Estimando o Nível de Segurança de Dados de Redes de Sensores sem Fio *

Alex Lacerda Ramos[1], **Raimir Holanda Filho**[1]

[1]Programa de Pós-Graduação em Informática Aplicada (PPGIA)
Universidade de Fortaleza (UNIFOR) – 60811-905 – Fortaleza – CE – Brasil

alex.lacerda@unifor.edu.br, raimir@unifor.br

Abstract. *The main purpose of Wireless Sensor Networks (WSN) is generating reliable data to their users. For that reason, sensor networks use a combination of different security mechanisms. However, in order to know whether these mechanisms are providing enough security for sensor data, it is interesting that users have access to a value that informs the security level provided by such mechanisms. Thus, users will be able to quickly decide on whether to use such data. This paper presents the Sensor Data Security Estimator (SDSE), a model to estimate the security level of data from sensor networks based on the existing security mechanisms deployed in them.*

Resumo. *O principal propósito das Redes de Sensores sem Fio (RSSF) é gerar dados confiáveis para seus usuários. Para isso, as redes de sensores utilizam uma combinação de diferentes mecanismos de segurança. No entanto, a fim de saber se estes mecanismos estão provendo segurança suficiente para os dados dos sensores, é interessante que os usuários tenham acesso a um valor que informe o nível de segurança proporcionado por tais mecanismos. Desse modo, os usuários serão capazes de decidir rapidamente sobre a utilização ou não desses dados. Este artigo apresenta o Sensor Data Security Estimator (SDSE), um modelo para estimar o nível de segurança dos dados de redes de sensores com base nos mecanismos de segurança existentes nelas.*

1. Introdução

O principal propósito das RSSF é capturar informações do mundo real e torná-las disponíveis para usuários interessados. Isso é feito pelos nós sensores que, de maneira distribuída, coletam e enviam informações para um dispositivo central, a estação base, para que os usuários possam acessá-las [Akyildiz et al. 2002]. Visto que os usuários de RSSF desejam utilizar informações confiáveis, é imperativo garantir a segurança dessas redes.

Em razão de suas características peculiares, para garantir segurança, as RSSF utilizam a combinação de diferentes mecanismos de segurança, cada um deles projetado para abordar vulnerabilidades específicas. Entretanto, somente a presença dos mecanismos de segurança não garante que os dados dos sensores sejam realmente confiáveis. É necessário portanto, medir o *nível de segurança* dos dados provido por esses mecanismos e informá-lo aos usuários, de modo que eles possam decidir sobre o uso desses dados.

*Esta pesquisa faz parte do Projeto Construindo Cidades Inteligentes [CIA][2] e Alex Lacerda Ramos é financiado por bolsa da CAPES.

Com esse propósito, este artigo apresenta o *Sensor Data Security Estimator* (SDSE), um modelo para calcular dinamicamente o *nível de segurança* de dados originados em RSSF. Para isso, o SDSE define *métricas de segurança* obtidas a partir dos mecanismos de segurança das RSSF. Os valores das métricas coletados a partir dos nós sensores são em seguida combinados para prover um valor global referente à confiabilidade dos dados. O *nível de segurança* representa portanto, a probabilidade dos dados de sensores serem seguros mesmo que a rede esteja sob ataque.

O cálculo do *nível de segurança* também pode ser usado para ajudar profissionais a tomar decisões sobre como avaliar diferentes mecanismos de segurança e modificar as configurações da rede a fim de aperfeiçoar a segurança [Ahmed et al. 2008].

O restante do artigo está organizado como segue. A Seção 2 discute os trabalhos relacionados. A Seção 3 descreve as *métricas de segurança* propostas e os *parâmetros* necessários para seu cálculo. A Seção 4 apresenta a estimativa do *nível de segurança*. A Seção 5 detalha o funcionamento do SDSE. A Seção 6 mostra as avaliações das *métricas* e do *nível de segurança* propostos. Por fim, a Seção 7 conclui o trabalho.

2. Trabalhos Relacionados

A maioria das propostas para estimativa de nível de segurança é baseada em vulnerabilidades e análise de riscos. Ahmed et al. [Ahmed et al. 2008] apresentam uma abordagem para medir nível de segurança baseada em métricas que quantificam vulnerabilidades de *serviços* de rede. Eles utilizam a norma *Common Vulnerability Scoring System* (CVSS) [Ahmed et al. 2008] para quantificar as métricas propostas e utilizam ferramentas automatizadas para identificar as vulnerabilidades existentes. Em seguida, eles combinam os valores das métricas para computar o nível de segurança global.

Frigault et al. [Frigault et al. 2008] propõem um esquema para medir segurança global de rede que utiliza grafos de ataque baseados em uma rede de *Bayes* para combinar os efeitos de todas as vulnerabilidades conhecidas do sistema. Eles usam métricas de probabilidades baseadas no CVSS e as combinam usando probabilidade condicional.

Li et al. [Li et al. 2011] apresentam um modelo estocástico para quantificar a segurança global de redes, de acordo com a capacidade dos mecanismos de segurança implantados e o grafo de vulnerabilidades subjacente.

Embora vulnerabilidades e análise de risco tenham sido abordadas recentemente para redes tradicionais, os trabalhos existentes não podem ser aplicados diretamente em redes de sensores, pois como a segurança de RSSF é uma área imatura, não existe muito sobre vulnerabilidades de serviços e ferramentas para identificá-las. Além disso, as RSSF dependem principalmente da *confiabilidade* e *resiliência* dos mecanismos de segurança para atenuar suas vulnerabilidades, que são *inerentes* e não necessariamente relacionadas a *serviços*, tais como canal de comunicação não confiável e exposição a ataques físicos.

Até o momento, pouquíssimos estimadores de segurança foram propostos para *redes sem fio*, principalmente para redes de sensores. Nesse contexto dinâmico das redes sem fio, Savola [Savola 2008] propõe um *framework* para estimar o nível de segurança de redes móveis sem fio baseado em métricas de segurança apropriadas. Contudo, em seu trabalho as métricas usadas não são mencionadas.

