UNIVERSIDADE DO ESTADO DE SANTA CATARINA - UDESC
CENTRO DE CIÊNCIAS TECNOLÓGICAS - CCT
MESTRADO EM COMPUTAÇÃO APLICADA

SERGIO HENRIQUE SILVA

ANÁLISE DO CICLO DE VIDA DE CREDENCIAIS EM REDES LORAWAN PRIVADAS PARA IIOT

JOINVILLE

2022

SERGIO HENRIQUE SILVA

ANÁLISE DO CICLO DE VIDA DE CREDENCIAIS EM REDES LORAWAN PRIVADAS PARA IIOT

Dissertação submetida ao Programa de Pós-Graduação em Computação Aplicada do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina, para a obtenção do grau de Mestre em Computação Aplicada.

Orientador: Dr. Charles Christian Miers

JOINVILLE

2022

Silva, Sergio Henrique
Análise do ciclo de vida de credenciais em redes LoraWan privadas para IIoT/ Sergio Henrique Silva – 2022.
102 p.

Orientador: Dr. Charles Christian Miers
Dissertação (mestrado) – Universidade do Estado de Santa Catarina, Centro de Ciências Tecnológicas, Programa de Pós-Graduação em Computação Aplicada, Joinville, 2022.

1. Internet Industrial das Coisas (IIoT). 2. LoraWan. 3. Gerenciamento de Ciclo de Vida de Credenciais I. Miers, Charles Christian. II. Universidade do Estado de Santa Catarina, Centro de Ciências Tecnológicas, Programa de Pós-Graduação em Computação Aplicada. III. Análise do ciclo de vida de credenciais em redes LoraWan privadas para IIoT.

Sergio Henrique Silva

Análise do ciclo de vida de credenciais em redes LoraWan privadas para IIoT

Esta dissertação foi julgada adequada para a obtenção do título de **Mestre em Computação Aplicada** área de de concentração em "Sistemas de Computação", e aprovada em sua forma final pelo Curso de Mestrado em Computação Aplicada do Centro de Ciências Tecnológicas da Universidade do Estado de Santa Catarina.

BANCA EXAMINADORA:

Charles Christian Miers, Dr.
Universidade do Estado de Santa Catarina
Orientador

Membros:

Maurício Aronne Pillon, Dr.
Universidade do Estado de Santa Catarina

Marco Antonio Torrez Rojas, Dr.
Instituto Federal Catarinense

Joinville, 20 de Dezembro de 2022

Dedicado ao desenvolvimento tecno-científico de Santa Catarina e do Brasil.

AGRADECIMENTOS

Em primeiro lugar, agradeço a Universidade do Estado de Santa Catarina (UDESC) pela experiência incrível que se tornou a realização deste curso de mestrado. Agradeço pela irretocável atuação da instituição em tudo que foi preciso para concretização deste sonho. Agradeço ao meu orientador, professor Charles Christian Miers, pelos muitos momentos de valiosíssimas orientações que me conduziram ao término exitoso desta jornada. Agradeço de maneira geral aos professores do Programa de Pós-Graduação em Computação Aplicada (PPGCAP) pelos conhecimentos adquiridos.

Agradeço também a todos os colegas desta jornada no mestrado: Rafael Tenfen, Danilo Farias Carvalho, Maria Teresa, Taynara Dutra, Marcos Vinnicius, Beatriz Bento e também ao Felipe Canever.

Desejo de mencionar também meus amigos de Camboriú, que sempre estiveram ao meu lado, mesmo quando eu estive indisponível nos dias de estudo. Estes também marcam minhas memórias em passeios incríveis, principalmente no Jardim Elétrico de BC, Marcelo Guimarães, João Salomão, Pedro Luiz entre outros! Meu muito obrigado pela parceria, meus queridões.

Quero aqui também tomar nota dos meus queridos amigos da República Localhost, os quais dividi muito do meu tempo na graduação, que tornou-se a base para chegar hoje no término do mestrado. Gabriel Moura Brasil, Guilherme Zorer, Bruno Gularte, Filipe Lima e Marco Nicolodi, obrigado por serem meus irmãos do coração.

Aproveito para agradecer a parceria dos meus colegas de trabalho, que se tornaram verdadeiros amigos: Hugo Rosso, Beatriz Bento, Rodrigo Risetto Terra, Danilo Santos, Fernando Mello, Renan Souza, Vitória Gonçalves, Marcos Ducati, Floriano Neto (que me deve um artigo escrito em conjunto) e Úrsula Rosa, nossa querida Ú.

Agradecer aos meus queridos amigos de Londrina, que me acompanharam em diversos momentos nesta caminhada, especialmente: Kleber Arantes. Gostaria de agradecer também, com apreço, a Mariana Oliveira. Por fim, mas não menos importante, agradecer a minha irmã Débora, minha mãe Cristina e toda minha família pelo apoio de sempre.

