IMDGuard: S cvo Eu esteu Dispositivos Médicos Antable com o guardião wearable externo

Fengyuan Xu *, Zhengrui Qin *, Chiu C. Tan †, Baosheng Wang ‡, e Qun Li *

***Resumo*—Estudos recentes revelaram vulnerabilidades de segurança em dispositivos médicos implantáveis (IMDs). Projeto de segurança para IMDs é complicado pela exigência de que os IMDs permaneçam operacionais em uma emergência, quando as credenciais de segurança adequadas podem ser indisponível. Neste artigo, apresentamos o IMDGuard, um abrangente amplo esquema de segurança para IMDs relacionados ao coração para atender a esse requisito. O IMDGuard incorpora duas técnicas adaptadas para fornecer proteções desejáveis para os IMDs. Um é um estabelecimento de chave baseado em ECG sem segredos compartilhados anteriores, e o outro é um mecanismo de controle de acesso resiliente a ataques de spoofing do adversário. A segurança e o desempenho do IMDGuard são avaliados em nossa implementação de protótipo.**

Eu eu NTRODUÇÃO

Os rápidos avanços da bioengenharia estão introduzindo uma explosão de IMDs acessíveis sem fio. Milhões de pacientes experimentam os benefícios dos IMDs na regulação do ritmo cardíaco, controle da pressão arterial, melhora da audição, visão visual e assim por diante. Em um futuro próximo, espera-se que os IMDs estejam cientes da Internet e se tornem um componente crucial em sistemas generalizados, como casas inteligentes e hospitais, tornando a segurança dos IMDs importante. Os pesquisadores identificaram que os IMDs estão enfrentando ameaças potenciais à segurança que podem causar consequências fatais. Investigações recentes sobre marcapassos [8] revelaram vulnerabilidades de segurança em ofertas comerciais existentes que permitem, entre outros ataques, uma entidade maliciosa reprogramar o IMD. Portanto,

Ao contrário dos sistemas embarcados convencionais, a segurança da engenharia em IMDs apresenta um desafio único. Mecanismo de segurança que impõe proteção o tempo todo pode levar a problemas quando a segurança supera as operações seguras de IMDs. Para ilustrar, considere um médico do pronto-socorro, que não é reconhecido como operador legítimo em termos de segurança, pode ter que acessar o IMD para salvar a vida do paciente em uma situação de emergência. A autorização temporária para o médico não é uma solução confiável, uma vez que o proprietário do IMD, nesta circunstância, pode ser fisicamente incapaz de fazer isso ou uma autoridade confiável remota não está disponível.

Intuitivamente, queremos um mecanismo de segurança semelhante a um LIGADO DESLIGADO mude para controlar as proteções de segurança. O interruptor pode ser acionado FORA em caso de emergência, sem assistência do paciente, mas em uma situação não emergencial, o paciente tem controle sobre quem tem acesso ao seu dispositivo. (Isso abre o IMD para atacar em caso de emergência, mas em condições de risco de vida superar tais preocupações.) Os pesquisadores têm defendido o emparelhamento do IMD com um dispositivo externo para fornecer segurança ao IMD, onde em caso de emergência, os médicos podem simplesmente remover o dispositivo externo e continuar a interagir com o IMD sem maiores obstáculos.

Existem dois desafios ao usar um dispositivo externo para proteger o IMD. Primeiro, o dispositivo externo e o IMD devem ter um meio de estabelecer um segredo *sem conhecimento prévio.*
Em outras palavras, não devemos pré-implantar nenhum segredo dentro do IMD. Isso evita situações em que o usuário não consegue lembrar o segredo pré-implantado e precisa redigitar o IMD. A solução convencional é usar um interruptor manual para “reinicializar” o IMD, mas como o IMD é implantado dentro do corpo do paciente, esta solução é inadequada. O segundo desafio é ter um método confiável para evitar que um adversário convença o IMD de que o dispositivo externo está ausente. Uma vez que o IMD está dentro do corpo do paciente e o dispositivo externo é colocado fora do corpo, temos que contar com a comunicação sem fio para transmitir informações. Isso abre a possibilidade de o adversário bloquear o canal para criar a aparência de que o dispositivo externo está ausente.

Neste artigo, propomos o IMDGuard, um esquema de segurança para dispositivos cardíacos implantáveis [1], que são implantados para monitorar ou tratar *cardíaco* condições médicas. Esses IMDs são um grupo amplamente utilizado de dispositivos médicos e exemplos incluem cardioversor-desfibrilador implantável, marca-passo e sensor de ECG (eletrocardiograma). O IMDGuard aproveita o Guardian, um dispositivo vestível externo, para coordenar as interações entre o IMD e o médico de forma a fornecer segurança em condições regulares e permitir o acesso com segurança em caso de emergência. Os sinais de ECG do paciente são explorados para extrair chaves compartilhadas exclusivamente entre o IMD e o Guardian. Este esquema de extração de chave não precisa de nenhum segredo pré-distribuído para que seja fácil recodificar o IMD quando o Guardian for perdido ou apresentar mau funcionamento. Além disso, torna o adversário incapaz de forjar Guardiões falsos, exceto por meio de contatos físicos com o paciente. O IMDGuard também pode prevenir com eficácia que o adversário capaz de bloquear falsifique o IMD que o Guardian está ausente. O engano do adversário será revelado por colaborações entre o IMD e o Guardian por meio do mecanismo de notificação baseado no bloqueio defensivo.

* Departamento de Ciência da Computação, The College of William and Mary, Email: {fxu, zhengrui, liqun}@cs.wm.edu.
† Departamento de Ciências da Computação e da Informação, Temple University, Email: cctan@temple.edu. Este trabalho foi feito quando o autor estava com o College of William and Mary.
‡ Escola de Computação, Universidade Nacional de Tecnologia de Defesa, República Popular da China. E-mail: bswang@nudt.edu.cn. Este trabalho foi feito enquanto o autor estava visitando o College of William and Mary.

[1] Referimo-nos aos IMDs como dispositivos cardíacos implantáveis no resto do papel

Nosso esquema de segurança do IMD faz as seguintes contribuições:

1) Trabalhos anteriores em acordos de chave baseados em ECG não extraíram corretamente a aleatoriedade dos dados de entrada ou avaliaram corretamente os resultados finais. Em contraste, somos os primeiros a propor um esquema de extração segura rigorosamente teórico da informação e avaliar seu desempenho em sistemas embarcados com recursos limitados.

2) Até onde sabemos, somos os primeiros a fi nalizar e implementar um protocolo seguro abrangente para a arquitetura proposta anteriormente que usa dispositivos externos como proxy de autenticação para proteger o IMD. Além disso, nosso design é feito sob medida para os IMDs e não requer nenhum hardware especial. Por exemplo, o esquema de extração de chave do IMDGuard é proposto com base na funcionalidade existente do IMD, e o dispositivo vestível não precisa de módulos transmissores poderosos para se defender contra os ataques de spoofing do adversário.

3) Realizamos experimentos extensivos em nosso protótipo para avaliar a validade e o desempenho do IMDGuard. O restante do artigo é assim. Revisamos o trabalho relacionado na Seção II e o histórico e a formulação do problema na Seção III. As Seções IV e V detalham o esquema do IMDGuard, incluindo protocolos de tempo de execução e inicialização de chave entre o IMD e o Guardian, e a Seção VI descreve nossa implementação de protótipo. Fornecemos avaliação sobre nosso esquema na Seção VII e concluímos na Seção VIII.