Com relação a RSSF, Ksiezopolski e Kotulski [Ksiezopolski and Kotulski 2005]

apresentam um modelo que usa vários parâmetros de protocolos de rede para estimar o nível de segurança global. No caso deste nível estar abaixo de certo patamar, a segurança do sistema é aperfeiçoada dinamicamente. No entanto, sua proposta supõe que existam nós centrais com maior proteção e mecanismos de validação para detectar incidentes em quaisquer lugares na rede.

Diferentemente dos trabalhos descritos acima, o SDSE define métricas que consideram a *resiliência* e a *confiabilidade* dos mecanismos de segurança, além de serem baseadas em informações que refletem o atual *estado de segurança* das redes, como será apresentado na Seção 3. Também é importante ressaltar que ao invés de estimar a segurança global da rede, este artigo considera somente os sensores pertencentes à rota pela qual os dados trafegaram até chegar à estação base, como será explicado na Seção 4.

3. Métricas de Segurança Propostas

Nesta seção, são apresentados os mecanismos de segurança considerados pelo SDSE e as métricas de segurança propostas para cada um desses mecanismos.

As métricas propostas avaliam os mecanismos de segurança instalados em uma rede de sensores e são calculadas na estação base pelo SDSE para cada nó sensor, a partir de parâmetros providos por esses mecanismos. Existem dois tipos de mecanismos de segurança para RSSF, os de prevenção e os detecção. Os mecanismos de prevenção evitam ou dificultam a execução de certos tipos de ataque, enquanto os mecanismos de detecção são responsáveis por identificar e isolar ataques que violaram os mecanismos de prevenção e passaram a realizar atividades maliciosas na rede.

Levando isso em conta, neste artigo são considerados quatro mecanismos de segurança, que juntos proporcionam uma solução de segurança completa para redes de sensores. Estes mecanismos são: *Criptografia* (prevenção), *Gerenciamento de Chave Criptográfica* (prevenção), *Sistema de Detecção de Intrusão* (detecção) e *Sistema de Gerenciamento de Confiança* (detecção). Vale ressaltar que o SDSE não é responsável por garantir a segurança da RSSF, ele apenas estima o nível da segurança provido pelos mecanismos já instalados na rede.

A seguir são descritas as métricas propostas para cada um dos mecanismos de segurança considerados.

3.1. Probabilidade de Força Criptográfica

Em uma Rede de Sensores sem Fio, adversários podem realizar ataques de força bruta nos algoritmos de criptografia e revelar as chaves secretas dos sensores. Neste tipo de ataque, todas as possíveis chaves são testadas até que se descubra a chave secreta procurada. É importante ressaltar que chaves inseguras podem ser utilizadas em RSSF devido à limitação de seus recursos ou por imperícia do próprio administrador da rede.

Para medir a capacidade de um sensor resistir à este tipo de ataque, definimos a métrica *Probabilidade de Força Criptográfica* (P_F), que varia de acordo com o tempo t, de acordo com a seguinte equação:

$$P_F(t) = \begin{cases} 1 - \dfrac{f \cdot t}{2^s} & \text{se } t \leq \dfrac{2^s}{f} \\ 0 & \text{caso contrário} \end{cases} \quad (1)$$

Onde s representa a força do algoritmo de criptografia, que é uma medida logarítmica do ataque computacional mais rápido conhecido para o algoritmo (medida em bits), t é o intervalo de tempo decorrido desde o momento da ativação da chave no sensor e f é a quantidade de chaves testadas por unidade de tempo.

Observe que P_F é calculada pelo complemento da probabilidade de uma chave ser descoberta, que por sua vez, é calculada pelo total de chaves que podem ser testadas até o tempo t (isto é, $f \cdot t$) dividido pelo total de chaves possíveis (2^s).

Existem diferentes forças criptográficas dependendo do algoritmo e do tamanho de chave utilizados. Normalmente, quanto maior o tamanho da chave (também medida em bits), maior sua força criptográfica. A força de um algoritmo de criptografia é sempre um valor menor ou igual ao tamanho de sua chave. Os valores de força criptográficas para algoritmos simétricos e assimétricos podem ser encontrados em relatórios anuais providos por organizações como NIST [Barker et al. 2012], entre outros.

Da mesma maneira, o valor de f pode ser determinado dependendo do algoritmo utilizado e corresponde ao maior *throughput* publicamente conhecido para dado algoritmo. Por exemplo, a máquina customizada *COPACOBANA RIVYERA* [SciEngines 2008] desenvolvida para quebrar o algoritmo DES (*Data Encryption Standard*) tem um *throughput* $f = 292$ bilhões de chaves por segundo e pode encontrar uma chave DES em menos de um dia [SciEngines 2008].

O SDSE é responsável por registrar e gerenciar os nomes dos algoritmos de criptografia e seus respectivos parâmetros f e s.

3.2. Probabilidade de Resiliência do Gerenciamento de Chave

Outra forma de descobrir chaves secretas é capturar fisicamente um sensor e violá-lo para ganhar acesso às chaves. O mecanismo responsável por abordar esse tipo de ataque é o *Gerenciamento de Chave*. Como os sensores compartilham chaves, a captura de um nó pode comprometer a segurança da comunicação entre outros nós. Portanto, a *Probabilidade de Resiliência do Gerenciamento de Chave* (P_R) calcula a capacidade de resistência do esquema de gerenciamento a um número x de nós capturados por adversários.

Desse modo, P_R representa a probabilidade de que um *link* de comunicação entre dois sensores quaisquer não capturados se mantenha seguro mesmo que x nós tenham sido capturados. Essa métrica é calculada da seguinte maneira:

$$P_R(x) = 1 - P_C(x) \tag{2}$$

Onde $P_C(x)$ representa a probabilidade de um *link* ser comprometido quando x nós são capturados e é definida pela probabilidade condicional $P_C(x) = P\{L_c|C_x\}$. Onde L_c é o evento de um *link* ser comprometido e C_x é o evento de x sensores serem capturados.

