SOTARU: Abordagem Baseada em Blockchain de Consórcio para Atualização Remota de Firmware no Cenário da IoT*

Cleber S. Peter[1], Lucas Penning[1], Alexandra Zimpeck[1], Felipe Marques[2], Jorge Barbosa[3], Adenauer Yamin[1,2]

[1]Mestrado em Engenharia Eletrônica e Computação (MEEC)
Universidade Católica de Pelotas (UCPEL)

[2]Programa de Pós-Graduação em Computação (PPGC)
Universidade Federal de Pelotas (UFPEL)

[3]Programa de Pós-Graduação em Computação Aplicada (PPGCA)
Universidade do Vale do Rio dos Sinos (UNISINOS)

Abstract. *Due to the required scalability, providing the storage and distribution of updates for the devices that make up the Internet of Things (IoT) has been a high challenge for network infrastructures. In this scenario, this article presents a new approach, called SOTARU, which proposes the use of a Blockchain consortium between embedded system manufacturers to provide a shared and decentralized, but also secure, infrastructure. The proposal was deployed on EXEHDA middleware nodes and through the distributed network emulator Common Open Research Emulator (CORE), it was also possible to evaluate its security and robustness. As a result, it was found that SOTARU stands out in terms of security when compared to other approaches proposed in the literature, as well as being functional even in high latency scenarios.*

Resumo. *Devido à elasticidade requerida, prover o armazenamento e distribuição das atualizações para os dispositivos que compõem a Internet das Coisas (IoT) tem se mostrado um elevado desafio para as infraestruturas de rede. Neste cenário, este artigo apresenta uma nova abordagem, denominada SOTARU, que propõe a utilização de uma Blockchain de consórcio entre os fabricantes de sistemas embarcados para fornecer uma infraestrutura compartilhada e descentralizada, mas também segura. A proposta foi implantada sobre os nodos do middleware EXEHDA e através do emulador de redes distribuídas Common Open Research Emulator (CORE) foi possível avaliar também a sua segurança e robustez. Como resultado, verificou-se que a SOTARU se sobressai em termos de segurança quando comparada às demais abordagens propostas pela literatura, bem como se mostra funcional mesmo em cenários de alta latência.*

1. Introdução

A atualização remota da aplicação executada pelos sistemas embarcados, comumente denominada de *Over-The-Air* (OTA), se mostra imprescindível no contexto das soluções para IoT. Afinal a correção de eventuais falhas, atualização dos protocolos de segurança,

*Trabalho realizado com apoio da CAPES - Brasil

adaptação a novos cenários de operação, bem como o emprego dos conceitos de integração e entrega contínuos (CI/CD) dependem diretamente do emprego de estratégias de OTA [Lopez-Viana et al. 2020].

Apesar das propostas para OTA viabilizarem a melhoria contínua das medidas de segurança adotadas em sistemas embarcados, estas acabam constituindo também um vetor para exploração de falhas de segurança. Afinal, conforme discutido em [Bettayeb et al. 2019], um mecanismo de atualização inseguro é suscetível a diferentes tipos de ataques de adulteração de *Firmware*. Logo, as abordagens de OTA, por realizarem a gestão da propriedade intelectual dos fabricantes, devem atentar para a tríade da segurança da informação e conferir: confidencialidade, integridade e disponibilidade (CID) a todas as etapas do processo de atualização.

Considerando isto, com o intuito de propor requisitos mínimos de segurança a serem atendidos por soluções de OTA para sistemas embarcados, um grupo de trabalho do *Internet Engineering Task Force* (IETF) propôs uma especificação denominada de *Software Updates for Internet of Things* (SUIT) [Moran et al. 2021]. O SUIT propõe a utilização de *End-to-End Encryption* (E2EE) entre o fabricante e o dispositivo a fim de garantir os requisitos de integridade e confidencialidade da atualização, mesmo com o emprego de um canal de comunicação público como os oferecidos pela IoT.

Contudo, a especificação proposta pelo IETF se caracteriza por ser uma abordagem centralizada na qual um único servidor é responsável por armazenar e distribuir os arquivos de atualização para todos os dispositivos. Sendo assim, o SUIT não garante a disponibilidade do serviço visto que apresenta no servidor de atualização um *Single Point of Failure* (SPOF) com a capacidade de comprometer a atualização de milhares ou até mesmo milhões de dispositivos [Choi and Lee 2020].

Com o objetivo de mitigar o SPOF, trabalhos recentes propuseram que o registro das emissões de *Firmware* fosse realizada em Blockchains públicas de tal forma a se aproveitar do caráter distribuído e imutável dos dados armazenados na rede [Yohan and Lo 2018, Zhao et al. 2019, Anastasiou et al. 2020]. Entretanto, nesta abordagem a disseminação dos dados é vinculada ao incentivo financeiro fornecido pelo fabricante como pagamento pelo armazenamento e distribuição das atualizações, ou seja, a emissão da atualização é condicionada aos recursos do fabricante, visto que o custo associado é volátil e regulado pelo mercado que se forma em torno da solução.

