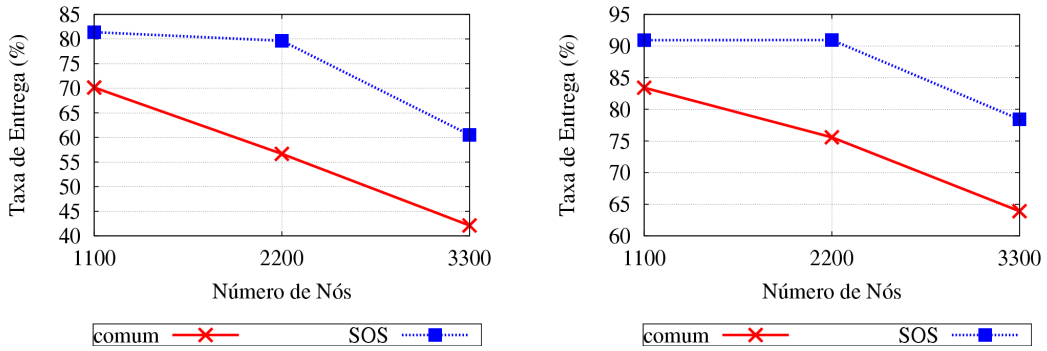
Figura 3: Taxa de entrega de mensagens prioritárias em função do percentual de nós comprometidos

A Figura 3 apresenta a taxa de entrega de mensagens prioritárias em função do percentual de nós comprometidos. Como já era esperado, as taxas caíram com o aumento dos nós comprometidos para ambas as redes e ambos os ataques. Os ganhos do SOS, contudo, apresentaram uma leve diferença para os dois ataques. No ataque de buraco negro (Figura 3(a)), por exemplo, o ganho se manteve em torno de 11% , para 25% e 50% dos comprometidos, e diminui até 6.2%, para 75% de nós comprometidos. Já no ataque de repasse seletivo (Figura 3(b)), o ganho iniciou em 7.5% (25% dos nós comprometidos), cresceu para 8.9% (50% dos nós comprometidos) e voltou a diminuir para 3.3% (75% dos nós comprometidos). Observe que para quaisquer dos ataques o ganho do SOS diminui com 75% da rede comprometida. Isso porque, neste cenário, são raras as rotas que não estão comprometidas e, conseqüentemente, é difícil de se achar rotas alternativas seguras.

A Tabela 4 apresenta o *overhead* de energia do SOS em função do percentual de nós comprometidos. Diferentemente do que ocorreu com o aumento de mensagens prioritárias, agora, o *overhead* diminui à medida que os nós comprometidos aumentou. Repare que para 50% de comprometimento da rede – ponto em que as diferenças na taxa de entrega entre o SOS e a rede comum foram mais discrepantes (Figura 3) –, o *overhead* por mensagem chegou a ser negativo para o ataque de buraco negro.

nós comprometidos	buraco negro			repasse seletivo		
	CH	geral	por mensagem	CH	geral	por mensagem
25%	43%	24%	7%	50%	26%	16%
50%	32%	22%	-1%	45%	24%	9%
75%	26%	21%	1%	31%	21%	15%

Tabela 4: Overhead de energia em função do percentual de nós comprometidos



(a) **buraco negro** (b) **repasse seletivo**

Figura 4: Taxa de entrega de mensagens prioritárias em função do tamanho da rede

A Figura 4 apresenta a taxa de entrega de mensagens prioritárias em função do tamanho da rede. As diferenças entre as taxas de entrega para redes de tamanho 1100, 2200, e 3300, foram, respectivamente, de 11,2%, 23% e 18,4%, para o ataque de buraco negro (Figura 4(a)), e de 7,5%, 15,4% e 14,5%, para o ataque de repasse seletivo (Figura 4(b)). Observe que taxa de entrega da rede comum reduziu à medida que a rede crescia. Isso porque aumentando o tamanho da rede, o tamanho da rota (número de *hops* médio para que uma mensagem alcance a ERB) também aumenta. E, por sua vez, a chance da rota conter um nó comprometido fica maior. O SOS, entretanto, conseguiu manter a mesma taxa de entrega da rede de 1100 nós na rede de 2200, aumentando seu ganho em relação à rede comum – 23% para o ataque de buraco negro e 15,4% para o de repasse seletivo.