Para cada esquema de gerenciamento de chave, existe uma equação para calcular $P_C(x)$. Existem vários tipos de esquemas de gerenciamento para RSSF, como esquemas de chave única para toda a rede, esquemas de estabelecimento de chaves aos pares e esquemas de predistribuição de chaves. O SDSE é responsável por registrar e gerenciar os nomes dos esquemas de gerenciamento e suas respectivas equações de $P_C(x)$.

Por exemplo, no esquema de predistribuição de chave aleatória *q-composite* [Chan et al. 2003], antes da implantação da rede, cada nó escolhe k chaves de um *pool*

de chaves de tamanho $|K|$. Após a implantação, dois nós podem estabelecer um *link* seguro somente se tiverem pelo menos q chaves em comum, isto é, dois nós tem um *link* de comunicação seguro quando compartilham i chaves, sendo $q \leq i \leq k$.

Para o esquema *q-composite*, Chan et al. [Chan et al. 2003] mostraram que $P_C(x)$ é calculado pela seguinte equação:

$$P_C(x) = \sum_{i=q}^{k} \left(1 - \left(1 - \frac{k}{|K|}\right)^x\right)^i \frac{p(i)}{p} \tag{3}$$

Onde $p(i)$ é a probabilidade de um *link* ser estabelecido com i chaves e é definida por Chan et al. como: $p(i) = \frac{\binom{|K|}{i}\binom{|K|-i}{2(k-i)}\binom{2(k-i)}{k-i}}{\binom{|K|}{k}^2}$ e $p = p(q) + p(q+1) + \ldots + p(k)$, isto é, p representa a probabilidade de dois nós estabelecerem um *link* seguro.

Os valores de k, q e p são dados pelo próprio esquema de gerenciamento. A partir desses valores, é possível calcular o valor de $|K|$. Porém, para calcular $P_C(x)$, é necessário ainda obter o valor de x. Nesse caso, o valor de x pode ser facilmente obtido através do *Sistema de Detecção de Intrusão* presente na rede.

3.3. Probabilidade de Legitimidade

O Sistema de Detecção de Intrusão (IDS - *Intrusion Detection System*) de uma rede de sensores analisa o tráfego de rede em busca de atividades maliciosas e é capaz de identificar e isolar nós sensores que realizam ataques. Devido à natureza distribuída das redes de sensores, neste artigo são considerados os sistemas de detecção distribuídos e colaborativos [Farooqi and Khan 2009].

Existem vários IDSs distribuídos e colaborativos propostos para redes de sensores [Farooqi and Khan 2009]. Nestes mecanismos, cada nó sensor monitora sua vizinhança à procura de comportamento suspeito. Assim que uma atividade maliciosa é detectada, nós vizinhos trocam informações sobre o nó suspeito. Neste processo de colaboração, cada sensor vizinho de um nó suspeito deve indicar seu ponto de vista (v) em relação a esse nó, que pode ser $v = 1$ para indicar nó malicioso e $v = 0$ para indicar nó legítimo.

Ao final da colaboração, um nó é considerado pelo IDS como malicioso quando pelo menos m de seus N vizinhos indicaram que ele está realizando atividade maliciosa, isto é, indicaram $v = 1$. Detalhes dos métodos de detecção e processo de colaboração podem ser encontrados em [Farooqi and Khan 2009].

Para estimar a segurança provida por um IDS, definimos uma métrica chamada de *Probabilidade de Legitimidade* (P_L) que calcula a chance de um sensor ser legítimo. Essa métrica avalia a *confiabilidade* do IDS, pois é baseada nas taxas de falsos positivos e verdadeiros negativos de cada vizinho, isto é, taxas de acerto e erro do IDS causados por seu método de detecção. Além disso, essa métrica também avalia a *resiliência* do IDS a ataques ao seu processo de colaboração, pois considera o caso em que os nós vizinhos realizam falsas indicações por também estarem comprometidos. Ademais, como veremos a seguir, essa métrica é calculada de modo diferente, dependendo da *conclusão* do IDS sobre um nó em questão, o que reflete o atual *estado de segurança* da rede.

Considere P_f a *taxa de falsos positivos* de cada vizinho, N a quantidade de vizinhos do nó em questão e m a quantidade mínima de nós vizinhos necessária para que o IDS conclua que o nó em questão é malicioso. Considerando o caso em que o nó em questão tenha sido detectado como *malicioso* pelo IDS, definimos a *Probabilidade de Legitimidade* (P_L) como a probabilidade de pelo menos m vizinhos terem errado ao detectar o nó como malicioso, por ele tratar-se, na verdade, de um nó legítimo:

$$P_L = \sum_{j=m}^{N} \binom{N}{j} \left(\frac{P_f}{2}\right)^j \left(1 - \frac{P_f}{2}\right)^{N-j} \tag{4}$$

A taxa de falsos positivos P_f é calculada pela razão entre a quantidade de atividades normais incorretamente marcadas como intrusões e o número total de atividades normais da rede.

Observe ainda que P_f é multiplicado por $\frac{1}{2}$ para representar a probabilidade de resiliência da detecção. Para compreender a utilização de $\frac{1}{2}$, considere o caso em que os vizinhos podem estar comprometidos e, portanto, realizando falsas indicações. Assim, a probabilidade do vizinho estar comprometido (ou não) é de 50%. Desse modo, utilizamos $\frac{1}{2}$ no cálculo de P_L para representar a resiliência do IDS a falsas indicações de vizinhos.

A outra forma de calcular P_L é utilizada quando o IDS conclui que o nó é legítimo. Neste caso a chance do nó ser realmente legítimo é dado pela probabilidade do IDS ter acertado em sua conclusão. Portanto, para calcular P_L, definimos a seguinte equação:

$$P_L = \sum_{j=N-m+1}^{N} \binom{N}{j} \left(\frac{P_n}{2}\right)^j \left(1 - \frac{P_n}{2}\right)^{N-j} \tag{5}$$

Onde P_n é a *especificidade* ou *taxa de verdadeiros positivos* de cada nó vizinho, definida como a razão entre as atividades normais corretamente marcadas como normais e o número total de atividades normais da rede. Observe que na equação 5, para que o nó não seja classificado como malicioso, deve haver no mínimo $N - m + 1$ nós vizinhos indicando que o nó em questão não é malicioso, isto é, indicando $v = 0$.