II. R TRABALHO ELATED

O aumento do uso de IMDs tem motivado pesquisadores a estudar as questões de segurança em tais dispositivos [6] - [8]. A solução proposta, embora segura, não aborda o que acontece em uma situação de emergência em que os médicos não conseguem obter as chaves necessárias do paciente.

O trabalho posterior de [5] explorou o conceito de segurança e propôs a ideia de *falha de abertura,* uma propriedade para contornar fisicamente a proteção de segurança do IMD em uma emergência, por meio do uso de um dispositivo externo. Isso introduz uma nova ameaça à segurança por meio da qual um adversário pode tentar induzir o estado de falha aberta para acessar o IMD. Nosso protocolo proposto também fornece a propriedade fail-open, mas difere de [5] em três aspectos. Em primeiro lugar, nosso design evita a transmissão periódica de mensagens, que consome energia considerável da bateria e expõe os pacientes a riscos de privacidade. Em segundo lugar, nossa solução protege o IMD sem qualquer suposição sobre a capacidade de transmissão do adversário. Terceiro, nosso esquema é abrangente e avaliado em sistemas embarcados com restrição de recursos.

Nossa solução inclui um mecanismo resistente a ataques de spoofing relacionado ao bloqueio. O congestionamento em redes de sensores foi estudado por [9], [13], [19]. No entanto, esses protocolos de interferência não consideram os recursos do IMD e não podem ser usados diretamente em nosso problema. Outras estratégias anti-jamming como [16] e modulação Direct-Sequence Spread Spectrum também não podem ser aplicadas por causa da limitação do hardware e da regulação da banda [2].

Nossa solução também inclui um algoritmo de extração de chave de Sinais de ECG para proteger o link entre o IMD e o 1863 uardiano. A ideia de usar sinais fisiológicos para proteger

G
comunicações inter-sensor foram introduzidas pela primeira vez em [4], e Poon *et al. [* 14] colocam este esquema em prática para sinais de ECG e PPG (fotopletismograma). Intervalos entre pulsos (IPIs) de batimentos cardíacos são explorados para extrair chaves em [3]. Para 16 IPIs individuais consecutivos, o tempo final em milissegundos (em)

de cada IPI é calculado, definindo 0 como o horário de início do primeiro IPI. Em seguida, os 7º e 8º dígitos das representações binárias dos tempos finais são extraídos para formar a chave. Mesmo que as sequências binárias extraídas possam passar por vários testes de aleatoriedade NIST [15], na verdade elas não são aleatórias como parecem. Como o IPI médio é de cerca de 850 milissegundos, o 7º e o 8º dígitos do tempo final não são aleatórios. A aleatoriedade está nos dígitos mais baixos, assim como o erro. Em comparação com ele, nossa solução explorou uma nova maneira de utilizar corretamente IPIs para extrair aleatoriedade.

Um esquema mais rápido foi proposto por [17] onde os coeficientes de Fast Fourier Transform (FFT) em amostras de sinais de ECG são usados para extrair chaves, no entanto, o artigo também não fornece uma análise rigorosa se as amostras de entrada contêm entropia suficiente para gerar uma chave com bits de entropia necessários e avalia a chave após o hashing , o que não é logicamente correto. Nosso esquema de extração de chave difere deste trabalho em duas facetas. Em primeiro lugar, damos um estudo teórico de informação rigorosa sobre a aleatoriedade da característica fisiológica a partir da qual a chave é

extraído. Em segundo lugar, mostramos que o adversário não pode obter nenhum conhecimento sobre as chaves geradas, exceto que ele pode medir os sinais de ECG simultaneamente sem ser capturado.

III. B ACKGROUND E F ORMULAÇÃO

Nesta seção, mostramos primeiro a configuração do IMDGuard, depois, o modelo do adversário e, finalmente, a abordagem contra o adversário.

A. Configuração do IMDGuard

O IMDGuard tem três componentes, o IMD, o Guardian e programador. O DMI, uma vez implantado, deve permanecer dentro do corpo por um longo período de tempo. O programador, como um controlador externo, fornece aos médicos uma interface para interagir com o IMD por meio de transmissão de radiofrequência para ajustar os parâmetros de funcionamento, alterar os modos de operação ou recuperar dados armazenados. Acima de dois são wireless padrão

instrumentos médicos programáveis. The Guardian é um wearable dispositivo com mais potência e recursos computacionais do que o IMD. Este Guardian funciona como um proxy para o IMD e realiza a autenticação em seu nome. Tanto o IMD quanto o Guardian são capazes de medir sinais de ECG. As interações desses componentes são ilustradas na Fig. 1. Link α

representa o processo de controle de acesso entre o Guardian e o programador. Ligação β representa o processo de emparelhamento inicial entre o IMD e o Guardian. Ligação γ representa a comunicação segura protegida pela chave atribuída pelo Guardian ao IMD e ao programador.

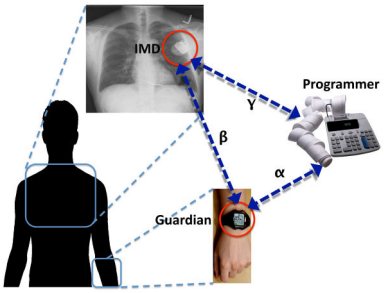

Fig. 1: Interações de comunicação no IMDGuard. A imagem parcial é adotada de [11].

B. Modelo Adversário

Consideramos um adversário cujo objetivo é tentar programar ou recuperar dados do IMD sem ser pego. O adversário é bem-sucedido se conseguir acessar as informações do IMD na presença do Guardião. Ataques de interrupção, como negação de serviço, são excluídos em nosso modelo de adversário. Presumimos que o adversário não pode medir fisicamente os sinais de ECG em tempo real do paciente sem ser detectado. Também assumimos que não há adversário em uma situação de emergência. Isso é razoável, pois em tal cenário, o paciente com o DMI provavelmente estará em uma instalação segura, como uma sala de emergência de um hospital, que pode limitar a presença de adversários.

Classificamos a estratégia de ataque do adversário contra o IMDGuard em dois aspectos. A primeira é quando o adversário tenta se passar pelo Guardian derivando a chave compartilhada entre o IMD e o Guardian de pesquisa de força bruta ou registros médicos históricos do paciente. A segunda é quando o adversário pode falsificar a ausência do Guardian bloqueando seletivamente as mensagens dele, a fim de convencer o IMD a desativar a proteção de segurança e mudar para o status de emergência.

C. Visão geral do IMDGuard

No IMDGuard, o Guardian executa duas funções essenciais. Primeiro, o Guardian é usado para controlar qual modo, *regular* ou *emergência,* o IMD deve entrar. Quando o paciente está usando o Guardian, o IMD deve funcionar no modo normal. Em um modo normal, o programador que precisa interagir com o IMD será primeiro autenticado pelo Guardian, que então emitirá as chaves apropriadas para o IMD e o programador. Quando o IMD não detecta a presença do Guardian, o IMD deve entrar no modo de emergência. A vantagem é que, em caso de emergência, o médico poderá remover fisicamente o Guardian e ter acesso irrestrito ao CPM.

Em segundo lugar, o Guardian autenticará o programador em nome do IMD. Isso conservará a bateria do IMD, reduzindo o número de operações realizadas por ele. Isso também simplifica o projeto geral do IMD, uma vez que o IMD não precisa manter materiais criptográficos, como chaves assimétricas e listas de controle de acesso.