Posto isto, este artigo apresenta a abordagem SOTARU que visa mitigar o SPOF pelo emprego de uma Blockchain de consórcio entre os fabricantes de sistemas embarcados. A SOTARU propõe o compartilhamento de recursos computacionais como forma de prover uma infraestrutura descentralizada e altamente disponível tanto para o armazenamento quanto para a distribuição das atualizações de *Firmware* aos dispositivos que compõem a Internet das Coisas.

Além desta introdução o artigo contempla a seguinte organização: a Seção 2 discute os trabalhos relacionados à abordagem SOTARU, a Seção 3 apresenta sua arquitetura em conjunto com seus principais componentes e a Seção 4 detalha a metodologia empregada em sua avaliação assim como os resultados obtidos. Por fim, a Seção 5 apresenta as considerações finais e discute a sequência da pesquisa.

2. Trabalhos Relacionados

O emprego de Blockchains como forma de prover uma infraestrutura distribuída e tolerante a falhas para as soluções da IoT tem sido explorada em diversas frentes de pesquisa. No âmbito das abordagens de OTA não é diferente, já que recentemente diversos trabalhos adotaram esta tecnologia como base de suas propostas.

Em [Yohan and Lo 2018], os autores propõem uma arquitetura formada pelo Repositório do Fabricante, nodos da Blockchain e dispositivos da IoT. Cada fabricante mantém a confidencialidade e a integridade das suas atualizações armazenando os arquivos em seu próprio repositório. A Blockchain serve somente como um indexador das atualizações armazenando de forma distribuída o histórico de todas as atualizações emitidas na rede. A abordagem proposta pelos autores atende aos requisitos de integridade e confidencialidade através do não compartilhamento dos arquivos de atualização. Todavia a centralização no armazenamento não atende ao requisito de disponibilidade visto que resulta em um SPOF para arquitetura e a torna não tolerante às falhas do Repositório do Fabricante.

Os contratos inteligentes presentes na Blockchain da criptomoeda Ethereum são explorados em [Zhao et al. 2019] para garantir que a remuneração dos agentes distribuidores seja realizada somente após a distribuição da atualização aos dispositivos. Apesar de atender parcialmente aos requisitos de segurança elencados, a proposta depende diretamente do interesse de terceiros para estabelecer a rede de armazenamento e distribuição descentralizada, interesse o qual está condicionado ao incentivo financeiro fornecido pelo fabricante. Além disso, este sistema de recompensa acentua a baixa disponibilidade dos arquivos de atualização que se destinam a uma quantidade reduzida de dispositivos da IoT, afinal a remuneração do Distribuidor é função da quantidade de atualizações aplicadas, o que incentiva a priorização das atualizações com maior público alvo.

Os autores em [Baza et al. 2019] abordam o problema da atualização remota dos sistemas automotivos, cujo grande diferencial em relação aos demais dispositivos da IoT reside em sua mobilidade. Nesta abordagem os próprios veículos se tornam Distribuidores das atualizações tendo a sua recompensa, de forma semelhante a [Zhao et al. 2019], vinculada a uma prova de distribuição solicitada por um contrato inteligente publicado na Blockchain. Os veículos nesta abordagem permitem a intercomunicação através da formação de redes oportunistas e se tornam excelentes alternativas para o armazenamento e distribuição descentralizada das atualizações. No entanto, a atualização de um Veículo Alvo está condicionada ao encontro com um Veículo Distribuidor, o que pode atrasar a aplicação de uma atualização crítica e colocar em risco a operação do veículo.

Em [Anastasiou et al. 2020] os autores propõem uma abordagem de OTA segura e distribuída também baseada em Blockchain, porém destinada à atualização de dispositivos conectados através de redes *Low Power Wide Area Network* (LPWAN). Nesta proposta o emissor da atualização armazena na Blockchain um arquivo de metadados referente à atualização e negocia com o servidor de aplicação uma janela para distribuição em Multicast. A abordagem se mostra restrita aos dispositivos conectados através de LPWAN e, portanto, não satisfaz a premissa de interoperabilidade esperada das soluções da IoT. Além disso, os autores não discutem mecanismos para conferir confidencialidade ao processo, o que torna a proposta suscetível a ataques de engenharia reversa.

Enfim, a Tabela 1 sintetiza a discussão dos trabalhos relacionados realizando uma análise comparativa do tipo de Blockchain utilizada, os requisitos de segurança observados e a capacidade de interoperação de cada uma das propostas em relação à SOTARU.

Tabela 1. Análise Comparativa dos Trabalhos Relacionados

Id	Artigo	Blockchain	CID	Suporte à Interoperabilidade
A1	[Yohan and Lo 2018]	Pública	CI	Sim
A2	[Zhao et al. 2019]	Pública	CI	Sim
A3	[Baza et al. 2019]	Consórcio	CI	Não
A4	[Anastasiou et al. 2020]	Pública	ID	Não
-	Este Trabalho	Consórcio	CID	Sim

Apesar de se mostrarem promissores, os trabalhos elencados não garantem de forma simultânea os três requisitos de segurança esperados de uma solução de OTA. Sendo assim, a seção a seguir apresenta a proposta deste trabalho cujo objetivo é preencher esta lacuna de pesquisa e conferir, através do compartilhamento de recursos entre os fabricantes de sistemas embarcados, confidencialidade, integridade e disponibilidade ao processo de OTA mesmo no cenário amplamente heterogêneo da IoT.