Para o cálculo de P_L, o valor de m é fornecido pelo próprio IDS e o valor de N pode ser facilmente obtido pelo algoritmo de roteamento da rede, já os valores de P_f e P_n devem ser previamente estabelecidos a partir de análise estatística do comportamento do IDS ou simulações que testam o comportamento do IDS em um ambiente controlado. Geralmente, as taxas para determinados cenários e tipos de redes são dadas nos próprios artigos em que os IDSs são propostos. O SDSE é responsável por registrar e gerenciar os nomes dos IDSs e suas respectivas taxas P_f e P_n.

3.4. Probabilidade de Entrega

Os mecanismos de *Gerenciamento de Confiança* são usados para medir a confiabilidade de nós sensores de acordo com seu comportamento na rede. Confiança é um relacionamento de três partes, que pode ser expresso como "entidade A confia na entidade B para fazer X" [Shaikh et al. 2009]. Em redes de sensores, a parte "fazer X" normalmente se refere ao repasse (entrega) de pacotes e as entidades A e B referem-se aos a nós sensores.

Uma abordagem comum para se avaliar a confiabilidade de um nó b para um nó a é calcular a probabilidade de b entregar pacotes provenientes de a de maneira satisfatória. O valor de confiança de um nó pode ser calculado a partir de interações diretas ou recomendações de vizinhos confiáveis sobre um nó especifico. De modo geral, o valor de confiança de um nó a em um nó b pode ser expresso por: $T_{a,b} = \frac{S_{a,b}}{S_{a,b} + U_{a,b}}$. Onde $S_{a,b}$ é o número total de interações bem sucedidas do nó a com o nó b e $U_{a,b}$ é o número total de interações mal sucedidas do nó a com o nó b. Neste caso, o valor de confiança é um valor entre 0 e 1 (inclusive). Contudo, cada mecanismo de Gerenciamento de Confiança pode ter valores de confiança em diferentes intervalos, tais como $[-1, 1]$ e $[0, 100]$, além de poder utilizar uma forma modificada da equação $T_{a,b}$ acima.

Por exemplo, Shaikh et al. [Shaikh et al. 2009] propõem um mecanismo de Gerenciamento de Confiança chamado GTMS que calcula $T_{a,b}$ com a seguinte equação:

$$T_{a,b} = \left[100 \left(\frac{S_{a,b}}{S_{a,b} + U_{a,b}} \right) \left(1 - \frac{1}{S_{a,b} + 1} \right) \right] \tag{6}$$

Onde $[\cdot]$ é a função do inteiro mais próximo e $S_{a,b}$ e $U_{a,b}$ são calculados para um intervalo de tempo Δt. Observe que o valor de $T_{a,b}$ está no intervalo $[0, 100]$.

Para o SDSE, definimos a métrica *Probabilidade de Entrega* (P_E) como o valor de confiança fornecido pelo mecanismo de Gerenciamento de Confiança convertido para o intervalo $[0, 1]$, como mostra a seguinte equação de normalização:

$$P_E = \frac{T_{a,b} - T_{min}}{T_{max} - T_{min}} \tag{7}$$

Onde T_{min} e T_{max} representam respectivamente o valor mínimo e o valor máximo de confiança possíveis para um nó. No caso do GTMS, $T_{min} = 0$ e $T_{max} = 100$.

Essa métrica provê dinamicamente o atual *estado de segurança* de cada nó. Quanto maior for o valor de confiança de um nó, mais seguro ele será e, consequentemente, os dados que ele repassa.

É importante ressaltar que o SDSE não calcula o valor de confiança de um nó, pois isso já é feito pelo mecanismo de gerenciamento de confiança instalado na rede. Em nosso exemplo, o GTMS utiliza a equação 6 para calcular o valor de confiança e o SDSE apenas normaliza esse valor para o intervalo [0,1]. Portanto, o SDSE é responsável por registrar e gerenciar os nomes dos esquemas de gerenciamento de confiança e seus respectivos intervalos $[T_{min}, T_{max}]$.

Apesar da avaliação da entrega de pacotes também ser feita pelo IDS, o gerenciamento de confiança realiza esta tarefa de maneira mais precisa, visto que ele obtém maiores taxas de acerto por ser uma solução específica para abordar ataques à entrega de pacotes. Por isso, quando há um ataque relacionado à entrega de pacotes, mesmo que o IDS o detecte, nós utilizamos a métrica do IDS (P_L, eq. 5) como se não tivesse ocorrido esse tipo de ataque, visto que ele já foi abordado pelo gerenciamento de confiança. Por outro lado, para os inúmeros ataques que não se relacionam à entrega de pacotes, nós utilizamos a métrica P_L da eq. 4.

4. Cálculo do Nível de Segurança

O Nível de Segurança (SL - *Security Level*) é um valor no intervalo $[0, 1]$ atribuído para cada dado originado em um sensor para indicar o quão seguro esse dado é. Neste caso, *dado* de um sensor refere-se às leituras realizadas pelo sensor retornadas como resposta a uma consulta de dados solicitada na estação base.

Assim que uma resposta chega à estação base, o SDSE obtém os parâmetros dos mecanismos de segurança e do mecanismo de roteamento. Na próxima seção, discutiremos o modo utilizado pelo SDSE para obter os parâmetros dos mecanismos de segurança.

Após obtenção dos valores dos parâmetros, o SDSE calcula as métricas (definidas na Seção 3) para cada nó pertencente à rota pela qual a resposta de consulta passou até chegar à estação base. Depois, o SDSE utiliza os valores das métricas para calcular o SL.