Presumimos que o Guardian sempre será usado pelo paciente. É razoável, pois o Guardião pode assumir a forma
de um relógio e o paciente pode usá-lo o tempo todo. W 1 e 864
também presuma que o adversário não pode remover fisicamente o Guardião sem que o paciente perceba.

Não assumimos que o IMD deva se associar exclusivamente a um Guardião. Portanto, antes de tornar o Guardian eficaz, ele precisa ser inicializado compartilhando uma chave secreta entre o IMD e este Guardian, para que se reconheçam. Além disso, no caso extremo de a corrente do Guardian usada ser quebrada ou perdida, um novo Guardian pode ser emparelhado com o IMD facilmente, sem a necessidade de recuperar a chave antiga ou redefinir o IMD.

Para realizar essa funcionalidade, um recurso importante do IMDGuard é um esquema de estabelecimento de chave seguro baseado em sinais de ECG. Tanto o IMD quanto o Guardian coletam localmente a mesma fonte aleatória simultaneamente, o ECG do paciente, e extraem uma chave simétrica dos recursos de ECG após o delineamento do ECG. Ao contrário da troca de chaves Dif fi e-Hellman ou extrações de chaves baseadas em wireless, este esquema é robusto contra ataques man-in-the-middle, desde que o adversário não possa medir fisicamente os sinais de ECG do paciente em tempo real.

A outra característica principal do IMDGuard é o mecanismo resistente a ataques de spoofing. Se o adversário tentar persuadir o IMD a entrar no modo de emergência bloqueando todas as mensagens transmitidas pelo Guardião, o Guardião ainda pode anunciar sua presença ao IMD por *obstruindo a transmissão da mensagem de desafio pelo IMD.* A intuição é que o Guardian dificilmente pode bloquear a transmissão do adversário para o IMD, uma vez que não tem conhecimento sobre o hardware e as capacidades do adversário. Em vez disso, o Guardian pode ser calibrado de acordo com os parâmetros de seu próprio IMD e pode sempre obstruir com sucesso as transmissões de seu IMD.

IV. P ROTOCOL D ENTRAR IMDG UARD

Aqui, apresentamos os protocolos do IMDGuard. Nós presumimos que o IMD e o Guardian já fizeram o emparelhamento com uma chave secreta compartilhada após a fase de estabelecimento da chave, que é descrita na seção a seguir. Presumimos que o Guardian tem uma lista de programadores legítimos e suas chaves públicas correspondentes. Essas informações podem ser instaladas com segurança durante a internação. A Tabela I resume as notações usadas.

TABELA I: Tabela de notações

G	o guardião
P	o programador
ni_j	a Eu o nonce gerado por j, j ∈ { IMD, G, P}
H (·)	função hash criptográfica padrão, por exemplo, SHA-1
SK	a chave secreta compartilhada entre o IMD e o Guardian usando extração de chave baseada em ECG (Seção V)
$PK+_j$	a chave pública de j, j ∈ { G, P}
$PK-_j$	a chave privada de j, j ∈ { G, P}
TK	a chave simétrica temporária usada para uma sessão de
EU IRIA	identificação do IMD

A. Protocolo IMD básico

O IMD será ativado periodicamente para determinar se há alguma solicitação do programador. Depois que o IMD recebe uma solicitação do programador, o IMD executará o Algoritmo 1. O IMD enviará de volta sua ID, e uma

nonce n 1 IMD, que é usado como a identidade da sessão para resistir contra o ataque de repetição. Então, o IMD inicia um cronômetro T 1 esperar pelo Guardian, se presente, notificá-lo para executar o *regular protocolo de condição.* No caso de não haver mensagem do Guardião quando T 1 tempo limite, o IMD executará o *emergência* 1865 Quando o IMD entrar no modo de emergência, ele executará *protocolo de condição.*

Algoritmo 1 Algoritmo de IMD básico

1: Envie de volta para o P, ID e n 1 IMD
2: Comece o cronômetro de espera T 1
(Guardian, se presente, executará o protocolo de autenticação (Fig. 2) durante T 1)
3: **enquanto** T 1 tempo limite == *FALSO* **Faz**
4: **E se** receber mensagem válida de G **então**
5: Condição regular, não é uma emergência. 6:
Execute o algoritmo de condição regular (Seção IV-C) 7: Possível condição de emergência.
Execute o algoritmo de condição de emergência (Seção IV-D)

B. Programador de autenticação do guardião

Na Etapa 2 do Algoritmo 1, o Guardian irá autenticar o programador dentro do período de tempo T 1 O protocolo de autenticação é mostrado na Fig. 2.

Um nonce aleatório n 1 G é assinado pelo programador e enviado de volta ao Guardian junto com outro nonce aleatório n 1 P. A assinatura do programador é veri fi cada pelo Guardian. Se não for válido, o Guardião informará o IMD para negar a solicitação de sessão por meio da mensagem { Não 1 IMD} SK (também irá informar o programador que a autenticação falhou e a sessão foi negada). Se válido, o Guardian atribuirá uma chave de sessão temporária TK ao IMD e ao programador para uma comunicação segura. Deixamos o IMD usar chaves simétricas ao se comunicar com o programador e o Guardian para reduzir a carga computacional no IMD. O Guardian e o programador usam chaves públicas para se autenticarem para um melhor gerenciamento de chaves.

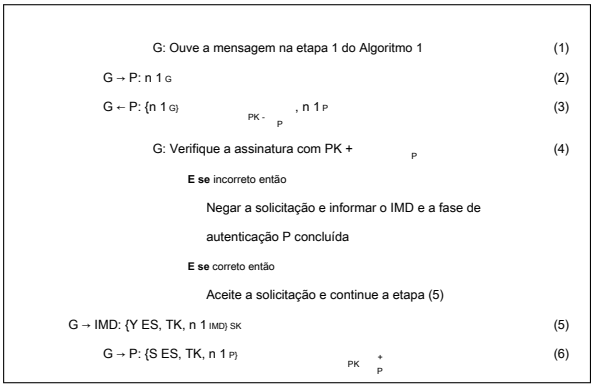

Fig. 2: Programador de autenticação Guardian

C. Protocolo de Condição Regular

Quando o IMD entrar na condição normal (Etapa 6 do Algoritmo 1), ele executará o protocolo mostrado na Fig. 3. Após descriptografar a mensagem R, o IMD negará o acesso se ele recebe uma mensagem NÃO. Caso contrário, uma mensagem SIM indica este cenário, a função do Guardião, o bloqueio defensivo

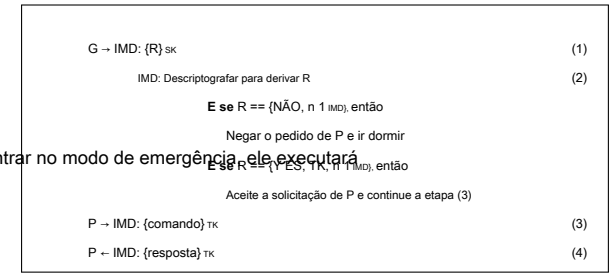

Fig. 3: Protocolo de condição regular do IMD

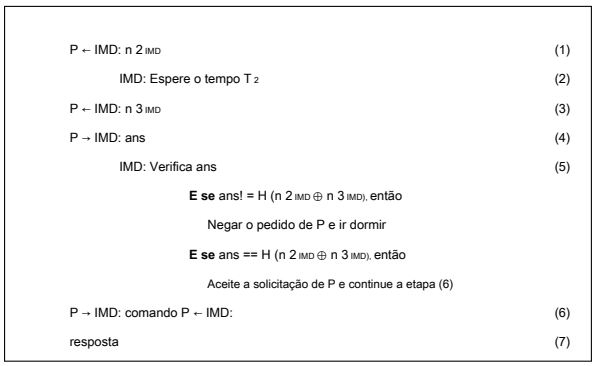

Fig. 4: Protocolo de condição de emergência do IMD

que o programador foi autenticado, e a chave de sessão para as etapas 3 e 4 é TK.

