Rotas Alternativas para Detecção e Aumento da Resiliência à Intrusão Distribuída em RSSF

Sérgio de Oliveira[1,2], Hao Chi Wong[1], José Marcos Nogueira[1], Wellington Paula[1]

[1]Instituto de Ciências Exatas – Universidade Federal de Minais Gerais (UFMG)
Av. Presidente Antônio Carlos, 6627 – Belo Horizonte – MG – Brasil

[2]Universidade Presidente Antônio Carlos (UNIPAC)
BR 482 Km 3 – Conselheiro Lafaiete – MG - Brasil

{sergiool, hcwong, jmarcos, wpassos}@dcc.ufmg.br

Resumo*: As redes de sensores sem fio estão sujeitas a diversos tipos de ataques, especialmente ataques do tipo negação de serviço (DoS). Este trabalho propõe uma estratégia de roteamento que considera rotas alternativas visando aumentar a resiliência a intrusão e permitir, de forma mais eficiente, a detecção de intrusos. O algoritmo de roteamento TinyOS beaconing foi modificado para que todo nó utilize dois caminhos para enviar suas informações para a estação base. Caso um intruso esteja presente em um desses caminhos, inutilizando-o, o caminho alternativo pode continuar funcionando, garantindo a entrega de parte das informações. O algoritmo modificado é apresentado, bem como uma avaliação do seu desempenho, em termos de energia, e eficácia. Os resultados foram verificados através de simulação. A detecção permite bons resultados mesmo com um alto número de intrusos.*

Abstract*: Wireless Sensor Networks are targets of several attack types, especially DoS attacks. This work proposes a routing strategy that considers alternative routes to increase intrude resilience and enable, in an efficient way, the intrusion detection. The routing algorithm TinyOS beaconing was changed to each node use two route to send its information to base station. In case an intruder is present in one of these paths, the other can continue working, guaranteeing the delivery of partial information. The changed algorithm is presented as well evaluation of its performance, in terms of power and efficiency. The results were verified through simulation. The detection allows good results even with a high number of intruders.*

1 Introdução

Redes de Sensores Sem Fio (RSSF) são redes *ad hoc* que permitem inúmeros tipos de aplicações, incluindo o monitoramento de regiões críticas ou remotas. São formadas por centenas ou milhares de nós miniaturizados, com baixo poder de processamento e curto alcance de rádio. Seu uso, de forma descartável, requer o uso de componentes de baixo custo e baixo consumo de energia para manter a longevidade da rede.

A simplicidade da estrutura da rede e a exigüidade de recursos exigem mecanismos de segurança simples e funcionais. Algumas características da rede como,

ambiente hostil, baixo poder de processamento dos nós e comunicação sem fio, tornam essas redes muito mais vulneráveis a intrusão. Diversos tipos de ataques podem ser usados nesse ambiente.

Os ataques de negação de serviço no roteamento estão entre os mais destrutivos [Karlof e Wagner, 2003][Wood e Stankovic, 2002]. Alguns poucos nós intrusos podem deixar boa parte da rede fora de serviço. Nesse tipo de ataque, parte da rede é mantida em silêncio ou apenas parte das mensagens é entregue. Essa parte da rede depende normalmente de um nó para o roteamento de suas mensagens para a estação base. Esse nó pode estar com falhas ou ser um nó intruso. Pode ser difícil identificar se um nó está silencioso por causa de uma falha ou uma intrusão. O intruso pode ainda simular uma falha intermitente, como em um ataque de reenvio seletivo.

Este trabalho propõe o uso de rotas múltiplas, com alternância no envio de informações, para aumentar a resiliência da rede e facilitar a detecção de nós intrusos. Rotas múltiplas são caminhos redundantes no roteamento. São usadas neste trabalho de forma alternada, sem replicação de informações. A alternância de rotas aumenta a resiliência da rede a intrusos, visto que oferece uma opção a mais para o nó. Caso exista um intruso em uma das rotas, uma rota alternativa possibilita o encaminhamento dos pacotes pela outra. Além disso, através da análise dos pacotes recebidos, é possível descobrir rotas que não entregam os pacotes corretamente. Uma análise dos pacotes entregues por todos os nós permite a identificação de nós que estejam gerando problemas no roteamento.

A resposta da rede deve ser diferente quando ocorrer uma falha ou uma invasão. No caso de uma falha, a rede deve tentar contornar o problema identificando outro nó que possa assumir a função do nó com falha. Uma nova execução do algoritmo de estabelecimento de rotas pode resolver o problema ocasionado por uma falha. No caso de uma intrusão, a rede deve, antes, isolar o nó intruso. Somente a execução do algoritmo de estabelecimento de rotas não elimina o intruso, pois ele vai participar desse processo e se inserir novamente na árvore de roteamento.

Vários trabalhos abordam o uso de rotas múltiplas em RSSF [Deng et al, 2003] [Karlof et al, 2002] [Ganesan et al, 2001]. As rotas múltiplas podem ser usadas de forma redundante ou de forma alternada. Rotas usadas de forma redundante aumentam a tolerância à falhas, porém aumentam o consumo de energia, pois replicam as informações pela rede em diversos caminhos. O uso de rotas de forma alternada, neste trabalho, foi escolhido para manter o consumo de energia bem próximo daquele verificado sem os mecanismos propostos.

O uso de rotas alternativas contribui para o aumento da resiliência e ainda permite a realização de detecção nós intrusos de forma eficiente. A proposta deste trabalho, publicada como um resumo estendido [Paula et al, 2005], é mostrada aqui em detalhes e com resultados. São mostrados resultados de simulações, indicando que o aumento da resiliência e ainda que o algoritmo de detecção de intrusos tem grande eficiência na presença de poucos intrusos.

2 Preliminares

2.1 Modelo

Este trabalho foi desenvolvido para RSSF planas, ou seja, todos os nós desempenham funções semelhantes. A rede não utiliza qualquer tipo de fusão de dados. O nó usado como referência é o nó Berkeley Mica2 Motes, desenvolvido na Universidade de Berkeley [Crossbow, 2004]. Esse tipo de nó sensor foi escolhido por estar disponível comercialmente, o que viabiliza a validação pela implementação e testes em ambientes reais, e ser bastante utilizando em experimentos na literatura.