Formalmente, para a rota $R = \{n_i \mid i = 1, 2, \ldots, q\}$ em questão, o SDSE inicialmente calcula para cada nó n_i pertencente à rota R, o grau de segurança (g_i) do nó n_i como o produto das métricas calculadas para este nó, visto que cada métrica corresponde a eventos independentes entre si, isto é, o valor de uma métrica não influencia o valor das outras. Esse cálculo é mostrado na seguinte equação:

$$g_i = P_F^i \times P_R^i \times P_L^i \times P_E^i \tag{8}$$

Dessa maneira, obtém-se o conjunto $G = \{g_i \mid i = 1, 2, \ldots, q\}$ contendo os graus de segurança de todos os nós da rota R. Como o nó com menor grau de segurança da rota representa o elo mais fraco e tem maiores chances de fazer com que o dado deixe de ser confiável, o SL é calculado como o mínimo valor de G, como mostra a seguinte equação:

$$SL = \min(g_1, g_2, \ldots, g_q) \tag{9}$$

Por fim, o SDSE atribui o SL calculado à respectiva resposta da consulta e a entrega aos usuários interessados.

5. Funcionamento do SDSE

Nesta seção, apresentamos todos os detalhes do funcionamento do SDSE, desde a geração da consulta pelo usuário, até a entrega da resposta juntamente com seu respectivo SL.

Antes de apresentarmos a consulta, é importante deixar claro que o SDSE possui previamente armazenados vários nomes de mecanismos de segurança e os respectivos valores de seus parâmetros estáticos, tais como os parâmetros de criptografia f (quantidade de chaves testadas por unidade de tempo) e s (força criptográfica), por se tratarem de valores fornecidos por fontes externas, tais como o NIST [Barker et al. 2012].

Além disso, como o SDSE fica instalado na estação base como uma aplicação do *TinyOS* [Karlof et al. 2004], sistema operacional dos nós sensores, ele é capaz de se comunicar com o TinyOS e obter o nome dos algoritmos instalados na rede. Com essa informação, ele pode ler sua base de dados e carregar na memória os parâmetros estáticos para os nomes de mecanismos fornecidos pelo TinyOS.

No entanto, o SDSE também precisa solicitar dos respectivos mecanismos de segurança, os seus parâmetros dinâmicos, tais como os parâmetros do gerenciamento de

chave q (número mínimo de chaves em comum para estabelecer um *link* seguro entre dois nós) e k (número de chaves armazenadas por cada nó). Para isso, o SDSE precisa se comunicar diretamente com esses mecanismos.

A comunicação direta entre o SDSE e os mecanismos da rede é feita através dos módulos e interfaces providos pelas aplicações executadas no *TinyOS*. Portanto, o SDSE verifica em sua base de dados quais métodos públicos são oferecidos pelos mecanismos da rede. De posse dessa informação, o SDSE está pronto para realizar chamadas aos métodos fornecidos pelas interfaces dos mecanismos e obter seus respectivos parâmetros dinâmicos quando necessário.

Entretanto, para que o SDSE consiga ter acesso aos parâmetros dos mecanismos, é preciso que esses mecanismos tenham um módulo na estação base capaz de fornecer tais informações. Essa é uma condição válida, visto que a maioria dos mecanismos de RSSF possui um módulo na estação base que gerencia seu funcionamento, como é o caso dos mecanismos de segurança abordados neste artigo (criptografia, gerenciamento de chave, detecção de intrusão e gerenciamento de confiança). Isto também se aplica aos algoritmos de roteamento, pois em vários deles a árvore de roteamento é gerada na estação base, como mostram Radi et al. [Radi et al. 2012]. Isto lhes permite conhecer todas as rotas da rede e a quantidade de vizinhos de cada sensor, como é o caso do roteamento *Directed Diffusion* [Radi et al. 2012], que é próprio para aplicações orientadas a consultas.

Dito isto, podemos apresentar o funcionamento do SDSE a partir do momento da geração da consulta. Para ilustrar, utilizaremos uma consulta no formato SQL que pode ser interpretada por aplicações como *TinyDB* e *Cougar* [Gehrke and Madden 2004].

Suponhamos então que um usuário W deseje que o nó n_6 envie algumas leituras para a estação base de 5 em 5 segundos, durante 1 minuto. Então ele realiza a seguinte consulta na estação base: ***SELECT temperatura, umidade, pressão, luminosidade FROM sensors s WHERE s.nodeid = 6 SAMPLE PERIOD 5s FOR 60s***.

Essa consulta é então disseminada pela rede até chegar a n_6. No momento em que este nó recebe a consulta, ele realiza as leituras necessárias e envia as respostas à estação base nos tempos determinados. Suponhamos ainda que as respostas enviadas por n_6 sejam encaminhadas pelos nós n_5, n_4, n_3, n_2 e n_1 até chegar à estação base. Neste caso, temos $R = \{n_1, n_2, n_3, n_4, n_5, n_6\}$.

Assim que cada resposta é recebida pela estação base, o SDSE inicia o processo para o cálculo do SL dessa resposta. Inicialmente, o SDSE solicita ao algoritmo de roteamento os $nodeids$ dos nós da rota R. Então, o SDSE realiza os quatro passos a seguir, para obter os parâmetros das métricas de segurança de cada nó de R.

Primeiro, o SDSE obtém o parâmetro t de cada nó e calcula a *Probabilidade de Força Criptográfica* (P_F), visto que ele já possui em memória os parâmetros estáticos f e s (equação 1) necessários para calcular P_F. Veja que os valores de f e s são os mesmos para todos os sensores da rede.

Segundo, o SDSE solicita ao mecanismo de detecção de intrusão a quantidade de nós capturados da rede (parâmetro dinâmico x) e calcula a *Probabilidade de Resiliência do Gerenciamento de Chave* (P_R), visto que ele já possui em memória os valores dos parâmetros estáticos k, q e p do gerenciamento de chave (que neste exemplo é o esquema

q-composite), necessários para calcular P_R (ver equações 2 e 3). Observe que os valores de k, q, p e x são os mesmos para todos os sensores da rede.