D. Protocolo de Condição de Emergência

o protocolo na Fig. 4, e envie a primeira parte do desafio, um nonce aleatório n 2 IMD. Depois de esperar T 2 Tempo, o IMD envia a segunda parte do desafio, n 3 IMD. Supondo que o Guardião foi removido fisicamente, o programador irá transmitir de volta a resposta correta ao desafio, H (n 2 IMD ⊕ n 3 IMD). Se o Guardian estiver presente, o programador não poderá retornar a resposta correta. Nós explique como o Guardian interrompe isso na próxima subseção.

E. Protocolo de resistência a ataques spoofing

Aqui, mostramos como nosso protocolo é resiliente ao ataque de spoofing do adversário baseado em jamming. O adversário pode tentar bloquear as comunicações entre o IMD e o Guardião para induzir o IMD a entrar no modo de emergência na Etapa 7 do Algoritmo 1. Em outras palavras, o adversário irá bloquear o canal por período de tempo T 1 Uma vez que o IMD não recebe nenhuma resposta do Guardião, o IMD irá prossiga para executar o algoritmo da condição de emergência. Para

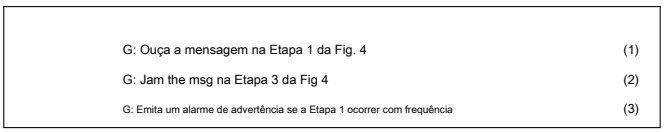

Fig. 5: O mecanismo de notificação do Guardian

descrito na Fig. 5, é acionado para bloquear esta sessão. Quando o Guardião ouve a primeira parte da mensagem de desafio (Fig. 4 Passo 1) enviado pelo IMD ao programador, o Guardião irá perceber que o link de comunicação entre o próprio IMD e ele está bloqueado por um adversário com alta probabilidade. Em seguida, o Guardian bloqueia a segunda parte da mensagem de desafio do IMD (Fig. 4 Etapa 3). Essa operação é viável na prática com base na sincronização solta do Guardian por meio da mensagem na Etapa 1 da Fig. 4. Em outras palavras, o Guardian saberá que há outra mensagem, que é o alvo de interferência, que vai enviar T tempo depois. Essas informações podem ajudar o Guardian a bloquear a mensagem alvo com menos esforço.

Existem duas vantagens em permitir que o Guardian bloqueie a mensagem do IMD em vez da mensagem do adversário. Em primeiro lugar, o hardware do adversário pode ser muito mais poderoso do que o Guardian, tornando difícil calibrar a força de broadcast do Guardian necessária para bloquear o sinal do adversário com sucesso. Em segundo lugar, o The Guardian pode cronometrar exatamente quando deve estar bloqueando, uma vez que sabe quando o IMD começará a transmissão. Isso conserva o poder do Guardian, reduzindo o período de interferência.

V. Key Estabelecimento em IMDGuard

Na seção anterior, presumimos que haja uma chave secreta já compartilhada entre o IMD e o Guardian para proteger sua comunicação. No entanto, esse estabelecimento-chave é desafiador se o IMD e o Guardião não compartilharem nenhum segredo de antemão. Nesta seção, apresentamos um esquema de extração de chave segura com base no delineamento de ECG para estabelecer uma chave secreta simétrica que une o IMD e o Guardian, tornando o adversário impossível de falsificar o Guardian.

A. Delineamento ECG

Conduzimos o delineamento do ECG com os algoritmos baseados em wavelet mencionados em [10], [12]. A Fig. 6 mostra um resultado de exemplo de nosso delineamento baseado em transformada wavelet. Usando as informações de cruzamentos locais máximos, mínimos e zero em diferentes escalas na transformada wavelet, o algoritmo é capaz de detectar todos os pontos significativos de ECG em um ciclo de batimento cardíaco, primeiro o pico R, depois pico Q e pico S, seguido pela onda T e pela onda P.

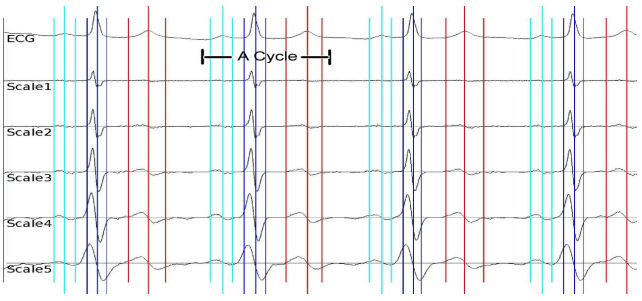

Fig. 6: Transformada wavelet das ondas de ECG nas primeiras cinco escalas. O primeiro painel é o sinal de ECG, os outros cinco, de cima para baixo, são as transformações wavelet correspondentes da escala 2^1 escalar 2^5

Conforme mostrado na Fig. 6, em cada ciclo de batimento cardíaco, as três linhas azuis no meio denotam o início, pico R e deslocamento de QRS complexo respectivamente. As três linhas ciano à esquerda denotam o início, o pico P e o deslocamento da onda P, respectivamente. As três linhas vermelhas à direita denotam o início, o pico T e o deslocamento da onda T, respectivamente.

1200 linhas de código. A alta precisão é alcançada para reduzir a taxa de incompatibilidade de IPIs, tornando a extração de chaves a seguir e fi ciente.

B. Um recurso de ECG para extração de chave

Dadas as duas medições de ECG feitas em diferentes partes do corpo humano, queremos extrair delas uma chave simétrica após o delineamento. Como um requisito fundamental, a chave deve ser *aleatória*. Assim, a chave deve ser extraída de um recurso de ECG de forma que: (1) o próprio recurso seja aleatório; e (2) o recurso tem lugares comuns para o IMD e o Guardian.

Após o delineamento do ECG, temos as informações de tempo de todas as características significativas do ECG, a saber, onda P, complexo QRS e onda T, para cada ciclo de batimento cardíaco. Como os sinais de ECG são periódicos, para garantir a aleatoriedade, não podemos usar diretamente todos os pontos de delineamento ao mesmo tempo. Depois que um recurso é escolhido, outros recursos no mesmo ciclo de batimento cardíaco não são mais totalmente aleatórios. Por exemplo, se sabemos a posição do pico R, podemos facilmente adivinhar em quais faixas o pico Q, pico S, pico T e pico P no mesmo ciclo caem. Mesmo para recursos em ciclos diferentes, as posições dos recursos não são totalmente aleatórias. Por exemplo, dada a posição do pico R em um ciclo de batimento cardíaco, o pico do R seguinte cairá em uma pequena faixa porque o intervalo interpulso comum (IPI) é conhecido. (Para adultos, o IPI comum é de cerca de 850 em)

Usaremos as informações dos picos R, que são mais salientes, para extrair as chaves. Dada uma sequência consecutiva de picos R, os IPIs são obtidos calculando a diferença de tempo dos dois picos R consecutivos. Suponha R_{Eu} denotando o tempo do Eu o pico R, então $IPI_{i} = R_{i+1} - R_{Eu.}$ Como o valor médio do IPI é bastante conhecido, temos que excluir a média valor ao extrair a chave. Primeiro, estimamos empiricamente quantos bits aleatórios podem ser extraídos de cada valor de IPI. Convertemos os valores IPI em representações binárias e, em seguida, examinamos a aleatoriedade de cada dígito das representações binárias. É claro que os bits aleatórios estão nos dígitos baixos. Para o Eu º dígito, contamos o número de amostras para os seguintes casos:

(1) $n_{00:}$ se o Eu º dígito da amostra j é 0, assim como o da amostra j + 1, então incremente n_{00} por 1;

(2) $n_{01:}$ se o Eu º dígito da amostra j é 0, e o da amostra j + 1 é 1, então incremente n_{01} por 1;

(3) $n_{10:}$ se o Eu º dígito da amostra j é 1, e o da amostra j + 1 é 0, então incrementa n_{10} por 1;

(4) $n_{11:}$ se o Eu º dígito da amostra j é 1, e também o de amostra j + 1, então incremente n_{11} por 1; Onde $1 \le j < n$, n é o número total de IPI consecutivos amostras.