3. Abordagem SOTARU: Concepção

A SOTARU, cujo nome significa *Secure Over The Air Remote Update*, se destina ao ambiente severamente heterogêneo dos *middlewares* da IoT, de tal forma que o requisito de interoperabilidade é norteador do processo de concepção detalhado a seguir.

3.1. Arquitetura

A Arquitetura da abordagem SOTARU, ilustrada pela Figura 1, é executada de forma distribuída pelos servidores que compõem a infraestrutura do sistema de atualização e têm seus módulos detalhados a seguir.

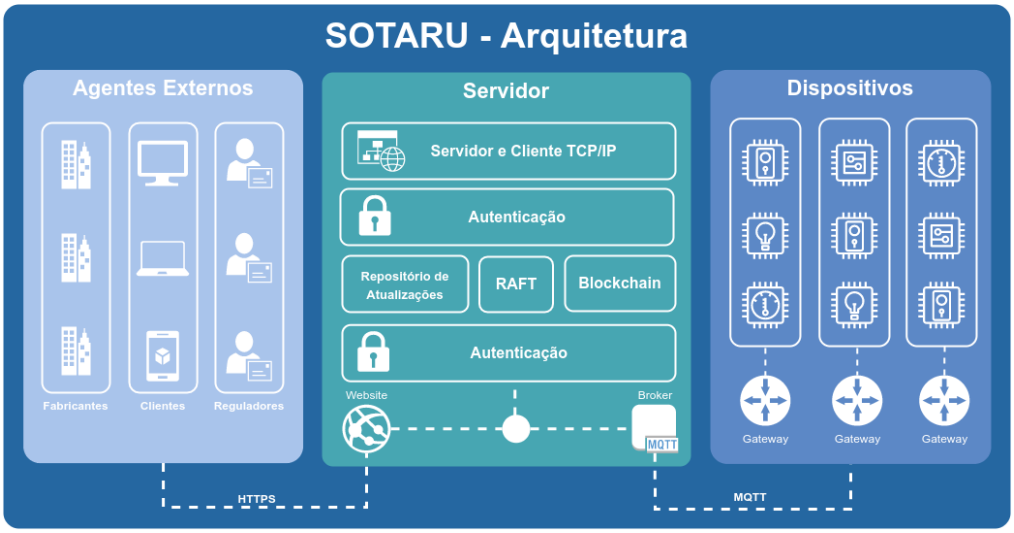

Figura 1. Arquitetura da Abordagem SOTARU

Os módulos Website e Broker são os responsáveis por realizar a interface de comunicação entre os Agentes e a Arquitetura. A divisão em dois módulos visa facilitar o acesso dos Dispositivos mediante o emprego do protocolo MQTT, largamente adotado nas soluções da IoT devido ao seu custo computacional reduzido quando comparado ao HTTP. A divisão, entretanto, também se aplica aos recursos disponibilizados, visto que através do Website os Agentes Externos possuem acesso a uma série de recursos como a emissão e monitoramento da distribuição das atualizações, o cadastro e listagem dos membros do consórcio e seus Dispositivos, assim como o acesso ao histórico de atualizações. Já os dispositivos por intermédio do Broker têm seu acesso restrito ao monitoramento e *download* das atualizações.

O Módulo de Autenticação se mostra um elemento essencial para a proposta, atuando como um *firewall* para a arquitetura, este módulo regula o acesso aos recursos conforme a identidade e nível de permissão do solicitante. Neste sentido, a tomada de decisão com relação à liberação de acesso está diretamente relacionada ao recurso solicitado e aos dados cadastrais dos Agentes armazenados de forma distribuída e descentralizada na Blockchain.

A Blockchain em conjunto com o Repositório de Atualizações representam os módulos de persistência da proposta. A natureza pública, sequencial e imutável da Blockchain conduz sua utilização para o armazenamento de eventos temporais sobre os quais não há interesse de alteração [Wust and Gervais 2018]. Sendo assim, nesta proposta a Blockchain é empregada para o armazenamento tanto do histórico de atualizações dos dispositivos quanto das informações cadastrais dos fabricantes membros do consórcio.

O armazenamento dos dados cadastrais na Blockchain visa explorar o seu caráter descentralizado com o objetivo de conferir flexibilidade ao fabricante para o gerenciamento e monitoramento de seus dispositivos. Afinal, neste cenário o fabricante pode gerenciar seus dispositivos não somente através de seu servidor, mas também mediante conexão com qualquer um dos nodos da rede tornando a comunicação possível mesmo mediante a indisponibilidade de sua infraestrutura.

Os arquivos de atualização, por outro lado, se armazenados na Blockchain seriam replicados em todos os nodos da rede diminuindo o grau de escalabilidade da solução. Posto isto, na abordagem SOTARU o Repositório de Atualizações de cada membro do consórcio armazena somente os arquivos de atualização dos dispositivos que em algum momento se conectaram ao servidor em busca de Atualizações. Esta estratégia garante o requisito de disponibilidade efetuando a replicação somente conforme necessidade.