Terceiro, o SDSE obtém do mecanismo de detecção de intrusão sua conclusão em relação à cada nó de R. Para os nós indicados como maliciosos pelo IDS, o SDSE deve utilizar a *Probabilidade de Legitimidade* (P_L) para nós maliciosos (equação 4). Para os nós indicados como legítimos pelo IDS, o SDSE deve calcular a *Probabilidade de Legitimidade* (P_L) para nós legítimos (equação 5). Após descobrir qual equação deve ser utilizada, o SDSE solicita ao algoritmo de roteamento o valor do parâmetro dinâmico N (quantidade de vizinhos) de cada nó. Em seguida, o SDSE calcula a *Probabilidade de Legitimidade* (P_L), uma vez que ele já possui os valores dos parâmetros estáticos m, P_f e P_n. Observe que os valores desses parâmetros são os mesmos para cada sensor. Diferentemente dos valores do parâmetro N, que são diferentes para cada sensor.

Quarto, o SDSE obtém do gerenciamento de confiança (que neste exemplo é o esquema GTMS [Shaikh et al. 2009]), o valor de confiança (ver equação 6) de cada nó pertencente a R. Com esses valores, o SDSE calcula a *Probabilidade de Entrega* (P_E) de cada nó por meio da normalização desses valores para o intervalo $[0, 1]$ (equação 7).

Após calcular as métricas de segurança para cada nó, o SDSE segue para o cálculo do SL. Para isso, ele calcula o grau de segurança g_i (equação 8) de cada nó de R e em seguida, obtém o SL (equação 9), que é o menor valor g_i encontrado entre os nós de R.

Por fim, o usuário W recebe a resposta para sua consulta juntamente com seu respectivo *Nível de Segurança* (SL). Supondo-se que o nível de segurança calculado para essa consulta tenha sido $SL = 0.9$, o usuário W pode interpretá-lo da seguinte maneira: *a resposta para minha consulta possui 90% de chance de estar correta, isto é, de ser uma resposta confiável, que não tenha sofrido ataques nem tenha sido adulterada indevidamente.* Em outras palavras, ela é um dado que possui 90% de chance de ser seguro.

6. Análise das Métricas e do Nível de Segurança

Nesta seção, o comportamento dos valores das métricas e do nível de segurança é analisado de acordo com variados valores de parâmetros dos mecanismos de segurança.

6.1. Probabilidade de Força Criptográfica

Para mostrar o comportamento de P_F, escolhemos o algoritmo de criptografia RC5 [Karlof et al. 2004], por ser um algoritmo bastante conhecido e ser implementado pelo TinySec [Karlof et al. 2004] (arquitetura de segurança implementada no TinyOS).

O RC5 possui um tamanho de chave variável (0 a 2040 bits). A organização *Distributed.net* [Distributed.net 2002] possui projetos para quebra das chaves do RC5 que chegam à uma taxa $f = 394.254.429.396$ chaves por segundo. A Figura 1(a) apresenta os valores de P_F variando com o tempo t, para diferentes valores de força criptográfica s.

Observe na Figura 1(a) que quanto maior a força criptográfica de um algoritmo, mais seguro ele será. Por outro lado, à medida que o tempo passa, a *Probabilidade de Força Criptográfica* diminui, visto que mais chaves podem ser testadas, o que aumenta as chances de um adversário quebrar o algoritmo.

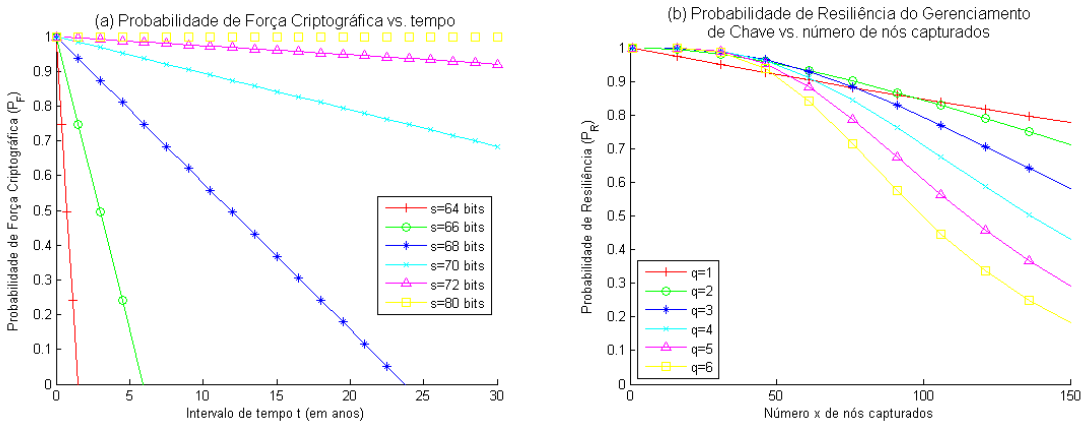

Figura 1. (a) Probabilidade de Força Criptográfica (P_F) em relação ao tempo t, para diferentes valores de s e taxa $f = 394.875.793.722$ chaves/segundo. (b) Probabilidade de Resiliência do Gerenciamento de Chave quando um adversário captura x nós da rede, para diferentes valores de q, para $k = 200$ e $p = 0,33$.

6.2. Probabilidade de Resiliência do Gerenciamento de Chave

Para apresentar o comportamento de P_R, escolhemos o esquema *q-composite* [Chan et al. 2003], por ser um mecanismo bastante difundido e que serve de comparação para vários outros mecanismos de gerenciamento de chave para RSSF.

A Figura 1(b) apresenta os valores de P_R quando x nós são capturados, para diversos valores de q (quantidade mínima de chaves necessárias para estabelecer um link seguro entre dois nós). Nesta figura, temos o número de chaves armazenada por cada nó $k = 200$ e a probabilidade de estabelecimento de um *link* seguro $p = 0,33$. A partir dos valores de k e p, o tamanho do *pool* de chaves $|K|$ é obtido para cada valor de q.

Na Figura 1(b), podemos perceber que até por volta de 50 nós capturados, a *Probabilidade de Resiliência* se mantém bastante próxima de 1, independentemente do valor de q. Entretanto, a partir de 50 nós capturados, P_R diminui de maneira mais significativa, além de ser menor para maiores valores de q.