As redes consideradas neste trabalho podem ter tamanhos diversos, desde algumas dezenas de nós até milhares de nós. A distribuição desses nós pode ser feita de forma aleatória, uniforme, ou ainda em outros formas presentes na literatura [Oliveira et al, 2004].

A energia utilizada para transmissão (27 mA a 38,4 Kbaud) é maior que a energia gasta no processamento (8 mA a 4 MHz) [Crossbow, 2004]. Dessa forma, as operações de rede são mais caras e demoradas que as operações de processamento.

A comunicação sem fio não é segura e é sujeita a escuta, inserção de pacotes e replicação de mensagens. Os nós são sujeitos a adulteração física (*tampering*). Se um nó é comprometido, todas as informações que ele manipula podem ser conhecidas pelo atacante. Um pressuposto válido é que a estação base, única, localizada no centro da rede e com recursos ilimitados, não pode ser comprometida.

A estação base conhece o número de mensagens enviadas pelos nós, que representam os dados enviados periodicamente e as respostas de consultas realizadas pela estação base.

2.2 Negação de Serviço

Ataques do tipo Negação de Serviço (*DoS - Denial of Service*) são ataques contra a disponibilidade da rede. Seu objetivo é tornar a rede indisponível indefinidamente ou durante um determinado período de tempo. Esses ataques bloqueiam o serviço da rede ou de parte dela, impedindo sua operação normal.

Wood e Stankovic classificaram os ataques de negação de serviço em RSSF de acordo com os componentes de rede em que estão focados [Wood e Stankovic, 2002]. Seguindo essa classificação, os ataques de negação de serviço no roteamento são o objetivo desse trabalho:

- Buraco Negro (*Black Hole*): supressão total do serviço de roteamento no nó inimigo inserido em uma rota;

- Reenvio Seletivo (*Selective Forward*): supressão parcial do serviço de roteamento, simulando falhas intermitentes e dificultando a identificação do ataque;

- Direção Falsa (*Misdirection*): envio de pacotes por nós invasores por caminhos errados, podendo exaurir a energia da rede e impedir o encaminhamento das informações para a estação base;

- Buraco de Verme (*Worm Hole*): tunelamento de mensagens recebidas em parte da rede para outra parte, gerando problemas de roteamento;

- Inundação de Atualização de Rotas (*Hello Flood*): amplificação de sinal de um nó intruso durante o processo de estabelecimento de rotas. Todos os nós que escutam esse sinal passam a identificar o nó cujo sinal foi amplificado como vizinho responsável pelo roteamento.

O ataque conhecido como Buraco Negro é o alvo de estudo desse trabalho, mas ele poderia ser estendido para qualquer um dos outros ataques, pois todos eles interferem no roteamento para reduzir a produção da rede.

Em uma rede com centenas ou milhares de nós, a presença de apenas um intruso pode não ser significativa, a não ser que ele tenha uma função importante na árvore de roteamento. O uso de um número maior de intrusos, em um ataque de negação de serviço distribuído, teria uma ação muito mais efetiva nesse ambiente. Por esse motivo, foi considerada a possibilidade de haver um número maior de intrusos, representando 10 ou 30% da rede.

2.3 Detecção de Intrusos

A detecção de intrusos em RSSF deve ter uma abordagem muito diferente daquela apresentada para as redes convencionais, devido às diferenças de modelo, ataques e recursos. Dois tipos de abordagens podem ser usados para a detecção de intrusos em RSSF. Na abordagem centralizada, a estação base é responsável por detectar os intrusos, a partir de informações extraídas da rede, especialmente a produção dos nós [Teixeira et al, 2005]. Na abordagem descentralizada, alguns ou todos os nós da rede executam operações simples para detecção de intrusos [Silva et al, 2005]. Na abordagem centralizada, a estação base tem um grande conjunto de informações à sua disposição, que podem facilitar o processo. Na abordagem descentralizada, a maior vantagem é a disponibilidade instantânea de informações, visto que os nós podem perceber os ataques no momento que acontecem.

A detecção de intrusos é normalmente acompanhada da revogação do nó intruso. A revogação representa a exclusão do nó da rede, eliminando as possibilidades de comunicação desse nó com seus vizinhos. Esse processo deve ser autenticado, para evitar que nós intrusos promovam a revogação dos nós autênticos. Como os nós não são protegidos contra *tampering* (violação física) no modelo utilizado, é bem mais seguro permitir que apenas a estação base promova a revogação de nós. Do contrário, um nó intruso que possa ser autenticado pela rede, provavelmente originado numa ação de *tampering*, poderia agir revogando nós autênticos, promovendo assim outro ataque do tipo negação de serviço.

3 Rotas Alternativas

3.1 Estabelecimento

Rotas múltiplas podem ser disjuntas, com todos os nós distintos entre as rotas; ou entrelaçadas, quando contém nós em comum. Ganesan et al apresentam uma comparação entre rotas múltiplas disjuntas e entrelaçadas. Rotas disjuntas são mais resilientes a falhas e intrusão [Ganesan et al, 2001]. Rotas entrelaçadas são mais baratas

em termos de energia para criação e manutenção. Um ponto de falha em um nó comum, porém, pode inutilizar todas as rotas existentes.

Vários algoritmos de roteamento foram propostos para RSSF. Os mecanismos utilizados para a criação e manutenção de rotas em cada um destes algoritmos são justificados em função do tipo de rede e do tipo de aplicação. É possível encontrar formas de criar rotas múltiplas para cada protocolo. Este trabalho, porém, restringe-se ao protocolo conhecido como PI, ou Propagação da Informação [Barbosa, 1996]. Esse algoritmo é usado no Sistema Operacional *Tiny OS*, onde é conhecido como *Tiny OS beaconing*.