O módulo de Consenso, por sua vez, refere-se à implementação do algoritmo RAFT [Ongaro and Ousterhout 2014], empregado nesta proposta para garantir a sincronização e consistência dos dados armazenados pela rede. Como premissa operacional, a proposta deve suportar a geração distribuída e simultânea de registros, porém devido às características de serialização e ordenação da Blockchain isto se mostra um desafio elevado. Neste cenário, o RAFT através da centralização proporcionada pela figura do líder, se mostra uma alternativa em potencial.

Por fim, o módulo TCP/IP, subdividido em um serviço cliente e outro servidor, representa a camada de conectividade da abordagem e permite a interconexão entre os servidores que compõem a infraestrutura do sistema de atualização.

3.2. Estratégia de Segurança

A estratégia de segurança adotada neste trabalho, semelhante à apresentada em [Peter et al. 2021], visa atender aos requisitos de segurança CID mediante a utilização de E2EE. Entretanto, para conferir confidencialidade aos dados é necessário que o fabricante e seus dispositivos compartilhem um segredo em comum, o que se mostra um elevado desafio no contexto amplamente distribuído da IoT.

Sendo assim, nesta proposta o segredo R_K empregado pelo fabricante para criptografar o conteúdo da atualização é gerado de forma randômica a cada emissão e compartilhado de forma cifrada com os dispositivos através de um arquivo de metadados, denominado de Manifesto. A garantia de integridade da atualização, por outro lado, é diretamente dependente dos mecanismos de assinatura digital e autenticação utilizados, cuja base é fundamentada na distribuição de pares de chaves criptográficas assimétricas, realizada conforme segue.

Durante o cadastramento de um fabricante, o módulo Website gera um par de chaves pública e privada (P_F, S_F). A chave privada do fabricante S_F é utilizada para assinar digitalmente os arquivos da atualização e deve ser mantida em segredo pelo próprio. A chave pública P_F, em contrapartida, é armazenada na Blockchain, nos dispositivos mantidos por este fabricante e serve para autenticação e verificação de integridade das atualizações. O não sigilo da S_F habilita a emissão de atualizações por agentes maliciosos. Sendo assim, como critério de concepção para o SOTARU, optou-se pelo não armazenamento de chaves privadas nos servidores da rede.

A cada novo Projeto criado também é associado um par de chaves pública e privada (P_P, S_P). A chave privada do Projeto S_P deve ser instalada em todos os dispositivos vinculados ao Projeto e é utilizada para assinar digitalmente a emissão das notificações enviadas à rede sobre a execução bem sucedida de uma atualização. A chave pública do projeto P_P, por sua vez, é armazenada na Blockchain e permite tanto autenticar as mensagens enviadas pelos dispositivos quanto estabelecer, através do Manifesto, um segredo em comum com os servidores.

3.3. Fases da Atualização

O procedimento de atualização dos dispositivos através da SOTARU é dividido em duas fases denominadas de emissão e distribuição. Ambas se referem à interação dos Agentes com a rede, porém a primeira retrata as ações do Fabricante enquanto a segunda a dos Dispositivos.

Na fase de emissão, ilustrada pelo diagrama de sequência da Figura 2, primeiramente o fabricante fornece ao módulo Website através de uma conexão HTTPS os arquivos da atualização, sua chave privada S_F e também as informações pertinentes ao arquivo de Manifesto da nova versão. Em seguida, o Website gera uma chave de sessão randômica R_K que é utilizada para criptografar os arquivos de atualização antes que estes sejam armazenados no Repositório de Atualizações do Servidor. Após, a chave pública P_P do Projeto ao qual se destina a atualização é obtida da Blockchain e utilizada para criptografar R_K produzindo S_K que é adicionada ao Manifesto. Enfim a chave S_F é empregada para produzir a assinatura digital dos arquivos de atualização e também do próprio Manifesto que é publicado na Blockchain. A etapa final compreende a sincronização da Blockchain entre os membros do consórcio realizada pelo algoritmo de consenso.