6.3. Probabilidade de Legitimidade

Para mostrar o comportamento de P_L, escolhemos o IDS distribuído e cooperativo proposto por Krontiris et al. [Krontiris et al. 2009], por ser um IDS implementado e testado no *TinyOS* e por servir de base para vários outros IDSs colaborativos para RSSF.

A Figura 2(a) apresenta o comportamento de P_L para nós detectados como maliciosos pelo IDS e a figura 2(b) mostra o comportamento de P_L para nós apontados como não maliciosos pelo IDS. Nessas figuras, P_L é calculado para diferentes valores de P_f (taxa de falsos positivos de um nó) e P_n (taxa de verdadeiros negativos de um nó), respectivamente. A quantidade de vizinhos utilizada é $N = 10$.

Observe que na Figura 2(a), P_L diminui à medida que o parâmetro de consenso m aumenta, isto é, à medida que mais nós vizinhos são necessários para detectar um nó como malicioso. Isso acontece quando os valores de P_f são menores que 0.5, isto é, quanto mais nós forem necessários para detectar seu vizinho como malicioso, mais serão os casos em que as probabilidades baixas de P_f precisarão ocorrer. Isso também significa

que, quanto maior P_f, maior a *Probabilidade de Legitimidade*.

Já na Figura 2(b), como os valores de P_n são maiores que 0.5, a *Probabilidade de Legitimidade* aumenta à medida que m aumenta. Da mesma maneira, quanto maior P_n, maior a *Probabilidade de Legitimidade*. Observe ainda que grande parte dos valores de P_L da figura 2(b) é maior que os valores de P_L da figura 2(a), isto acontece porque um nó apontado como legítimo pelo IDS, possui muito mais chances de ser realmente legítimo do que um nó apontado como malicioso.

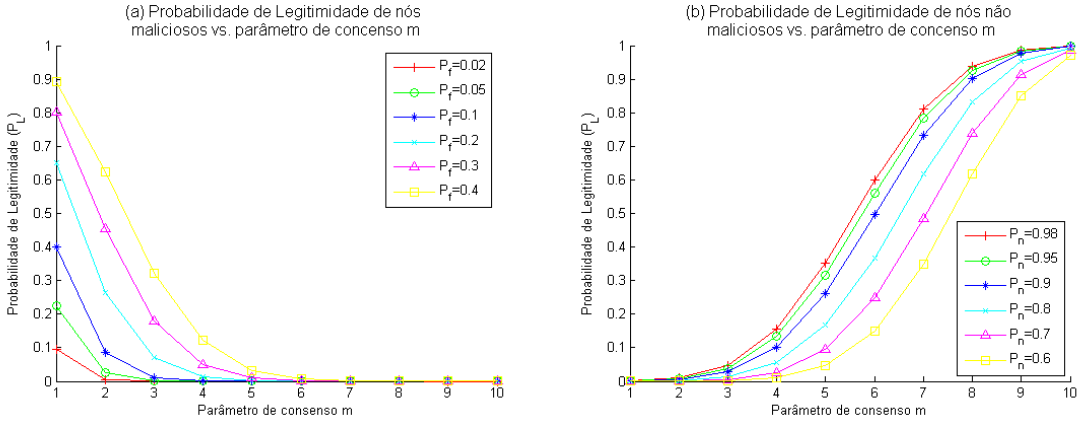

Figura 2. (a) Probabilidade de Legitimidade de nós maliciosos em relação a m, para diferentes valores de P_f e $N = 10$. (b) Probabilidade de Legitimidade de nós não maliciosos em relação a m, para diferentes valores de P_n e $N = 10$.

6.4. Probabilidade de Entrega

Para apresentar o comportamento de P_E, selecionamos o esquema de gerenciamento de confiança GTMS, por se tratar de um mecanismo bastante conhecido e estar entre os melhores mecanismos de gerenciamento de confiança para RSSF.

A Figura 3(a) apresenta os valores de P_E para variadas quantidades de interações bem sucedidas entre dois nós (S). Nesta figura, o eixo x representa a fração de interações bem sucedidas, isto é, a expressão $(\frac{S}{S+U})$ da equação 6.

Na Figura 3(a), vemos que a *Probabilidade de Entrega* aumenta à medida que a fração de interações bem sucedidas cresce. Além disso, quando o número de interações bem sucedidas é $S = 5$, P_E é bem menor do que para os outros valores de S, que possuem praticamente o mesmo P_E para frações de interações bem sucedidas entre 0 e 0.4. No entanto, para frações a partir de 0.4, P_E é ligeiramente maior para maiores valores de S.

6.5. Nível de Segurança

Para compreendermos o comportamento do *Nível de Segurança*, de acordo com as métricas de segurança utilizadas, a Figura 3(b) mostra como o *Nível de Segurança* varia considerando-se a quantidade de vizinhos de um nó (N) e o número x de nós capturados na rede. Para gerar a Figura 3(b), foram utilizados os mesmos algoritmos das subseções anteriores com os seguintes valores de parâmetros:

- **Criptografia - RC5.** $f = 394, 254, 429, 396$ chaves/s, $t = 3$ anos, $s = 80$ bits;
- **Ger. de Chave - *q-composite*.** $k = 200$ chaves, $q = 2$ chaves, $p = 0, 33$;

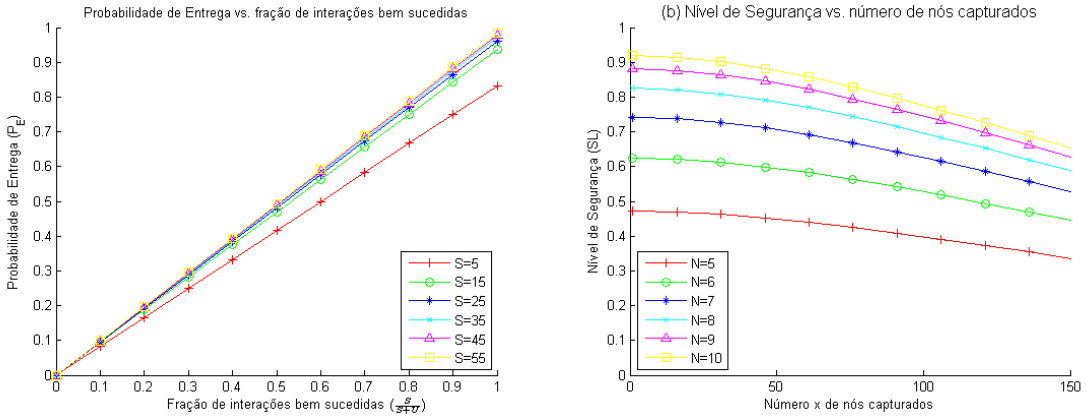

Figura 3. (a) Probabilidade de Entrega (P_E) em relação à fração de interações bem sucedidas ($\frac{S}{S+U}$). (b) Nível de Segurança de um nó para diferentes quantidades de vizinhos (N), quando um adversário captura x nós da rede.