Este aumento, aliado ao fato de que o *overhead* geral das redes variou pouco (Tabela 5), fez com que os valores de *overhead* por mensagem alcançassem os valores mais baixos de nossos resultados. São eles -13% para o ataque de buraco negro e 5% para o ataque de repasse seletivo.

número de nós	buraco negro			repasse seletivo		
	CH	geral	por mensagem	CH	geral	por mensagem
1100	43%	24%	7%	50%	26%	16%
2200	51%	27%	-10%	53%	28%	6%
3300	41%	25%	-13%	56%	29%	5%

Tabela 5: Overhead de energia em função do tamanho da rede

8. Conclusão

Neste trabalho apresentamos um mecanismo de roteamento seguro em RSSFs chamado SOS. O SOS é baseado em ROs e estabelece e monitora rotas alternativas a fim de des-

cobrir rotas mais seguras que a rota padrão. Essas rotas são utilizadas para o envio de mensagens que contêm informações sensíveis.

Os custos, benefícios e escalabilidade do SOS foram avaliados em diversos cenários. Para cada um, os ataques de buraco negro e repasse seletivo foram considerados. O SOS mostrou-se eficaz no aumento da taxa de entrega de mensagens prioritárias e eficiente em termos de consumo de energia – como descrito com números concretos na Seção 7.

Até onde sabemos, o SOS é o primeiro mecanismo de segurança para RSSFs baseado em ROs e diversos caminhos podem ser tomados a partir deste primeiro passo. Por exemplo, pode-se aferir qual o número ideal de rotas alternativas que devem ser atribuídas por nó *overlay*. Outro caminho é verificar a eficácia do trabalho em relação a outros tipos de ataques ou quando múltiplos ataques ocorrem simultaneamente.

Referências

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

Andersen, D., Balakrishnan, H., Kaashoek, F., and Morris, R. (2001). Resilient Overlay Networks. In *The 8th ACM Symposium on Operating Systems Principles (SOSP'01)*, pages 131–145, Banff, CA.

Bohge, M. and Trappe, W. (2003). An authentication framework for hierarchical ad hoc sensor networks. In *2003 ACM workshop on Wireless security*, pages 79–87.

Boukerche, A., Pazzi, R. W. N., and Araujo, R. B. (2004). A fast and reliable protocol for wireless sensor networks in critical conditions monitoring applications. In *the 7th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems (MSWiM '04)*, pages 157–164, New York, NY, USA. ACM Press.

Capkun, S. and Hubaux, J.-P. (2003). Biss: building secure routing out of an incomplete set of security associations. In *ACM workshop on Wireless security (WISE'03)*, pages 21–29. ACM Press.

de Oliveira, S., Wong, H. C., and Nogueira, J. M. S. (2005). Nekap: Estabelecimento de chaves resiliente a intrusos em rssf. In *23º Simposio Brasileiro de Redes de Computadores (SBRC'05)*.

Deng, J., Han, R., and Mishra, S. (2003). A performance evaluation of intrusion-tolerant routing in wireless sensor networks. In *IPSN*, volume 2634 of *Lecture Notes in Comp. Science*, pages 349–364. Springer.

Du, W., Deng, J., Han, Y. S., Varshney, P. K., Katz, J., and Khalili, A. (2005). A pairwise key pre-distribution scheme for wireless sensor networks. *ACM Transactions on Information and System Security*. Also appeared in 10th ACM CCS '03.

Eschenauer, L. and Gligor, V. D. (2002). A key management scheme for distributed sensor networks. In *9th ACM conference on Computer and communications security (CCS'03)*, pages 41–47. ACM Press.

Estrin, D., Govindan, R., Heidemann, J. S., and Kumar, S. (1999). Next century challenges: Scalable coordination in sensor networks. In *Mobile Computing and Networking*, pages 263–270, Seattle, WA.

Fall, K. and Varadhan, K. (2001). *Network Simulator Notes and Documentation*. The VINT Project.

Ferreira, A. C., Vilaça, M. A., Oliveira, L. B., Habib, E., Wong, H. C., and Loureiro, A. A. F. (2005). On the security of cluster-based communication protocols for wireless sensor networks. In *4th IEEE International Conference on Networking (ICN'05)*, volume 3420 of *Lecture Notes in Computer Science*, pages 449–458, Reunion Island.

Ganesan, D., Govindan, R., Shenker, S., and Estrin, D. (2001). Highly-resilient, energy-efficient multipath routing in wireless sensor networks. In *2nd ACM international symposium on Mobile ad hoc networking & computing (MobiHoc'01)*, pages 251–254, New York, NY, USA. ACM Press.

Gura, N., Patel, A., Wander, A., Eberle, H., and Shantz, S. C. (2004). Comparing elliptic curve cryptography and rsa on 8-bit cpus. In *CHES*, pages 119–132.