Tiny OS beaconing é um algoritmo de roteamento baseado no uso de um *beacon*, enviado em modo de difusão pela estação base. Cada nó determina qual dos seus nós vizinhos será o próximo salto em direção à estação base a partir da recepção do *beacon*. A ordem de recepção do *beacon* pelos nós é que determina a criação de rotas. Para a criação de rotas múltiplas, o nó deve estabelecer mais rotas além daquela originada no nó que enviou o *beacon* primeiro. Assim, a segunda rota é estabelecida a partir do segundo nó vizinho que repasse o *beacon*. Esse processo não garante a criação de rotas disjuntas. Porém, garante que cada nó terá saídas alternativas no primeiro salto, através de dois nós de saída. A rota criada após o recebimento do *beacon* pela primeira vez será tratada como rota padrão e a rota criada após o recebimento do *beacon* pela segunda vez será tratado como rota alternativa.

Neste trabalho, o algoritmo de roteamento de cada nó terá um tratamento diferenciado para as mensagens originadas no nó e aquelas originadas em outros nós que devem ser repassadas. Para as mensagens originadas no nó existirão duas rotas usadas de forma alternada: a rota padrão e a rota alternativa. Para as mensagens originadas em outros nós será usada apenas a rota padrão. Três motivos contribuem para essa decisão. O primeiro é o fato de ser caro identificar em cada pacote todo o caminho percorrido. A identificação de apenas um salto alternativo é simples, ocupando apenas um bit. Ao contrário, para registrar todo o caminho seria necessário usar um bit para cada salto. O segundo motivo é o fato de que a rota padrão tem custo menor ou igual à rota alternativa, pois a rota padrão foi criada a partir do nó que repassou o *beacon* primeiro. O terceiro motivo diz respeito à criação de laços no algoritmo de roteamento. Caso todos os pacotes pudessem seguir por ambos caminhos, a existência de laços no roteamento poderia fazer um pacote passar várias vezes por um mesmo nó. Porém, se após o primeiro salto os pacotes seguirem apenas as rotas padrão, os pacotes não percorrerão laços.

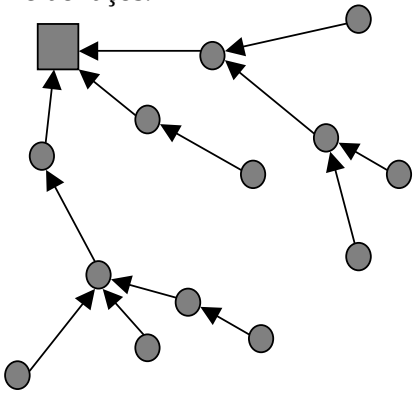

Figura 1 – Árvore gerada pelo Tiny OS

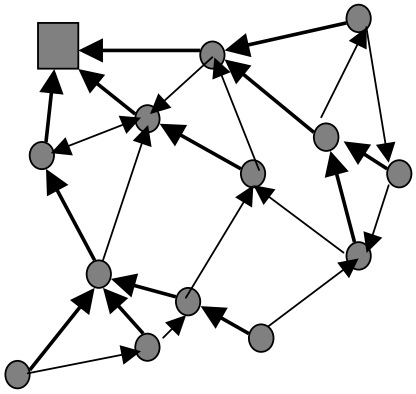

Figura 2 – Grafo resultante com rotas múltiplas

As figuras 1 e 2 mostram o resultado da criação das rotas alternativas sobre uma pequena rede. As setas com linha mais forte na figura 2 indicam as rotas padrão, já existentes na figura 1. As demais setas indicam as rotas alternativas.

Depois do estabelecimento das rotas, é necessário um mecanismo em cada nó para decidir qual rota será usada para cada pacote enviado. Cada nó tem opções de destinos para o repasse das informações até a estação base. A escolha do caminho a ser seguido rumo a estação base deve levar em consideração alguns requisitos, levando em conta a detecção e o isolamento dos intrusos: o volume de dados deve ser equilibrado entre as duas rotas; e a estação base e o nó devem conhecer *a priori* qual rota usada para cada informação a ser enviada. Assim, na ausência de um dado, a estação base sabe de onde deveria ter sido originada.

3.2 Conhecimento da Topologia

Para identificar intrusos, a estação base deve conhecer a topologia da rede. Mecanismos de conhecimento da topologia já estão disponíveis na literatura. Staddon et al propõem uma forma de conhecimento da topologia pelo envio do identificador de um nó vizinho em cada medida efetuada pelo nó sensor [Staddon et al, 2002]. Com isso o custo de energia é menor, pois não existem mensagens extras. Porém, o tempo necessário para o conhecimento da topologia é maior. A opção por um mecanismo capaz de descobrir a topologia em um tempo menor permite a detecção de intrusos desde os momentos iniciais de funcionamento da rede.

Neste trabalho, para a estação base conhecer a topologia da rede, cada nó envia uma mensagem indicando quais são os vizinhos responsáveis pelas rotas até a estação base. Após o recebimento das informações dos nós, a estação base é capaz de criar um grafo de conectividade da rede, que será usado no algoritmo de detecção de intrusos. Nós e rotas em silêncio, ou gerando um número baixo de informações, serão marcadas no grafo para possibilitar a localização de intrusos.

4 Detecção de Intrusos

Para a estação base descobrir a existência de um nó sensor intruso, a diferença entre o número de mensagens esperadas pela estação base, provenientes de cada nó, e o número de mensagens efetivamente recebidas é calculada. Esse dado indica a produção do nó pela rota.

É possível identificar perdas nas rotas. Para tanto, toda mensagem enviada deve conter a informação de qual rota foi utilizada. Assim, com base na função de distribuição das mensagens pelas rotas, é possível comparar o número de mensagens recebidas com o número de mensagens esperadas em cada rota.

A partir dessas informações é possível propor um algoritmo para identificar intrusos. O algoritmo é recursivo, partindo da estação base em direção aos nós folha no grafo de conectividade. Utiliza o princípio de divisão e conquista, onde, a cada nó existe a possibilidade de localização de uma intrusão. Caso o efeito do intruso seja verificado na produção de outros nós, a possibilidade de sua presença será reforçada.