- **IDS de Krontiris et al.** $m = N - 2$ nós, $P_n = 0,98$;
- **Gerenciamento de Confiança - GTMS.** $T = 0,98$.

Na Figura 3(b), é possível visualizar como o *Nível de Segurança* diminui à medida que a quantidade de nós capturados aumenta, o que já era de se esperar, visto que é o mesmo comportamento da *Probabilidade de Resiliência do Gerenciamento de Chave* (P_R), que é afetada pelo valor de x. Por outro lado, quanto maior a quantidade de vizinhos de um nó, maior será seu *Nível de Segurança*. Isto acontece porque um número maior de vizinhos de um nó gera uma maior *Probabilidade de Legitimidade* (P_L). Portanto, a partir dessa figura, podemos perceber como a alteração de parâmetros referentes a métricas distintas influenciam o valor final do SL.

7. Conclusões

Este artigo apresentou o *Sensor Data Security Estimator* (SDSE), um estimador do grau de confiabilidade dos dados de RSSF que utiliza métricas calculadas a partir de parâmetros dos mecanismos de segurança. Com a utilização do *nível de segurança*, os usuários podem tomar decisões informadas quanto ao uso dos dados dos sensores.

A avaliação apresentada neste artigo nos permitiu analisar o comportamento das métricas e do *Nível de Segurança* perante diferentes valores de parâmetros dos mecanismos. Isso permite que o usuário tenha conhecimento do que afeta a segurança dos dados de sua rede, além de permitir que profissionais de segurança utilizem o modelo proposto para selecionar valores de parâmetros adequados para garantir maior segurança.

Além disso, a proposta mostrou-se viável, uma vez que foram mostrados exemplos de como extrair efetivamente os parâmetros dos mecanismos. Atualmente a base de dados do SDSE está senso construída e já possui vários mecanismos, parâmetros e seus respectivos métodos de obtenção registrados.

Para avaliar a dinâmica do modelo proposto, estamos atualmente com um trabalho em andamento que inclui uma simulação detalhada do SDSE em uma rede com diferentes mecanismos de segurança implementados. Esperamos que estes resultados estejam disponíveis em breve em nosso próximo artigo.

Referências

Ahmed, M. S., Al-Shaer, E., and Khan, L. (2008). A Novel Quantitative Approach For Measuring Network Security. In *2008 IEEE INFOCOM - The 27th Conference on Computer Communications*, pages 1957–1965. IEEE.

Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). A survey on sensor networks. *IEEE Communications Magazine*, 40(8):102–114.

Barker, E., Barker, W., Burr, W., Polk, W., and Smid, M. (2012). Recommendation for Key Management - Part 1 : General (Revision 3). *Technical Report*, 1(July):1–147.

Chan, H., Perrig, A., and Song, D. (2003). Random key predistribution schemes for sensor networks. In *Proceedings of the 2003 IEEE Symposium on Security and Privacy*, pages 197–. IEEE Computer Society.

Distributed.net (2002). distributed.net - RC5-72 Overall Project Stats.

Farooqi, A. and Khan, F. (2009). Intrusion detection systems for wireless sensor networks: A survey. In *Communication and Networking*, volume 56, pages 234–241. Springer Berlin Heidelberg.

Frigault, M., Wang, L., Singhal, A., and Jajodia, S. (2008). Measuring network security using dynamic bayesian network. In *Proceedings of the 4th ACM workshop on Quality of protection*, pages 23–30. ACM.

Gehrke, J. and Madden, S. (2004). Query processing in sensor networks. *IEEE Pervasive Computing*, 3:46–55.

Karlof, C., Sastry, N., and Wagner, D. (2004). Tinysec: a link layer security architecture for wireless sensor networks. In *Proceedings of the 2nd international conference on Embedded networked sensor systems*, pages 162–175, New York, NY, USA. ACM.

Krontiris, I., Benenson, Z., Giannetsos, T., Freiling, F. C., and Dimitriou, T. (2009). Cooperative intrusion detection in wireless sensor networks. In *Proceedings of the 6th European Conference on Wireless Sensor Networks*, pages 263–278. Springer-Verlag.

Ksiezopolski, B. and Kotulski, Z. (2005). On scalable security model for sensor networks protocols. In *22nd CIB-W78 Conference Information Technology in Construction (CIB- W78)*, pages 463–469.

Li, X., Parker, P., and Xu, S. (2011). A stochastic model for quantitative security analyses of networked systems. *IEEE Trans. Dependable Secur. Comput.*, 8(1):28–43.

Radi, M., Dezfouli, B., Bakar, K. A., and Lee, M. (2012). Multipath routing in wireless sensor networks: Survey and research challenges. *Sensors*, 12(1):650–685.

Savola, R. (2008). Holistic Estimation of Security, Privacy and Trust in Mobile Ad Hoc Networks. In *2008 3rd International Conference on Information and Communication Technologies: From Theory to Applications*, pages 1–6, Damascus. Ieee.

SciEngines (2008). Break DES in less than a single day (COPACOBANA RIVYERA).

Shaikh, R. A., Jameel, H., d'Auriol, B. J., Lee, H., Lee, S., and Song, Y.-J. (2009). Group-based trust management scheme for clustered wireless sensor networks. *IEEE Trans. Parallel Distrib. Syst.*, 20(11):1698–1712.