Em seguida, calculamos as quatro possibilidades $P_{lk} = n_{lk} / (n - 1)$, Onde $lk \in \{00, 01, 10, 11\}$. Se o Eu o dígito é aleatório e independentes, todas as quatro possibilidades devem ficar em torno de 25%.

TABELA II: A qualidade da aleatoriedade de cada dígito.

Eu	P_{00} (%)	P_{01}	P_{10}	P_{11}
1	29,2	24,4	24,4	21,9
2	28,9	24,3	24,4	22,4
3	25,2	24,6	24,7	25,5
4	27,9	25,6	25,6	20,9
5	57,5	18,8	18,8	4,9
6	2,5	13,2	13,2	71,1
7	99,1	0,4	0,4	0,0
8	99,7	0,1	0,1	0,0

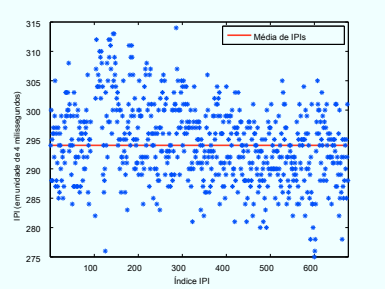

Fig. 7: A flutuação dos IPIs contra a média, que é 294 por unidade de 4 em (Taxa de amostragem de 250 Hz).

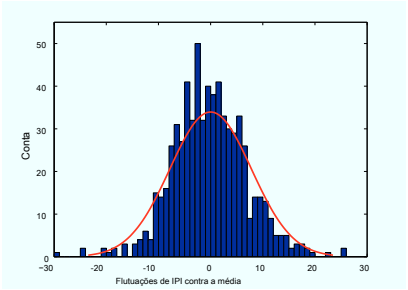

Fig. 8: A distribuição normal ajustando-se à flutuação de IPIs contra a média, com μ = 0 e
$\sigma^2 = 61$

Listamos as possibilidades para os 8 dígitos mais baixos das amostras de IPI na Tabela II. Estabelecemos um limite de 5%. Conforme mostrado na tabela, os últimos 4 dígitos são aleatórios, enquanto o 5º dígito não é. Para o
5º dígito, P_{00} é mais de 50% enquanto P_{11} é inferior a 5%. Também calculamos a entropia das flutuações diretamente do
dados originais, que são cerca de 5. Portanto, podemos extrair com segurança 4 bits de cada IPI.

TABELA III: Distribuição normal dividida em 16 seções iguais.

	Domínio		Domínio
1	$(-\infty, \mu - 1.534\ \sigma)$	9	$(\mu, \mu + 0{,}157\ \sigma)$
2	$(\mu - 1.534\ \sigma, \mu - 1.151\ \sigma)$	10	$(\mu + 0{,}157\ \sigma, \mu + 0{,}319\ \sigma)$
3	$(\mu - 1.151\ \sigma, \mu - 0{,}887\ \sigma)$	11	$(\mu + 0{,}319\ \sigma, \mu + 0{,}489\ \sigma)$
4	$(\mu - 0{,}887\ \sigma, \mu - 0{,}675\ \sigma)$	12	$(\mu + 0{,}489\ \sigma, \mu + 0{,}675\ \sigma)$
5	$(\mu - 0{,}675\ \sigma, \mu - 0{,}489\ \sigma)$	13	$(\mu + 0{,}675\ \sigma, \mu + 0{,}887\ \sigma)$
6	$(\mu - 0{,}489\ \sigma, \mu - 0{,}319\ \sigma)$	14	$(\mu + 0{,}887\ \sigma, \mu + 1.151\ \sigma)$
7	$(\mu - 0{,}319\ \sigma, \mu - 0{,}157\ \sigma)$	15	$(\mu + 1.151\ \sigma, \mu + 1.534\ \sigma)$
8	$(\mu - 0{,}157\ \sigma, \mu)$	16	$(\mu + 1.534\ \sigma, +\infty)$

C. Quantização

Esta subseção mostrará como extrair 4 bits aleatórios de cada amostra de IPI. Não podemos usar os últimos 4 dígitos das representações binárias do IPI diretamente, porque a ligeira diferença entre os dados em ambos os lados pode causar grandes diferenças nos últimos 4 dígitos das representações binárias, levando a taxa de incompatibilidade a até 20%. Na verdade, os 4 dígitos mais baixos são as flutuações dos IPIs. A Fig. 7 mostra as flutuações do IPI em relação ao valor médio. Em termos de entropia, as flutuações não perdem
qualquer entropia de amostras de IPI. Embora o IPI médio seja interrompido 1 e 867 Uma vez que o IMD está medindo os sinais de ECG o tempo todo,
estável, a diferença entre o IPI individual e o valor médio é imprevisível. Pode ser positivo, negativo ou zero. E o valor da flutuação é bastante aleatório em um certo intervalo. Portanto, as flutuações podem ser escolhidas como a base para extrair a chave.

Na perspectiva das estatísticas, as flutuações se configuram em uma distribuição normal. A Fig. 8 mostra o histograma das amostras na Fig. 7, com um ajuste de distribuição normal. A distribuição normal ajustada também resulta em uma entropia $_1$
$_{2}$registro$(2\pi e\sigma^2) \sim$ = 5, qual
está dentro da faixa resultante dos dados originais. E provamos que os últimos 4 dígitos são aleatórios, o que implica que a entropia real é pelo menos 4. Nesse sentido, a distribuição das flutuações é de fato uma distribuição normal, ou pelo menos próxima de.

O IMDGuard fornece o seguinte algoritmo para fazer a quantização. Este algoritmo é baseado na suposição de que as flutuações formam uma distribuição normal. Para uma distribuição normal
N (μ, σ), dado μ e σ, podemos dividir a função de densidade de probabilidade em 16 seções consecutivas de modo que, em cada seção, a densidade de possibilidade cumulativa seja 1/16. O domínio de cada seção pode ser denotado por uma função de σ e μ, conforme mostrado na Tabela III. E se μ ou qualquer ponto inicial / final de qualquer domínio é um número inteiro, dividimos as amostras com esse valor em duas partes, cada uma indo para um dos domínios próximos. A divisão
depende do índice da amostra. As amostras com índice ímpar formam uma parte e aquelas com índice par formam a outra. Observe que
σ é grande o suficiente para que cada domínio contenha pelo menos um inteiro, uma vez que a entropia indica que σ não é pequeno. O objetivo da divisão é aproximadamente, mas não precisamente, igualar o número de amostras em cada domínio, tornando a quantização imprevisível. Os 16 domínios são mapeados um para um para os códigos cinza de 4 bits.