O resultado da execução do algoritmo é a marcação de nós em silêncio e de possíveis nós intrusos. Um nó em silêncio não envia qualquer tipo de informação. Um

nó pode estar em silêncio por uma falha ou pela presença de intrusos. Caso já tenha sido identificado um intruso no caminho de roteamento de um nó em silêncio, os pontos do intruso aumentarão devido a esse nó. Do contrário, o nó é marcado como falha.

Nós com falhas intermitentes ou canais com muitos erros podem ser confundidos com a presença de intrusos realizado ataques do tipo reenvio seletivo. As falhas intermitentes, porém, tem padrão aleatório. Já esses ataques tendem a seguir uma lógica, restringindo os pacotes com base em alguma informação disponível, como a origem. Mesmos os ataques de reenvio seletivo, onde os pacotes são repassados aleatoriamente, podem ser distinguidos das falhas intermitentes pelo percentual de perda. Este trabalho não está interessado em ataques que tenham baixo percentual de perda.

O algoritmo 1 apresenta apenas as inicializações e a chamada à função Detecta Intruso, mostrada no algoritmo 2. Na inicialização do algoritmo são calculadas as perdas em cada uma das rotas, além da inicialização das variáveis e da chamada da função de detecção de intrusos a partir da estação base.

Pontos do Intruso ← 0;

Nó Intruso ← vazio;

Para todo Nó I pertencente à rede

1. Perda padrão (I) ← Número de mensagens enviadas na rota padrão – número de mensagens recebidas na rota padrão / Número de mensagens esperadas na rota padrão
2. Perda alternativa (I) ← Número de mensagens enviadas na rota alternativa – número de mensagens esperadas na rota alternativa / Número de mensagens enviadas na rota alternativa

Detecta intruso (Estação Base, Intruso, Pontos do Intruso)

Algoritmo 1 - Inicializações e chamada da detecção de intrusos

O algoritmo 2 apresenta a detecção de intrusos, que é realizada de forma recursiva a partir da estação base. Cada execução do algoritmo tem um nó em foco, na variável X. Esse algoritmo pode ser dividido em 3 partes: A verificação das perdas nos nós vizinhos do nó X que o utilizam como rota padrão, a verificação das perdas nos nós vizinhos do nó X que o utilizam como rota alternativa e a chamada da função, de forma recursiva, para os nós vizinhos que usam X como rota padrão.

Caso o nó X seja um nó intruso, os nós vizinhos de X que dependem desse nó devem apresentar taxas altas de perdas na rota que passa por X. O mesmo não deve acontecer na outra rota. A primeira e a segunda parte do algoritmo 2 verificam a diferença entre as perdas das duas rotas. Caso exista uma diferença grande entre essas perdas, o nó é um provável intruso. Para possibilitar uma maior acurácia na identificação dos intrusos, os prováveis nós intrusos recebem uma pontuação que é incrementada sempre que houver a suspeita que esse nó é um intruso.

A pontuação do intruso representa a extensão de seu ataque. Quanto mais abrangente o ataque maior será a pontuação. Caso a rota a partir do intruso identificado tenha poucos elementos, a pontuação do intruso também será baixa. O significado dessa baixa pontuação é um ataque que afeta poucos nós. Os nós que não tem nenhum filho na árvore de roteamento, conseqüentemente, não são detectados. Isso, porém, não representa problema, uma vez que seu ataque não é efetivo.

Detecta intruso (Nó X, Nó Intruso, Pontos do Intruso)

1. Para cada nó I, vizinho de X que utiliza esse nó como rota padrão:
   a. Se Perda padrão (I) >> Perda alternativa (I) então
      i. Se Pontos do Intruso = 0 então
         Intruso ← X;
      ii. Incrementa Pontos do Intruso;
   b. Se Perda padrão (I) = Perda alternativa (I) = 100 % então
      i. Se Pontos do Intruso ≠ 0
         Incrementa Pontos do Intruso;
      ii. Senão
         Marcar I como Falha
2. Para cada nó I, vizinho de X que utiliza esse nó como rota alternativa
   a. Se Perda alternativa (I) >> Perda padrão (I) então
      i. Se Pontos do Intruso = 0 então
         Intruso ← X;
      ii. Incrementa Pontos do Intruso;
   b. Se Perda padrão (I) = Perda alternativa (I) = 100 % então
      i. Se Pontos do Intruso ≠ 0
         Incrementa Pontos do Intruso;
      ii. Senão
         Marcar I como Falha
3. Para cada nó I, não marcado como falha, vizinho de X que utiliza esse nó como rota padrão:
   a. Se Pontos do Intruso = 0 então
      i. Detecta intruso(I, Intruso, Pontos do Intruso)
      ii. Se Pontos do Intruso ≠ 0 então
         Marcar Intruso e Pontos do Intruso
         Pontos do Intruso ← 0
         Intruso ← vazio;
   b. Senão
      i. Detecta intruso(I, Intruso, Pontos do Intruso)

Fim Detecta Intruso

Algoritmo 2 – Algoritmo recursivo para detecção de intrusos

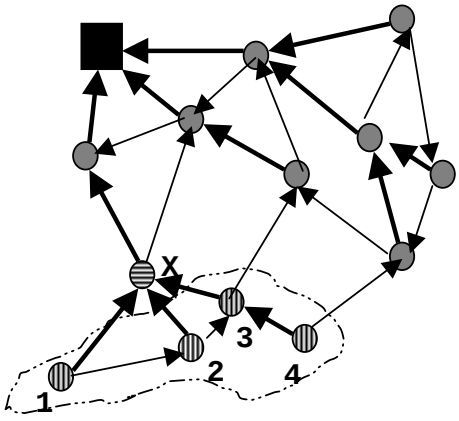

Figura 3 - Exemplo de detecção de intrusos

A partir do momento que um nó intruso é identificado, todos os indícios de intrusão percebidos nas rotas que dependem desse nó são associados a ele. Essa associação é realizada na terceira parte do algoritmo. Se um nó intruso já foi identificado, o segundo parâmetro da função "Detecta intruso" recebe o nó identificado como intruso. Se esse parâmetro estiver assinalado, então os indícios de intrusos são associados a esse nó.