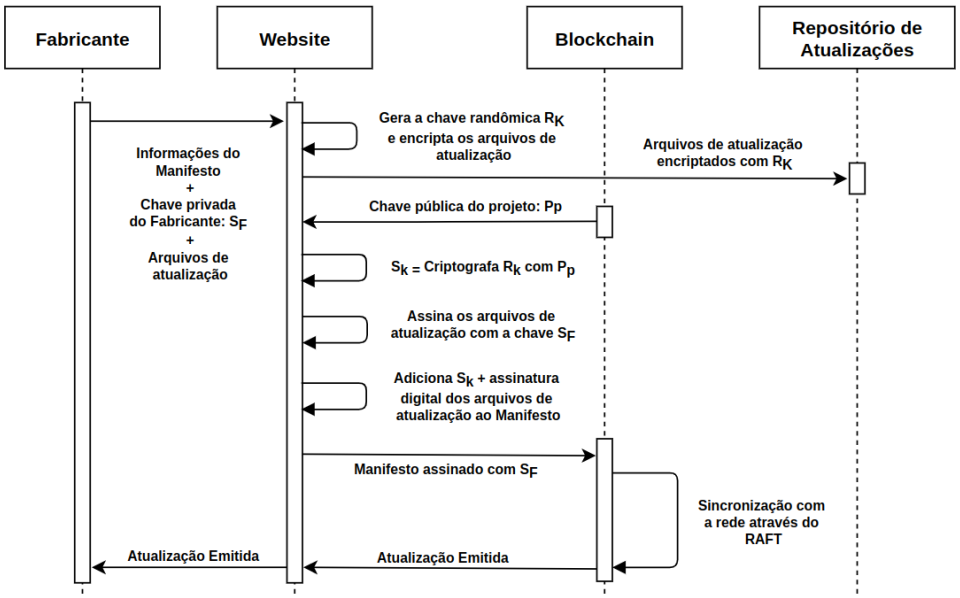

Figura 2. Diagrama de Sequência da Fase de Emissão

Na fase de Distribuição, ilustrada pelo diagrama de sequência da Figura 3, o dispositivo busca por atualizações através de uma estratégia de *publish/subscribe* direcionada ao módulo Broker presente em um dos servidores membros do consórcio. A ação de inscrição realizada pelo dispositivo é o gatilho necessário para que o servidor passe a monitorar os avanços da Blockchain em busca de atualizações destinadas ao dispositivo. Após a detecção de uma nova versão, o servidor disponibiliza o arquivo de Manifesto ao Dispositivo que, através da chave pública do fabricante P_F, verifica sua autenticidade. Se comprovada, o dispositivo descriptografa a chave S_K presente no Manifesto com a chave privada do Projeto S_P para obter a chave de sessão randômica R_K.

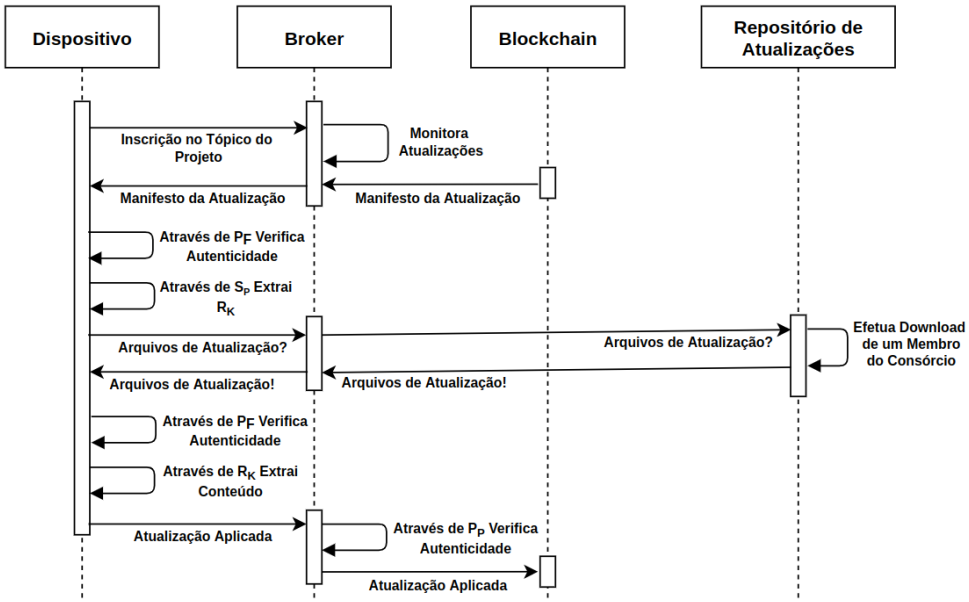

Figura 3. Diagrama de Sequência da Fase de Distribuição

De posse de R_K, o dispositivo então solicita ao servidor o download dos arquivos da atualização. Após o recebimento, o dispositivo verifica a autenticidade dos arquivos com a chave pública do fabricante P_F que, se comprovada, conduz ao início do processo de descriptografia do conteúdo da atualização através da chave de sessão randômica R_K obtida anteriormente. Neste momento o dispositivo, baseado em suas características operacionais, determina o instante mais oportuno para a aplicação da atualização.

4. Avaliação

Esta seção tem por objetivo apresentar a avaliação realizada sobre a proposta deste trabalho. Com este propósito, a seguir são elencados os aspectos selecionados para avaliação bem como são detalhados os métodos e ferramentas empregadas no processo.

4.1. Avaliação de Segurança

A seguir é analisado o comportamento da abordagem SOTARU quando submetida a ataques cibernéticos. Os ataques, apresentados na forma de uma árvore de ataque pela Figura 4, foram selecionados com base em sua aplicabilidade no contexto das abordagens de OTA [Bettayeb et al. 2019].