Uma vez que o IMD está medindo os sinais de ECG o tempo todo, é capaz de calcular σ e μ por um longo período, digamos 15 minutos, e guarde-o. Durante a geração da chave, o IMD pode enviar esses parâmetros ao Guardian. Este processo não vaza nenhuma informação sobre a chave, pois o adversário ainda não sabe qual amostra está em qual domínio e quantas amostras estão em cada domínio. A quantização é mostrada no Algoritmo 2.

Algoritmo 2 Algoritmo de Quantização

Entrada: n IPIs consecutivos de ECG, $IPI_1, IPI_2, ..., IPI_n$.
Resultado: 4 n bit string binária.

```
1   Obtenha parâmetros μ e σ
2   Calcule 16 domínios com base na Tabela III: D1, D2, ..., D16
3   Numere os códigos cinza de 4 bits: G1, G2, ..., G16
4   Saída = φ
5   para Eu ← 1 para n
6       para j ← 1 para 16
7           E se IPIEu cai em Dj
8               Saída = Concatenar (Saída, Gj)
```

D. Reconciliação

Devido à alta precisão do delineamento de ECG, as duas sequências binárias quantizadas respectivamente pelo IMD e pelo Guardian

têm uma baixa taxa de incompatibilidade. Para dois blocos de 4 bits correspondentes ao mesmo ciclo de batimento cardíaco em ambos os lados, um bit é diferente na maioria dos casos se houver uma incompatibilidade. Em alguns casos, existem dois bits diferentes. Não há caso em que 3 ou 4 bits sejam diferentes. Com base nessas observações, projetamos um algoritmo de reconciliação de 2 rodadas. Ele realiza a Rodada 2 somente se a Rodada 1 falhar.

Rodada 1: Para cada IPI, o IMD e o Guardian obtêm um bloco de 4 bits. Ambos os lados calculam a paridade de seu próprio bloco e trocam essas informações. Se as paridades forem diferentes, o bloco é descartado. Caso contrário, cada lado extrairá os primeiros 3 bits do bloco; o quarto bit é descartado porque a paridade vaza informações de um bit. Esse processo continua até que ambos os lados obtenham 129 bits. O IMD então faz o hash com a função de hash SHA-1 e envia o valor de hash para o Guardian. O Guardian compara esse valor de hash com o seu próprio e notifica o IMD. Se os dois valores de hash corresponderem, o algoritmo é encerrado. Caso contrário, a Rodada 2 será realizada.

2 ª rodada: Para os 43 IPIs escolhidos na Rodada 1, o IMD e o Guardian calculam a paridade dos últimos 2 bits de cada bloco de 4 bits e trocam essas informações. Novamente, os blocos cujas paridades não coincidem são descartados. Para os blocos restantes, ambos os lados extraem o 2º e o 3º bits; o primeiro bit é descartado, pois a segunda paridade também vaza informações de um bit. Obviamente, o comprimento da chave é inferior a 128. Em seguida, ambos os lados continuam a analisar os seguintes IPIs. Nesse momento, eles verificam duas paridades ao mesmo tempo e extraem 2 bits de cada bloco que passa na verificação de paridade. O processo continua até que ambos os lados obtenham 128 bits.

VI. Protótipo e Implementação

Um desafio que envolve experimentos de IMD é a dificuldade em obter códigos-fonte e plataformas abertas de fornecedores comerciais. Em nosso sistema de protótipo, escolhemos o TelosB com TinyOS 2.1, uma plataforma de pesquisa aberta do sistema embarcado com restrição de recursos como um substituto do IMD. Os detalhes relacionados são descritos abaixo.

Comparação do transmissor: O TelosB utiliza o transmissor CC2420 para comunicação sem fio. O CC2420 é comparável ao rádio típico do serviço de comunicação de implantes médicos (MICS), como o ZL70101 [1], usado em IMDs. Ambos são dispositivos de rádio de baixa potência com quantidade semelhante de consumo de energia durante a transmissão. O ZL70101 gasta 5 mA, enquanto 8,5 mA é alcançável para o CC2420. Além disso, ambos compartilham outros recursos comuns, como suporte a vários canais e ciclo de trabalho. A diferença entre CC2420 e ZL70101 é que o rádio MICS opera banda de frequência mais baixa entre 402-405 MHz devido às características razoáveis de propagação do sinal no corpo humano. Isso não tem impacto em nossa avaliação, pois nossa implementação não depende da frequência ou do número de canais disponíveis.

Tamanho do código: O tamanho do código de cada componente após a compilação é mostrado na Tabela IV. ECC [18] é a criptografia de curva elíptica que desenvolvemos para fornecer o esquema de chave pública entre o programador e o Guardian. Para referência, um IMD típico produzido em 2002 é capaz de conter 2 MB de memória [5].

TABELA IV: Tamanho do código de nossa implementação de protótipo

Módulo	ROM(*bytes)*	RAM (*bytes)*
IMD	20656	1056
Programador	20754	1060
Guardião	20614	1050
ECC	42190	1931
Extração de chave	10078	887
Delineamento ECG	18720	9652

VII. Avaliação de IMDGuard

A avaliação de desempenho do IMDGuard é dividida em três porções. Primeiro, avaliamos de forma abrangente a qualidade da chave extraída dos sinais de ECG. Em seguida, conduzimos uma série de experimentos sobre a eficácia de defender os ataques de spoofing do adversário. Finalmente, apresentamos a eficiência de cada um dos componentes críticos em nossa implementação.

A. Estabelecimento-chave

Nesta seção, avaliaremos a chave gerada de acordo com aos algoritmos na Seção V. Os sinais de ECG são do banco de dados PhysioBank (http://www.physionet.org/physiobank). Abordaremos três características importantes da chave: (1) variância temporal; (2) eficiência; (3) aleatoriedade.

1) Variância Temporal: Dada uma chave de 128 bits gerada pelo IMD e Guardian, queremos saber se o adversário pode obter alguma ajuda se puder acessar registros históricos / futuros dos sinais de ECG do paciente. A métrica usada é o hamming distância entre a chave e qualquer outra string aleatória de 128 bits antes ou depois dela. A distância de Hamming entre duas sequências binárias de igual comprimento é o número de posições em quais os símbolos correspondentes são diferentes. Dadas duas strings aleatórias, se forem independentes, a possibilidade de comer hamming dis (tan) ce k segue uma distribuição binomial, que é: $P(k) = {}_{n}{}_{k} p^{k} (1 - p)^{n-k}$, Onde n = 128 e p = 1/2. E o valor médio de k, também conhecido como valor esperado, é E (k) = np = 1/2 × 128 = 64.

Examinamos essas distâncias de hamming. Não há sinal valor igual a 0 ou 128, e todos os pontos estão entre 45 e 85. Quanto mais próximo da distância média de hamming, que é 64, mais densos os pontos. Plotamos a possibilidade das distâncias de hamming, como mostrado na Fig. 9. Como podemos ver, os dados medidos combinam muito bem com a distribuição binomial teórica com n = 128 e p = 1/2. Do prospecto estatístico, a chave de 128 bits gerada por nosso esquema não se relaciona ao ECG histórico ou aos sinais de ECG futuros. Assim, mesmo o adversário obtém dados históricos ou futuros do ECG do paciente, ele não pode obter nenhuma ajuda deles. Isso também indica a aleatoriedade da chave de 128 bits.