Um exemplo de detecção de intrusos pode ser vista a partir da figura 3. Nesse exemplo, o nó marcado com X realiza um ataque, não repassando as mensagens enviadas pelos nós

marcados por 1, 2, 3 e 4. Os nós que dependem do nó marcado com X para seu roteamento estão circulados. O algoritmo de detecção do intruso iniciará a partir da estação base. Quando o nó marcado com X for analisado pelo algoritmo de detecção, será verificado as perdas por ambas as rotas dos nós marcados de 1 a 4. As perdas dos nós 1 e 2 ocorrerão na mesma proporção pelas duas rotas, pois ambas dependem do nó X. Não poderá ser constatada a presença do intruso pela análise desses nós. Os nós 3 e 4, porém, terão perdas muito maiores na rota padrão, que depende do nó X, que na rota alternativa, que não depende do nó X. Na primeira parte do algoritmo 2, na análise das perdas dos vizinhos que usam o nó X como rota padrão, serão identificadas as perdas excessivas dos nós 3 e 4 e o nó X será identificado como possível intruso, tendo sua pontuação de intruso registrada em duas unidades.

5 Avaliação

Três aspectos devem ser analisados para a avaliação da solução aqui apresentada: desempenho, funcionalidade e escalabilidade. O desempenho é verificado pelo consumo de energia, métrica de desempenho mais importante nas RSSF. A funcionalidade é verificada pela redução do número de nós silenciados por ataques de negação de serviço, bem como pelo resultado do algoritmo de detecção de intrusos. E a escalabilidade é verificada através de avaliações em diferentes massas de dados, variando de algumas dezenas de nós a milhares de nós.

Para avaliar esses aspectos, foram realizados três conjuntos de simulações. Embora alguns simuladores permitam simular as características de RSSF, como o SensorSim [Park et al, 2000], TOSSIM [Levis et al, 2003] e o PowerTOSSIM [Shnayder et al, 2004], esses simuladores não têm recursos para simular ataques. Dessa forma, o simulador apresentado em [Martins et al, 2005] inclui implementação de diversos tipos de ataque e é mais adequado a esse trabalho. O objetivo específico de cada conjunto de simulações pode ser assim definido:

- Consumo de Energia: Simulação de rede com protocolo de roteamento com rotas alternativas com o objetivo de verificar o aumento do consumo de energia causado pelo uso de rotas alternativas;

- Eficácia das Rotas Alternativas: Simulação de rede com protocolo de roteamento com rotas alternativas e com a presença de intrusos. O objetivo é verificar a eficácia do uso de rotas alternativas para reduzir o número de nós silenciados por certos ataques de negação de serviço;

- Eficácia da Detecção de Intrusos: Simulação de rede com protocolo de roteamento com rotas alternativas, com a presença de intrusos e execução do algoritmo de detecção de intrusos. O objetivo é verificar a eficácia do algoritmo de detecção de intrusos;

As redes simuladas utilizam distribuições diversas com quantidades de nós variando entre 40 e 1025, distribuídas de diversas formas, desde a distribuição aleatória até a distribuição uniforme, conforme apresentado em [Oliveira et al, 2004]. O uso de cenários diversos possibilita verificar a escalabilidade da solução aqui apresentada.

5.1 Consumo de Energia

A medida de desempenho mais importante em RSSF é o consumo de energia, pelo fato das fontes serem usadas de forma descartável. Qualquer mudança deve ser avaliada, então, em relação ao seu consumo de energia. Esta seção apresenta algumas simulações de RSSF típicas sem a presença de rotas alternativas e também com a presença de rotas alternativas. O objetivo é verificar qual o aumento no consumo de energia causado pelo uso dessa abordagem.

As condições de simulação incluem a geração de dados pelos nós em intervalos fixos com seu envio à estação base. A energia gasta com o processamento é uma função das tarefas realizadas pelo nó e pelo tempo de simulação. A energia gasta com transmissão é uma função do número de pacotes transmitidos. O algoritmo de estabelecimento de rotas só é executado uma única vez. Dessa forma, alguns poucos nós, responsáveis por repassar pacotes de muitos outros, têm consumo muito maior que os demais.

O resultado obtido, nesse caso, é a energia média gasta pelos nós. A tabela 1 mostra o consumo médio para o uso de rotas simples e o consumo médio com o uso de rotas alternativas.

Tabela 1 - Aumento do consumo pelo uso de Rotas Alternativas

Teste	Número de Nós	Consumo Médio sem alternância	Consumo Médio com alternância	Percentual de aumento
Aleatório	1025	501,54	519,45	3,57%
Aleatório Curto	399	346,17	360,68	4,19%
Aleatório Mini	40	145,03	168,70	16,32%
Faixas	1024	486,28	504,78	3,80%
Faixas Curtos	399	330,96	346,46	4,68%
Hexagonal	1020	517,98	532,22	2,75%
Hexagonal Curto	399	336,48	347,77	3,35%
Pertuba 5	1024	519,89	534,05	2,72%
Pertuba 5 Curto	399	338,90	350,29	3,36%

O uso de rotas alternativas representou um aumento no consumo da rede entre 2 e 16 %, sendo que esse último só foi registrado para uma vez, para uma rede muito pequena, com apenas 40 nós. Observa-se que o aumento do consumo pelo uso de rotas alternativas é maior para um número menor de nodos. Em redes com um maior número de nós, a energia adicional gasta com a rota alternativa é menos significativa, em função da maior energia gasta para roteamento do dado até a estação base.

A distribuição da energia gasta pelos nós da rede também é interessante para o nosso trabalho. Como o simulador não implementa nenhum esquema de renovação das rotas, alguns nós são sobrecarregados, resultando em consumo excessivo de energia. Essa situação pode ser representada através dos gráficos das figuras 4 e 5, que apresentam posição *versus* energia para os nós na simulação da rede na distribuição aleatória com 1025 nós.