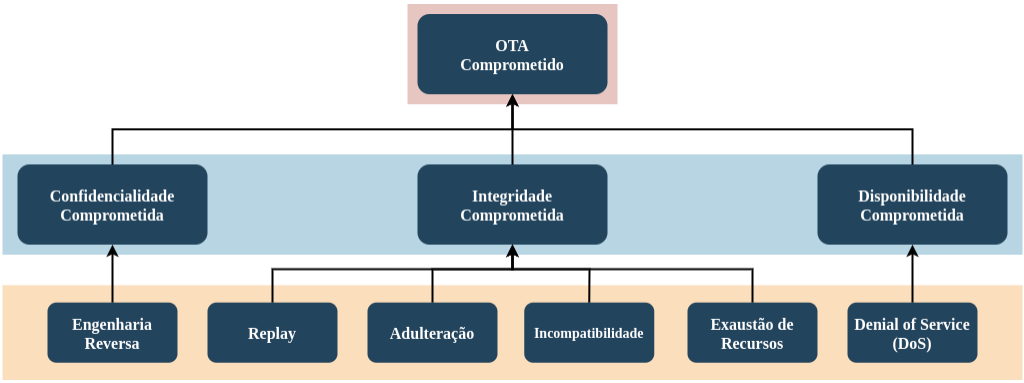

Figura 4. Árvore de Ataque Proposta

Na abordagem SOTARU a proteção contra o ataque de *Replay* se dá pela utilização de um número sequencial que indica a versão do *Firmware*. Desta forma, o sistema embarcado somente é atualizado para versões superiores à atual. A garantia de integridade e autenticidade da atualização, por outro lado, são resultantes dos mecanismos de assinatura digital e autenticação utilizados, os quais tornam a proposta também segura contra ataques de Adulteração de *Firmware*.

A proteção contra o ataque de Incompatibilidade é garantida pela associação entre o dispositivo e o Projeto, ou seja, o dispositivo somente executa uma atualização se esta tiver sido emitida para o Projeto ao qual pertence. Devido ao emprego do recurso de E2EE, a proposta também confere confidencialidade ao processo de atualização, o que a torna segura contra ataques de Engenharia Reversa. O ataque de Exaustão de Recursos, por sua vez, depende do envio contínuo de atualizações fraudulentas ao dispositivo. Na SOTARU, entretanto, a autenticação do arquivo de Manifesto ocorre de forma prévia ao download da atualização mitigando este ataque. Por fim, a disponibilidade e a proteção contra ataques de DoS direcionados à rede são garantidos pela adoção de uma infraestrutura distribuída e descentralizada tanto para o armazenamento quanto para a distribuição das atualizações.

A Tabela 2 sintetiza a análise realizada comparando a SOTARU às abordagens discutidas na Seção 2. É possível constatar que a proposta deste trabalho atende simultaneamente aos três requisitos de segurança elencados, assim como se mostra segura em relação aos principais ataques cibernéticos aplicáveis contra abordagens de OTA.

Tabela 2. Comparação de Vulnerabilidade entre as Abordagens

Ataques	A1	A2	A3	A4	Este Trabalho
Replay	●	●	○	○	○
Adulteração	○	○	○	○	○
Incompatibilidade	●	○	○	○	○
Engenharia Reversa	○	○	○	●	○
Exaustão de Recursos	○	○	●	●	○
Denial of Service (DoS)	●	●	●	○	○

● Vulnerável ○ Não Vulnerável

4.2. Avaliação de Robustez

A avaliação de robustez da proposta foi realizada com a emulação de operação da abordagem SOTARU sobre uma infraestrutura de rede distribuída. Para tanto foi empregado o emulador CORE [Ahrenholz et al. 2008] como forma de emular diferentes configurações de rede nas quais o comportamento da abordagem proposta é avaliado.

Primeiramente foi elaborada uma rede ideal, sem atrasos, definida pela matriz $R = F.[A_{ij}]_{SxN}$, tal que S representa a quantidade de sub-redes, N a quantidade de nodos de cada sub-rede e $A_{ij} = (i * 8) + (j + 1)$. Posteriormente foi implementado no CORE uma topologia com 32 servidores de parâmetros $S = 4$, $N = 8$, $T_L = 1s$ e $T_R = [100, 300]ms$. Em seguida, a emulação da eleição do líder da rede foi acompanhada através de um diagrama de tempo, ilustrado pela Figura 5, que contém a evolução de estado de cada um dos servidores da rede bem como as mensagens trocadas entre estes.