Heinzelman, W. R., Chandrakasan, A., and Balakrishnan, H. (2000). Energy-efficient communication protocol for wireless microsensor networks. In *IEEE Hawaii Int. Conf. on System Sciences*, pages 4–7.

Hill, J. L. and Culler, D. E. (2002). Mica: A wireless platform for deeply embedded networks. *IEEE Micro*, 22(6):12–24.

Intanagonwiwat, C., Govindan, R., and Estrin, D. (2000). Directed diffusion: a scalable and robust communication paradigm for sensor networks. In *Proceedings of the sixth annual international conference on Mobile computing and networking*, pages 56–67, Boston, MA USA.

Karlof, C. and Wagner, D. (2003). Secure routing in wireless sensor networks: Attacks and countermeasures. *Elsevier's AdHoc Networks Journal, Special Issue on Sensor Network Applications and Protocols*, 1(2–3):293–315.

Kong, J., Luo, H., Xu, K., Gu, D. L., Gerla, M., and Lu, S. (2002). Adaptive Security for Multi-layer Ad-hoc Networks. *Wireless Communications and Mobile Computing, Wiley Interscience Press*, 2(5):533–547. Special Issue.

Levis, P., Madden, S., Polastre, J., Szewczyk, R., Whitehouse, K., Woo, A., Gay, D., Hill, J., Welsh, M., Brewer, E., and Culler, D. (2004). TinyOS: An operating system for wireless sensor networks. In Weber, W., Rabaey, J., and Aarts, E., editors, *Ambient Intelligence*. Springer-Verlag, New York, NY.

Liu, D., Ning, P., and Li, R. (2005). Establishing pairwise keys in distributed sensor networks. *ACM Transactions on Information and System Security (TISSEC)*, 8(1):41–77. Also appeared in CCS '03.

Lou, W., Liu, W., and Fang, Y. (2004). Spread: Enhancing data confidentiality in mobile ad hoc networks. In *INFOCOM*.

Marti, S., Giuli, T. J., Lai, K., and Baker, M. (2000). Mitigating routing misbehavior in mobile ad hoc networks. In *Mobile Computing and Networking*, pages 255–265.

Melo, E. J. D. and Liu, M. (2002). The effect of organization on energy consumption in wireless sensor networks. In *IEEE Globecom 2002*.

Oliveira, L. B., Wong, H. C., Bern, M., Dahab, R., and Loureiro, A. A. F. (2006). SecLEACH – a random key distribution solution for securing clustered sensor networks. In *5th IEEE International Symposium on Network Computing and Applications (NCA'06)*, Cambridge,MA,USA. to appear.

Oliveira, L. B., Wong, H. C., Bern, M., Habib, E., Loureiro, A. A. F., and Dahab, R. (2005a). SecLEACH - uma solução segura de distribuição de chaves para redes de sensores sem fio hierárquicas. In *V Simpósio Brasileiro em Segurança da Informação e de Sistemas Computacionais (SBSeg'05)*, Brazil.

Oliveira, L. B., Wong, H. C., and Loureiro, A. A. F. (2005b). LHA-SP: Secure protocols for hierarchical wireless sensor networks. In *9th IFIP/IEEE International Symposium on Integrated Network Management (IM'05)*, pages 31–44, Nice, France.

Papadimitratos, P. and Haas, Z. J. (2002). Secure routing for mobile ad hoc networks. In *Communication Networks and Distributed Systems Modeling and Simulation Conference (CNDS' 02)*, pages 193–204.

Perrig, A., Szewczyk, R., Wen, V., Culler, D., and Tygar, J. D. (2002). SPINS: Security protocols for sensor networks. *Wireless Networks*, 8(5):521–534. Also appeared in MobiCom'01.

Shamir, A. (1979). How to share a secret. *Communications of the ACM*, 22(11):612–613.

Staddon, J., Balfanz, D., and Durfee, G. (2002). Efficient tracing of failed nodes in sensor networks. In *the 1st ACM international workshop on Wireless sensor networks and applications*, pages 122–130.

Wood, A. and Stankovic, J. (2002). Denial of service in sensor networks. *IEEE Computer*, 35(10):54–62.

Zhou, L. and Haas, Z. J. (1999). Securing ad hoc networks. *IEEE Network*, 13(6):24–30.

Zhu, S., Setia, S., and Jajodia, S. (2003). LEAP: efficient security mechanisms for large-scale distributed sensor networks. In *10th ACM conference on Computer and communication security*, pages 62–72.