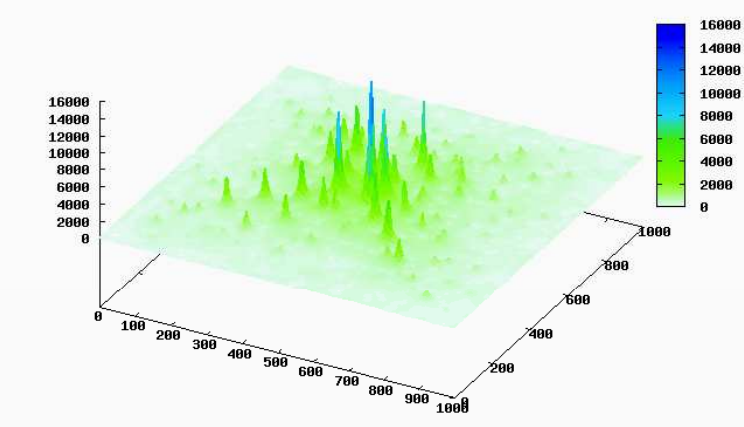

Figura 4 - Gráfico de Energia Gasta para Rede com Rotas Simples

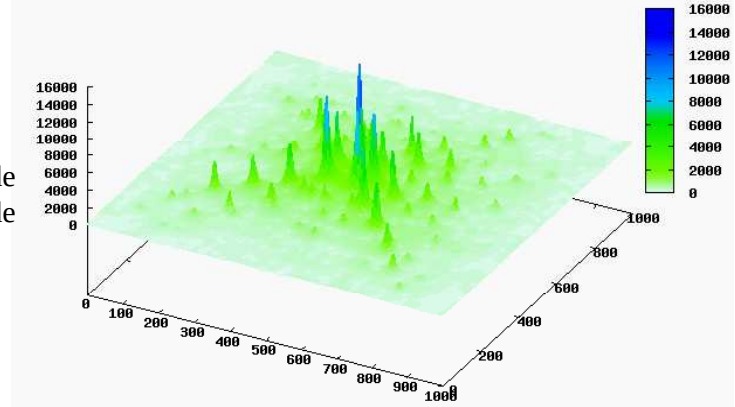

Figura 5 - Gráfico de Energia Gasta para Rede com Rotas Alternativas

É possível observar, visualmente, que não existem muitas diferenças em relação à distribuição de energia com o uso de rotas alternativas. Alguns poucos nós representam consumo excessivo em ambos os gráficos, especialmente aqueles próximos à estação base que desempenham funções importantes no roteamento. Esse resultado era esperado, uma vez que as diferenças no roteamento dizem respeito apenas ao primeiro salto. E o consumo excessivo de energia de alguns nós se devem ao número muito grande de rotas que passam por esse nó, na maioria, proveniente de nós distantes.

5.2 Eficácia das Rotas Alternativas

O segundo grupo de simulações tem objetivo de verificar a eficácia da alternância de rotas para o aumento da resiliência à intrusão. Isso se deve ao fato que a existência de um intruso em uma das rotas de um nó pode não ser suficiente para silencia-lo, uma vez que seus dados podem ser entregues pela outra rota, mesmo que em menor freqüência.

Esta seção apresenta algumas simulações de RSSF com rotas alternativas e a presença de intrusos. Para garantir a escalabilidade da solução aqui apresentada, várias simulações foram realizadas com diferentes quantidades de nós sensores, bem como nós intrusos. As distribuições de nós são subconjunto das apresentadas na seção anterior.

O ataque escolhido para essa análise foi o ataque de black hole, devido à abrangência desse ataque, capaz de silenciar todos os nós que dependem do nó intruso para o roteamento dos pacotes até a estação base. Os nós intrusos foram sorteados aleatoriamente, representando 10 ou 30% do total de nós. Ataques abaixo de 10% dos

nós da rede podem ter um impacto muito pequeno, e, acima de 30 %, um impacto muito grande, silenciando a maioria dos nós. O número de pacotes entregue, por cada nó, à estação base, foi registrado. Ao final, foi verificado o número de nós silenciados para as simulações das RSSF com rotas alternativas e sem o uso desse mecanismo. Os resultados podem ser vistos na tabela abaixo:

Tabela 2 - Aumento da Resiliência pelas Rotas Alternativas

			Sem Alternância	Com Alternância	
Teste	Número de Nós	Percentual de Nós Intrusos	Nós silenciados	Nós silenciados	Nós com resposta reduzida
Aleatório10	1024	10 %	378	274	221
Aleatório30	1024	30 %	739	649	269
Aleatório Curto10	399	10 %	71	49	81
AleatórioCurto 30	399	30 %	192	155	134
Aleatório Mini10	40	10 %	0	0	0
Aleatório Mini30	40	30 %	14	7	19
Faixas 10	1024	10 %	387	144	204
Faixas 30	1024	30 %	827	635	231
Faixas Curto10	399	10 %	105	78	117
Faixas Curto30	399	30 %	213	154	141

Os resultados aqui apresentados mostram um número menor de nós silenciados pelo uso de rotas alternativas. Isso acontece porque a presença de nós intrusos não silencia vários nós que apresentam alternativas para o caminho até a estação base. As informações desses nós são importantes para a estação base, pois aumentam a cobertura da rede. Na maioria das aplicações a cobertura da rede é mais importante que a freqüência de obtenção de dados.