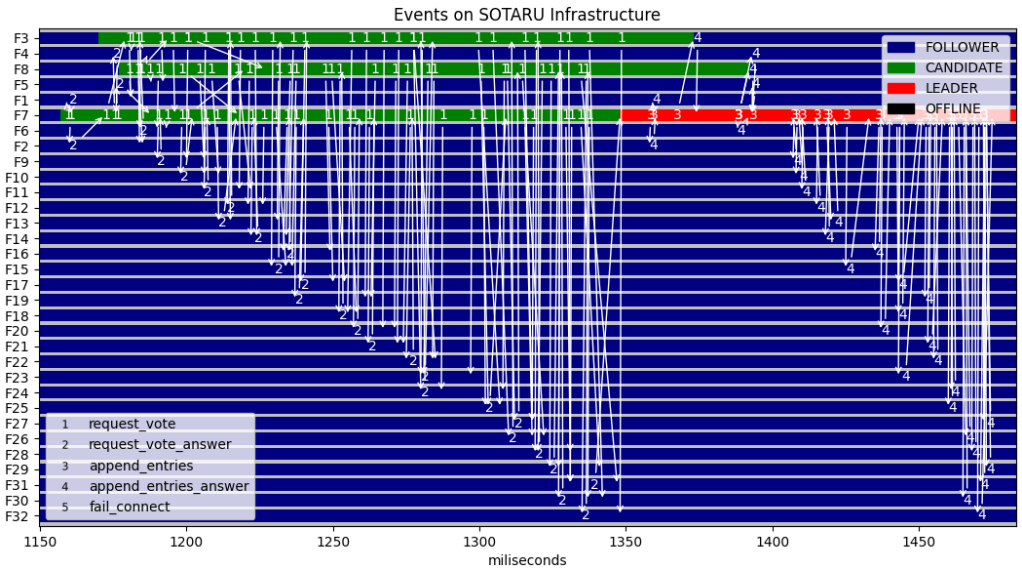

Figura 5. Resultado da Eleição Emulada no CORE

Na eleição ilustrada pela Figura 5 é possível verificar que o nodo F7 foi o primeiro a iniciar sua candidatura evoluindo para o estado CANDIDATE, porém logo em seguida os nodos F3 e F8 também passaram a solicitar votos aos demais integrantes da rede. Devido a ausência de atrasos na rede e por ter sido o primeiro candidato, o nodo F7 recebe uma quantidade suficiente de votos e evolui seu estado para LEADER.

Posteriormente com o objetivo de avaliar o funcionamento da abordagem SOTARU sob uma condição de operação mais realista, as estatísticas de latência do serviço de nuvem Azure [Mahesh 2021] foram utilizadas para mapear uma distribuição heterogênea de atrasos na comunicação entre os nodos. Neste cenário, é definida a matriz $L = [A_{ij}]_{SxN}$ tal que $A_{ij} = (i * 100) + (j * 10)$ define a latência em milissegundos de cada nodo conforme sua posição na rede. Após foram realizadas emulações do processo de inicialização da rede em que o tempo necessário para o término da eleição bem como o nodo eleito foram registrados. Como resultado, a Figura 6 apresenta um histograma do tempo necessário para conclusão da eleição e as Figuras 7 e 8 mostram os histogramas dos servidores eleitos nas redes sem e com atrasos, respectivamente.

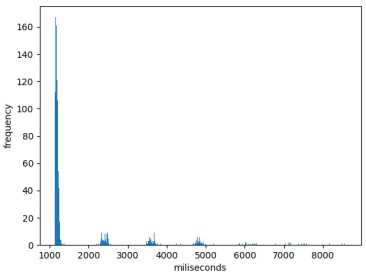

Figura 6. Eleição - Duração

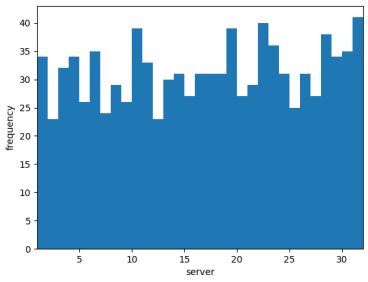

Figura 7. Eleição - Rede Ideal

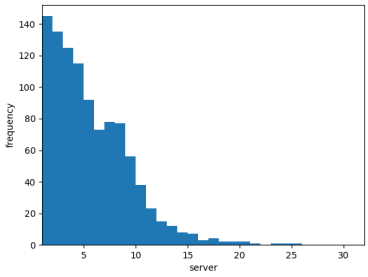

Figura 8. Eleição - Rede Real

É possível constatar através da Figura 6 que aproximadamente 80% das eleições realizadas foram concluídas com somente uma eleição, enquanto que menos de 3% das eleições necessitaram de 5 ou mais eleições. A Figura 7 mostra o caráter randômico e portanto democrático do processo de eleição da rede ideal, afinal o histograma obtido reflete uma distribuição uniforme de probabilidade na escolha do líder. A Figura 8, por sua vez, apresenta uma Distribuição Exponencial o que revela a tendência para a eleição dos nodos com menor latência associada. Este resultado é devido à necessidade de confirmação das mensagens recebidas pelo destinatário, de tal forma que a latência de comunicação entre dois nodos é dada pela soma da latência individual de cada nodo.

Posto isto, fica claro que em uma rede real os nodos com menor latência associada possuirão uma latência de comunicação média menor em relação aos demais membros da rede e como consequência levarão vantagem na disputa pelos votos. Tal característica é vantajosa quando considerada a etapa de sincronização dos dados, afinal estes serão mais eficientes para a disseminação das atualizações. Todavia o desbalanceamento de latência entre os membros da rede sacrifica o caráter democrático do processo eleitoral favorecendo os fabricantes com mais recursos computacionais.

Ressalta-se que a tolerância às falhas dos servidores que compõem a infraestrutura é provida pelo algoritmo de consenso adotado o qual garante a sincronização dos nodos desatualizados e, mediante a falha do líder, prevê a realização de novas eleições.

4.3. Prova de Conceito

Com o objetivo de avaliar a interoperabilidade da proposta uma prova de conceito foi desenvolvida sobre os nodos do EXEHDA, um middleware adaptativo ao contexto e baseado em serviços, que visa criar e gerenciar um ambiente ubíquo [Machado et al. 2017]. Primeiramente foram instanciados quatro nodos EXEHDAbase, cada qual com seu domínio, para representar os fabricantes. Em seguida, foi associado a cada fabricante um projeto e um nodo EXEHDAgateway com o propósito de refletir os dispositivos aos quais as atualizações se destinam.