Também realizamos a mesma avaliação entre as chaves geradas a partir de sinais de ECG de pessoas diferentes e obtemos o mesmo resultado conforme o esperado. O registro histórico / futuro de uma mesma pessoa não ajuda o adversário, nem o das outras pessoas. [17] fizeram avaliação semelhante sobre seu esquema. No entanto, eles fizeram isso depois de fazer o hash de uma string idêntica entre duas partes com uma função hash unilateral. Embora tenham obtido resultados semelhantes, seus

os resultados não puderam provar o que eles alegaram. Hashing fará 1 e uma string aleatória, não importa se a string original é aleatória ou não.

2) Eficiência: Na fase de reconciliação, existem duas rodadas. Na primeira rodada, 3 bits aleatórios são extraídos de cada IPI. Portanto, ele precisa de 43 IPI para obter uma chave de 128 bits. Se a primeira rodada falhar, o algoritmo fará a segunda, extraindo 2 bits de cada IPI. Nesse caso, são necessários mais 21 IPIs, além dos 43 IPIs da primeira rodada. Em 88% dos casos, a primeira rodada é bem-sucedida. A segunda rodada é sempre bem-sucedida, pelo menos não encontramos uma única falha em todos os nossos rastros. Assim, em média, ele precisa

45,5 IPIs, sem contar o IPI descartado. Durante a Rodada 1, cerca de 25% das amostras são descartadas, e durante a Rodada 2, apenas

0,3% das amostras são descartadas. Levando em consideração as amostras descartadas, são necessários 61 IPIs, correspondentes a aproximadamente 45 segundos, para gerar uma chave com sucesso.

3) Aleatoriedade: Para avaliar a aleatoriedade do fluxo de bits gerado empregado como chaves secretas, executamos os testes de aleatoriedade no conjunto de testes do NIST [15]. Existem no total 15 testes estatísticos diferentes, e executamos 9 deles. Os outros 6 requerem um fluxo de bits muito longo que não podemos gerar do banco de dados do PhysioBank. Nosso fluxo de bits passa em todos os 9 testes, mostrando uma boa qualidade de aleatoriedade.

As avaliações mostram que mesmo os registros de ECG do paciente não podem ajudar o adversário a prever a chave compartilhada entre o IMD e o Guardião, a menos que ele meça fisicamente os sinais de ECG do paciente simultaneamente durante o estabelecimento da chave. No entanto, é impossível para o adversário medir fisicamente o ECG do paciente sem que o paciente esteja ciente disso. Este estabelecimento-chave é robusto para ataques man-in-the-middle. Se uma chave simétrica for estabelecida com sucesso, o Guardian deve ser legítimo.

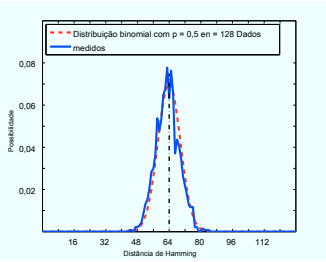

Fig. 9: As distâncias entre o fluxo de 128 bits do sinal de ECG atual e os dos registros históricos se encaixam em uma distribuição binomial com n = 128 e

p = 1/2.

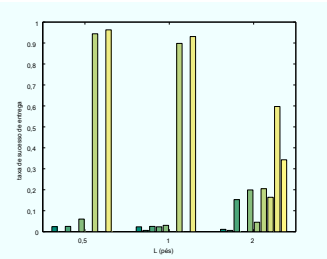

Fig. 10: Resultados para a eficácia de O Guardian está bloqueando quando almeja um programador malicioso. Eixo X representa o distância entre o IMD e o Guardião, e o eixo Y é a taxa de sucesso das mensagens entregues. Para cada distância, dez tentativas foram conduzido.

B. Experiências Relacionadas a Jamming

Existem dois experimentos relacionados a interferência. Primeiro, validamos experimentalmente nossa decisão de bloquear a comunicação do IMD em vez das mensagens do adversário. Em segundo lugar, examinamos nosso método de bloqueio defensivo usando diferentes configurações em termos de nível de força e distância.

Deixamos três ciscos TelosB agirem como o IMD, o Guardião e o programador malicioso (adversário). Para se concentrar em 869 o desempenho de interferência do Guardian, esses ciscos são instalado com o IMDGuard simpli fi cado que será descrito em cada experimento abaixo, bem como a detecção de portadora e o backoff aleatório em motes são desabilitados. Todos os experimentos são realizados em uma grande mesa de escritório em um ambiente interno.

Experimento 1: Variamos a distância entre o IMD e o Guardian de 0,5 a 2 pés e configuramos o programador malicioso a 11 pés de distância do ponto central desses dois dispositivos. A potência de transmissão do Guardian está configurada para ser -15 dBm. Isso é 10 dB maior do que a potência do IMD, mas é 10 dB menor do que o poder do programador malicioso. O intervalo de transmissão, ou seja, o tempo de chegada entre quaisquer duas mensagens, é 20 em. Em seguida, deixamos o adversário enviar mensagens ao IMD, enquanto o Guardian está bloqueando. Após o término da transmissão, determinamos a proporção de mensagens recebidas com sucesso pelo IMD. Os resultados são mostrados na Fig. 10. Como podemos ver, as mensagens para o IMD conseguem escapar do bloqueio com uma probabilidade incerta, baixa em alguns casos, mas alta em outros. Esta observação indica que bloquear a transmissão do adversário não funciona na prática, uma vez que nossas configurações de programador malicioso, como a força de potência relativa (10 dB) e a localização (11 pés), ou mesmo condições mais rigorosas, podem ser alcançadas por um adversário.

Em seguida, repetimos o experimento novamente, desta vez deixando o Guardian bloquear a transmissão do IMD. The Guardian é capaz de *obstruir com sucesso todas as mensagens*. Essa abordagem é mais confiável e eficaz do que bloquear o adversário. Omitimos a figura para os resultados. O sucesso do bloqueio defensivo se deve ao fato de que o Guardian está ciente de todas as configurações do IMD, e que o Guardian é mais poderoso do que o IMD por design.

Experimento 2: Neste experimento, o Guardian bloqueia a mensagem que o IMD envia ao programador malicioso da mesma forma que o experimento acima, mas sob várias configurações. A distância entre o IMD e o programador malicioso é fixada em 1 pé, que é considerada a posição mais próxima que o malicioso poderia ter sem ser detectado pelo paciente. O Guardian é colocado afastado, de 1 a 7 pés, do ponto central do IMD e do programador em cada nível de potência diferente. As taxas de entrega bem-sucedida de todas as mensagens transmitidas em todas as condições são registradas na Fig. 11. É evidente que, desde que o Guardian esteja perto o suficiente, por exemplo, dentro de 2 pés do IMD, a transmissão do IMD para o programador malicioso é totalmente bloqueado, mesmo no caso de teste extremo em que o poder de interferência do Guardian é *20 dB menos do que o do IMD*.

Esta observação é importante porque o IMD geralmente fica nesta faixa de distância se o Guardian for usado pelo paciente.

C. Avaliação de despesas gerais

Sobrecarga criptográfica: Escrevemos um programa de teste em TelosB para registrar o tempo médio de criptografia / descriptografia baseada em ECC, SHA-1 hash, Advanced Encryption Standard (AES) para uma mensagem de 20 bytes e os dados são mostrados na Tabela V.