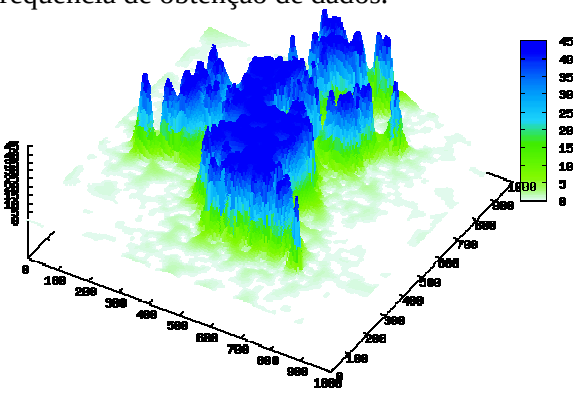

Figura 6 - Gráfico de produção para rede com 30% de intrusos com Roteamento Normal

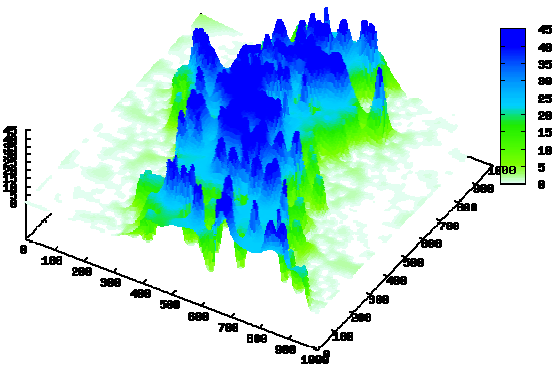

Figura 7 - Gráfico de Produção para rede com 30 % de intrusos com Roteamento Alternativo

Esse resultado pode ser visto também na forma de um gráfico de posição *versus* produção, apresentados nas figuras 6 e 7 para a rede com distribuição aleatória com 1024 nós e 10% de intrusos. Nesses gráficos é possível verificar também os nós que tiveram a produção reduzida pela presença de um intruso. Graficamente, podemos verificar que o uso de rotas alternativas permitiu uma cobertura maior à rede para um mesmo tipo de ataque. Além disso, um número menor de nós foi silenciado, permitindo o acesso a informações de uma maior região da rede.

5.3 Eficácia da Detecção de Intrusos

O último grupo de simulações a ser apresentado neste trabalho tem o objetivo de validar o algoritmo de detecção de intrusos apresentado e mostrar sua eficácia. Para tanto, esse algoritmo foi implementado no simulador utilizado [Martins et al, 2005].

De acordo com o algoritmo apresentado, a detecção de intrusos é eficiente para os intrusos que apresentam uma hierarquia mais alta na árvore de roteamento. Os intrusos que não tem função no algoritmo de roteamento não são detectados, pois sua presença não tem efeito sobre a rede.

Tabela 3 - Eficácia da detecção de intrusos para um intruso

			Ação dos intrusos		Resultados da detecção	
Teste	Número de Nós	Número de Simulações	Simulações com intruso efetivo (participa no roteamento)	Simulações com intruso inócuo (não participa do roteamento)	Simulações com Intruso detectado	Simulações com intruso não detectado
Aleatório	1024	40	16	24	16	24
Aleatório Curto	399	40	19	21	19	21
Aleatório Mini	40	40	20	20	20	20
Faixas	1024	40	15	25	15	25
Faixas Curto	399	40	19	21	19	21

A detecção de intrusos tem eficácia total para detectar um pequeno número de intrusos, desde que eles não estejam presentes em rotas comuns, ou seja, interferindo na produção de nós em comum. A tabela 3 apresenta os resultados para o caso de apenas um intruso presente no roteamento.

A detecção de intrusos tem sua eficácia reduzida para um número grande de intrusos. Mas ainda sim, nesse caso, o algoritmo de detecção consegue localizar parte dos intrusos. A revogação desses intrusos, com o restabelecimento de rotas e nova execução do algoritmo de detecção pode permitir a descoberta de outros intrusos, de forma que a solução converge para a detecção total do número de intrusos. A tabela 4 apresenta os resultados para as mesmas distribuições de nós usadas nas outras simulações. As quantidades de intrusos simulados foram de 10 e 30% do total de nós. Esses números foram escolhidos porque, a partir de 10% de intrusos, a rede tem sua produção reduzida de forma significativa. Acima de 30% de intrusos, a rede apresenta uma perda muita alta da produção, inviabilizando sua recuperação.

Tabela 4 - Detecção de Intrusos com grande número de intrusos

Teste	Número de Nós	Número de Intrusos	Número de Intrusos presentes no roteamento	Número de Intrusos detectados	Intrusos não detectados
Aleatório10	1024	121	49	30	19
Aleatório30	1024	355	129	29	100
Aleatório Curto10	399	49	23	16	7
AleatórioCurto 30	399	137	63	24	39
Aleatório Mini10	40	5	1	1	0
Aleatório Mini30	40	17	7	4	3
Faixas 10	1024	99	34	25	9
Faixas 30	1024	302	114	28	86
Faixas Curto10	399	49	19	11	8
Faixas Curto30	399	125	60	23	37

Não foi verificado nenhum caso de falso positivo. A presença de falhas intermitentes poderia levar a existência de falsos positivos, mas somente em situações muito específicas que não puderam ser obtidas nas simulações.

6 Trabalhos Relacionados

O trabalho proposto por Staddon et al identifica nós silenciosos que podem estar com falhas ou sem comunicação, devido a problema em outros nós [Staddon et al, 2002]. Caso estejam sem comunicação por causa de outros nós, a comunicação é restabelecida através do estabelecimento de outras rotas. Não são levadas em consideração, nesse trabalho, as características diversas dos ataques. Apenas são identificados os nós que não enviam informações à estação base. Ataques como reenvio seletivo não poderiam ser detectados nesse trabalho. O trabalho deste artigo é mais abrangente, pois possibilita que um nó sujeito a um ataque consiga enviar informações à estação base por uma rota alternativa, indicando que não existe falha nesse nó e sim a presença de um intruso que inibe o roteamento.

O trabalho conhecido por INSENS propõe o uso de rotas múltiplas usadas de forma redundante para aumentar a tolerância à intrusão [Deng et al, 2003]. Essa solução, porém, é limitada a um pequeno número de intrusos. Além disso, o uso de rotas de forma redundante aumenta o consumo de energia da rede pela retransmissão de informações semelhantes por várias rotas.

Silva et al propõem mecanismos de detecção de intrusos de forma descentralizada, onde alguns nós executam funções de monitores, detectando a presença de intrusos [Silva et al, 2005]. O problema dessa abordagem é que os nós monitores podem sofrer a ação de *tampering* de um inimigo, permitindo ao inimigo gerar falsos positivos para acarretar o isolamento de nós autênticos.