Posteriormente foram emitidas quatro atualizações, duas destinadas ao mesmo projeto. A Figura 9 ilustra a aba de Releases do módulo Website que permite ao usuário verificar o histórico de atualizações emitidas pelos fabricantes membros do consórcio. Para cada atualização publicada é possível também constatar o fabricante emissor, o número da versão correspondente, data da emissão, os servidores que atualmente a hospedam, os dispositivos aos quais se destina e também os próprios arquivos que a compõem.

Figura 9. Histórico de Atualizações através da Aba Releases do Website

Com o objetivo de avaliar o processo de sincronização realizado pelo líder, durante as emissões das duas últimas atualizações dois fabricantes são desconectados da rede. Como consequência o consenso é atingido somente entre os servidores que se mantiveram online gerando dessincronização do histórico de atualizações armazenado pelos membros do consórcio. A Figura 10 apresenta a aba servidores que registra o momento subsequente no qual um dos servidores é reconectado e se encontra em processo de sincronização. Esta aba ainda permite ao usuário verificar qual membro do consórcio é o líder da rede assim como averiguar o grau de sincronização da rede com relação à atualização da Blockchain.

Figura 10. Visão Geral da Rede através da Aba Servidores do Website

5. Conclusões

Este trabalho abordou a SOTARU, uma abordagem de OTA baseada no compartilhamento de recursos entre os fabricantes de sistemas embarcados. Os resultados obtidos permitem concluir que a proposta é viável no contexto amplamente heterogêneo da IoT, afinal atende aos requisitos de segurança requisitados de abordagens de OTA, bem como se mostra robusta e confiável mesmo quando empregada em redes de alta latência.

O modelo de falhas adotado na condução deste artigo permite aos servidores integrantes do consórcio apresentarem qualquer tipo de inatividade em relação ao algoritmo de consenso empregado, entretanto, não é previsto que estes atuem de forma maliciosa. Sendo assim, como direcionamento para trabalhos futuros sugere-se o estudo de algoritmos de consenso também tolerantes a falhas bizantinas.

Referências

Ahrenholz, J., Danilov, C., Henderson, T. R., and Kim, J. H. (2008). CORE: A real-time network emulator. In *MILCOM 2008 - 2008 IEEE Military Communications Conference*. IEEE.

Anastasiou, A., Christodoulou, P., Christodoulou, K., Vassiliou, V., and Zinonos, Z. (2020). IoT device firmware update over LoRa: The blockchain solution. In *2020 16th International Conference on Distributed Computing in Sensor Systems (DCOSS)*. IEEE.

Baza, M., Nabil, M., Lasla, N., Fidan, K., Mahmoud, M., and Abdallah, M. (2019). Blockchain-based firmware update scheme tailored for autonomous vehicles. In *2019 IEEE Wireless Communications and Networking Conference (WCNC)*. IEEE.

Bettayeb, M., Nasir, Q., and Talib, M. A. (2019). Firmware update attacks and security for IoT devices. In *Proceedings of the ArabWIC 6th Annual International Conference Research Track on - ArabWIC 2019*. ACM Press.

Choi, S. and Lee, J.-H. (2020). Blockchain-based distributed firmware update architecture for IoT devices. *IEEE Access*, 8:37518–37525.

Lopez-Viana, R., Diaz, J., Diaz, V. H., and Martinez, J.-F. (2020). Continuous delivery of customized SaaS edge applications in highly distributed IoT systems. *IEEE Internet of Things Journal*, 7(10):10189–10199.

Machado, R., Almeida, R. B., da Rosa, D. Y. L., Lopes, J. L. B., Pernas, A. M., and Yamin, A. C. (2017). EXEHDA-HM: A compositional approach to explore contextual information on hybrid models. *Future Gener. Comput. Syst.*, 73:1–12.

Mahesh, N. (2021). Azure network round-trip latency statistics.

Moran, B., Tschofenig, H., Brown, D., and Meriac, M. (2021). A Firmware Update Architecture for Internet of Things. RFC 9019.

Ongaro, D. and Ousterhout, J. (2014). In search of an understandable consensus algorithm. In *Proceedings of the 2014 USENIX Conference on USENIX Annual Technical Conference*, USENIX ATC'14, page 305–320, USA. USENIX Association.

Peter, C. S., Oliveira, T., Monks, E. M., Motta, F. P., Barbosa, J. L. V., and Yamin, A. C. Y. (2021). iota: An approach to secure over-the-air updates on the internet of things scenario. In *Anais do XXVII Simpósio Brasileiro de Sistemas Multimídia e Web*, pages 173–176, Porto Alegre, RS, Brasil. SBC.

Wust, K. and Gervais, A. (2018). Do you need a blockchain? In *2018 Crypto Valley Conference on Blockchain Technology (CVCBT)*. IEEE.

Yohan, A. and Lo, N.-W. (2018). An over-the-blockchain firmware update framework for IoT devices. In *2018 IEEE Conference on Dependable and Secure Computing (DSC)*. IEEE.

Zhao, Y., Liu, Y., Tian, A., Yu, Y., and Du, X. (2019). Blockchain based privacy-preserving software updates with proof-of-delivery for internet of things. *Journal of Parallel and Distributed Computing*, 132:141–149.