Sobrecarga de comunicação e operação: As informações de tempo para as operações críticas em diferentes cenários são fornecidas na Tabela VI. Em um caso de autenticação, em média

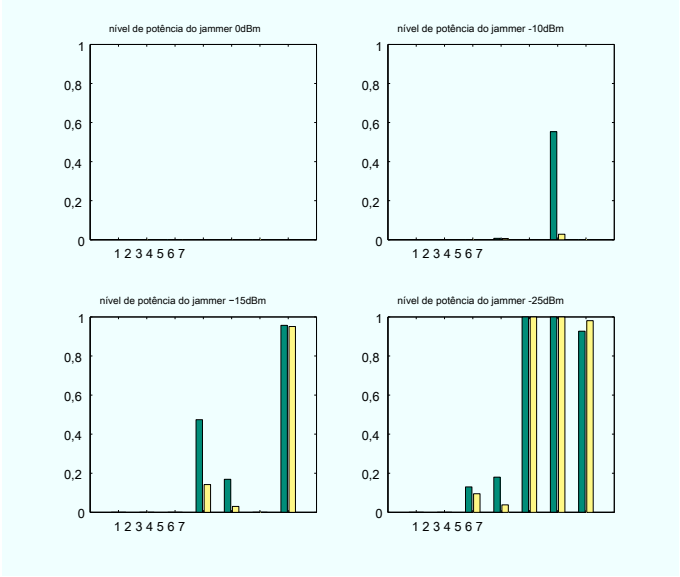

Fig. 11: Resultados da eficácia do bloqueio defensivo do Guardian (atuou como bloqueador) ao mirar no IMD. O eixo X significa a distância entre o Guardian e o ponto central do IMD e o programador malicioso em pés, e o eixo Y é a taxa de sucesso da entrega. A potência de transmissão do IMD é configurada para -5 dBm durante todo o experimento. Para cada distância, foram realizadas duas tentativas.

TABELA V: Informações de tempo de segurança

Encriptação	Decifrar	SHA-1 Hash	AES
3,3 s	1,7 s	4 ms	1 ms

o Guardian leva 3821 em para autenticar um programador no total. Está dividido em (1) 1550 em para o programador gerar uma assinatura do determinado desafio (20 bytes aleatórios
dados), (2) 2221 em para o Guardian verificar esta assinatura, e (3 1) 870 50 em para outro overhead de comunicação. Quando o Guardião não está presente, o processo de condição de emergência (Fig. 4) custa aproximadamente 512 + 14 = 526 em antes que o IMD aceite o programador. Se ocorrer o bloqueio defensivo, a sessão irá ser negado pelo IMD em cerca de 1501 em desde o recebimento da solicitação.

TABELA VI: Informações de tempo do protótipo

Sobrecarga no tempo (em)		
Situação	Operação	A sobrecarga
Autenticação	Assinando (20 bytes)	1550
	Verificação (20 bytes)	2221
	Outras	50
Guardião Removido	Transferência de Desafio	512
	Outras	14
Guardian Jamming	Sessão Negada	1501

VIII. C ONCLUSÃO

Neste documento, propomos o IMDGuard, um esquema de segurança abrangente para proteger dispositivos cardíacos implantáveis em termos de segurança e confiabilidade. O protótipo do IMDGuard é implementado para demonstrar sua funcionalidade de proteção de IMDs na prática.

UMA RECONHECIMENTO

Os autores gostariam de agradecer a todos os revisores por seus comentários úteis. Este projeto foi apoiado em parte pelos subsídios da Fundação Nacional de Ciências dos EUA CNS-0831904 e pelo CAREER Award CNS-0747108.

R EFERÊNCIAS

[1] Comunicações RF de potência ultrabaixa para aplicação médica implantada e Sistemas de Ciclo de Trabalho Baixo. http://www.zarlink.com/zarlink/lowpower-rf dutycycle wp nov06.pdf.

[2] Regras e regulamentos da FCC, Plano de Banda MICS. 2003

[3] S. Bao, C. Poon, Y. Zhang e L. Shen. Usando o tempo informações de batimentos cardíacos como um identificador de entidade para proteger a rede de sensores corporais. *IEEE Trans Inf Technol Biomed,* 12 (6): 772-9, 2008

[4] S. Cherukuri, K. Venkatasubramanian e S. Gupta. BioSec: A abordagem baseada em biometria para proteger a comunicação em redes sem fio de biossensores implantados no corpo humano. *Conferência Internacional sobre Workshops de Processamento Paralelo,* 2003

[5] T. Denning, K. Fu e T. Kohno. Ausência faz o coração crescer ponderar: novas orientações para segurança de dispositivos médicos implantáveis. Dentro *HotSec 2008.*

[6] T. Denning, Y. Matsuoka e T. Kohno. Neurossegurança: segurança e privacidade para dispositivos neurais. *Foco Neurocirúrgico,* 2009.

[7] D. Halperin, T. Heydt-Benjamin, K. Fu, T. Kohno e W. Maisel. Segurança e privacidade para dispositivos médicos implantáveis. *IEEE Pervasive Computing 2008.*

[8] D. Halperin, T. Heydt-Benjamin, B. Ransford, S. Clark, B. Defend, W. Morgan, K. Fu, T. Kohno e W. Maisel. Marca-passos e desfibriladores cardíacos implantáveis: ataques de rádio por software e defesas de potência zero. Dentro *Simpósio IEEE sobre Segurança e Privacidade, 2008.*

[9] L. Sang e A. Arora. Capacidades sem fio de baixa potência jammers. Relatório Técnico OSU-CISRC-5/08-TR24, Ohio State Univ., 2008.

[10] C. Li, C. Zheng e C. Tai. Detecção de característica de ECG pontos usando transformações wavelet. *IEEE Trans. em Engenharia Biomédica,* 42 (1), 1995.

[11] G. Marcus. Imagem do IMD. http://knol.google.com/k/-/-/hCjLTV2A/bdmV3w / ICD.CXR.jpg.

[12] J. Martinez, R. Almeida, S. Olmos, A. Rocha e P. Laguna. UMA Delineador de ECG baseado em wavelet: avaliação em bancos de dados padrão. *IEEE Trans. em Engenharia Biomédica,* 51 (4), 2004.

[13] I. Martinovic, P. Pichota e J. Schmitt. Jamming for good: a nova abordagem para comunicação autêntica em RSSFs. Dentro *Wisec 2009.*

[14] C. Poon, Y. Zhang e S. Bao. Um novo método biométrico para redes seguras de sensores de área corporal sem fio para telemedicina e m-health. *IEEE Communications Magazine,* 44 (4): 73–81, 2006. [15] A. Rukhin, J. Soto, J. Nechvatal, M. Smid e E. Barker. Um conjunto de testes estatísticos para geradores de números aleatórios e pseudo-aleatórios para aplicações criptográficas. *NIST,* 2001.

[16] M. Strasser, C. Pöpper, S. Capkun e M. Cagalj. Jamming-estabelecimento de chave resistente usando salto de frequência descoordenado. Dentro *Simpósio IEEE sobre Segurança e Privacidade, 2008.*

[17] K. Venkatasubramanian, A. Banerjee e S. Gupta. Ekg-acordo de chave baseado em redes de sensores corporais. Dentro *Workshops de comunicação de computador,* 2008

[18] H. Wang, B. Sheng e Q. Li. Criptografia de curva elíptica-controle de acesso baseado em redes de sensores. *Jornal Internacional de Segurança e Redes,* 1 (3): 127–137, 2006.

[19] W. Xu, W. Trappe, Y. Zhang e T. Wood. A viabilidade de lançar e detectar ataques de interferência em redes sem fio. Dentro *MobiHoc 2005.*