Teixeira et al propõem mecanismos de detecção de intrusos de forma centralizada, com base nas informações disponíveis na estação base, sem alteração na rede. Os dados são analisados em redes bayesianas. Os testes foram realizados com apenas um intruso

na rede. O trabalho aqui apresentado pode apresentar desempenho melhor, até mesmo com um número maior de intrusos.

A idéia do trabalho aqui apresentado foi proposta como resumo estendido em [Paula et al, 2005]. Esse artigo apresentou o princípio de rotas alternativas e o algoritmo de detecção de intrusos a ser utilizado. O presente trabalho apresenta os resultados das idéias apresentadas naquele artigo, que apresenta o mesmo grupo de autores.

7 Conclusões e Trabalhos Futuros

Este trabalho apresenta a proposta de uso de rotas alternativas para o roteamento em RSSF. O objetivo desse uso é aumentar a resiliência da rede à presença de intrusos, bem como permitir uma detecção eficiente de intrusos. Essa abordagem é necessária em redes onde a presença do inimigo pode significar a inserção de muitos nós intrusos, com o objetivo de eliminar as funcionalidades da rede.

As restrições existentes em RSSF, especialmente de consumo de energia foram consideradas no desenvolvimento desse trabalho. Resultados de simulações mostram que o aumento no consumo de energia é pequeno e pode ser assumido em redes onde a segurança é fator crítico.

A eficácia no aumento da resiliência também foi comprovada. Essa afirmação é possível considerando que é mais interessante para a rede aumentar a área com produção ativa do que a produção total. Os resultados das simulações mostram que o uso de rotas alternativas permite aumentar o número de nós produzindo, aumentando assim a área monitorada, embora a produção total da rede seja mantida sobre os mesmos parâmetros.

O próximo passo desse trabalho consiste em eliminar os intrusos detectados e executar novamente o algoritmo de detecção, visando a eliminação de outros intrusos que não foram detectados na primeira fase. Além disso, também serão testados outros ataques e outros algoritmos de roteamento, visando verificar a extensibilidade do mecanismo aqui proposto.

8 Referências

Barbosa, Valmir C., (1996) *An Introduction to Distributed Algorithms*, The MIT Press, Cambridge, Massachusetts, 1996.

Crossbow Technology Inc (2004) - *Mica 2 Wireless Measurement System* - disponível em http://www.xbow.com/Products/Product_pdf_files/Wireless_pdf/6020-0042-04_B_MICA2.pdf, acessado em 10 de março de 2004, San Jose, CA, USA, February 2004.

Deng, J.; Han, R.; Mishra, S. (2003) - *INSENS: Intrusion-Tolerant Routing in Wireless Sensor Networks* - Poster paper. In the 23rd IEEE International Conference on Distributed Computing Systems (ICDCS 2003), Providence, RI, May 2003.

Ganesan, D., Govindan, R., Shenker, S. and Estrin, D (2001) - *Highly-resilient, Energy-Efficient Multipath Routing in Wireless Sensor Networks* - ACM SIGMOBILE Mobile Computing and Communications Review, 5(4):11–25, 2001.

Karlof, C., Li, Y.; Polastre, J. (2002) - ARRIVE: Algorithm for Robust Routing in Volatile Environments - Technical Report UCB/CSD-03-1233, University of California at Berkeley, May 2002

Karlof, Chris and Wagner, David (2003) - *Secure Routing in Wireless Sensor Networks: Attacks and Countermeasures*, to appear First IEEE International Workshop on Sensor Network Protocols and Applications, May 2003.

Levis, P.; Lee, N.; Welsh, M. and Culler, D. (2003) Tossim: Accurate and scalable simulation of entire tinyos applications, in Proc of the 1st Int'l Conf on Embedded Networked Sensor Systems, 2003

Martins, M. H. T., Silva, A. P. R. da ; Loureiro, A. A. F. and Ruiz, L. B. (2005) *An IDS simulator for wireless sensor networks*. Sensornet Technical Report, Comp Sci Dept, Federal University of Minas Gerais, May 2005.

Oliveira, S.; Wong, H. C.; Nogueira, J. M. (2004) - NEKAP: Estabelecimento de Chaves Resiliente a Intrusos em RSSF – Simpósio Brasileiro de Redes de Computadores, Fortaleza, 2004

Park, S.; Savvides, A. and Srivastava, M. B. (2000) - Sensorsim: A simulation framework for sensor networks, in Proc of the 3rd ACM Int'lWorkshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems, 2000

Paula, W. P. ; Oliveira, S.; Wong, H. C; Nogueira, J. M (2005) - Detecção de Intrusos por Rotas Redundantes em RSSF – Simpósio Brasileiro de Segurança em Sistemas Computacionais – SBSeg 2005, Florianópolis-SC, 2005

Shnayder, V.; Hempstead , M.; Chen , B. Rong; Allen, G. W.; and Welsh, M. (2004), *Simulating the power consumption of large-scale sensor network applications*, in Proc of the 2nd Int'l Conf on Embedded Networked Sensor Systems, 2004

Silva, A. P. R.; Loureiro, A. A. F.; Martins, M. H. T.; Ruiz, L. B.; Rocha, B. P. S., Wong, H. C. (2005) - *Decentralized intrusion detection in wireless sensor networks* - ACM Q2SWinet 2005

Staddon, J.; Balfanz, D.; Durfee, G.; (2002) – *Efficient Tracing of Failed Nodes in Sensor Networks* – WSNA'02, Atlanta, Geórgia, USA

Teixeira, F. A.; Nogueira, J. M., Wong, H. C. (2005) - *Detecção de Intrusos por Observação em Redes de Sensores Sem Fio* – Dissertação defendida junto ao Programa de Pós-graduação em Ciência da Computação da Universidade Federal de Minas Gerais, novembro de 2005

Wood, A. D.; Stankovic, J. A. (2002), *Denial of Service in Sensor Networks* – IEEE Computer, October 2002