De uma forma geral, agradeço a todos e todas que compartilharam comigo estes dois anos de muito trabalho, mas também de muito crescimento. Avante! Agradeço ainda, o apoio da Fundação de Amparo à Pesquisa e Inovação do Estado de Santa Catarina (FAPESC).

"O otimista pode errar, mas o pessimista já começa errando."
Juscelino Kubitschek

RESUMO

A adoção de dispositivos inteligentes na indústria culminou na realização do conceito da Internet Industrial. A incorporação destes dispositivos em contextos industriais tem diversos requisitos. Um destes requisitos é de segurança e autenticidade no controle de acesso e ciclo de vida das identidades. Para tanto, existem recomendações aderentes com as entidades padronizadoras para o ciclo de vida de identidades em *Industrial Internet of Things* (IIoT). A utilização de redes *Low Power Wide Area Network* (LPWAN) LoraWan devido aos requisitos de distância, em alguns projetos, implementa um ciclo de vida de identidades fora do padrão para IIoT. Esta problemática entre outras questões correlatas são alvo desta pesquisa. Por consequência a proposta deste trabalho é uma análise do ciclo de vida de identidades e credenciais no ambiente de IIoT aplicado a redes LoraWan privadas. Esta análise é realizada usando uma solução de gerenciamento de dispositivos (i.e., ChirpStack) capaz de implementar este ciclo de ciclo de vida de identidades em LoraWan em conformidade com os padrões IIoT. Um ambiente de testes com ChirpStack foi empregado para realizar testes com LoraWan 1.1 a fim de analisar se as etapas do referido ciclo de vida. Os resultados da análise revelaram fases que se adéquam ou não aos padrões, permitindo claramente identificar cada fase do ciclo de vida de identidades no LoraWan 1.1 usando ChirpStack.

Palavras-chaves: Internet Industrial das Coisas (IIoT), LoraWan, Controle de Acesso, Gerenciamento de Identidade, Gerenciamento de Ciclo de Vida de Credenciais.

ABSTRACT

The adoption of smart devices in the industry has culminated in the Industrial Internet concept. Incorporating these devices in industrial contexts has several requirements; one of these requirements is security and authenticity in access control and identity lifecycle. Thus, there are recommendations complying with standard entities for the identity lifecycle in *Industrial Internet of Things* (IIoT). Due to distance requirements, in some projects, the use of *Low Power Wide Area Network* (LPWAN) LoraWan networks implements a non-standard identity lifecycle for IIoT. This issue, among other related issues, is the target of our research. Consequently, this work analyzes the lifecycle of identities and credentials in the IIoT environment applied to private LoraWan networks. This analysis was carried out in a device management solution (i.e., ChirpStack) capable of implementing this identity lifecycle in LoraWan in a way that harmonizes with IIoT standards. A testbed environment with Chirpstack was used to carry out tests with LoraWan 1.1 to analyze the stages of the aforementioned life cycle. The analysis results revealed complying and non-complying aspects, allowing to the identification of each phase of the identity lifecycle in LoraWan 1.1 using ChirpStack.

Key-words: Industrial Internet of Things (IIoT), LoraWan, Access Control, Identity Management, Credential Lifecycle Management.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

3GPP *3rd Generation Partnership Project*
5G *Fifth Generation Networks*
SE *Secure Elements*
AES *Advanced Encryption Standard*
ABP *Activation by Personalization*
API *Application Programming Interface*
ANATEL Agência Nacional de Telecomunicações
AppKey *Application encryption Key*
CA Controle de Acesso
CSS *Chirp Spread Spectrum*
DevEUI *End-device serial Unique Identifier*
ECB *Electronic CodeBook*
ECDH *Elliptic Curve Diffie-Hellman*
EUI *Extended Unique Identifier*
FMEA *Failure Mode and Effect Analysis*
FOTA *Firmware Over-The-Air*
GHz Giga Hertz
gRPC *Google Remote Procedure Call*
HD Hierarchical Deterministic
HSM *Hardware Security Modules*
HTTP *HyperTextTransferProtocol*
IIC *Industrial Internet Consortium*
IIoT *Industrial Internet of Things*
IoT *Internet of Things*
IEEE *Institute of Electrical and Electronics Engineers*
IAM *Identity and Access Management*
IP *Internet Protocol*
ISO *International Organization for Standardization*
ISM *Industrial Scientific Medical*
ITU *International Telecommunication Union*
LabP2D Laboratório de Processamento Paralelo Distribuído
LoRa *Long Range*
LTE *Long Term Evolution*

LPWAN *Low Power Wide Area Network*

MAC *Medium Access Control*

MIC *Message Integrity Code*

MQTT *Message Queuing Telemetry Transport*

MBA Mecanismo de Busca Acadêmica

MIC *Message Integrity Code*

MHz Mega Hertz

NB-IoT *Narrowband IoT*

NwkKey *Network Encryption Key*

NIST *National Institute of Standards and Technology*

OSI *Open Systems Interconnection*

OTAA *Over-the-Air Activation*

PPGCAP Programa de Pós-Graduação em Computação Aplicada

TI Tecnologia da Informação

UDESC Universidade do Estado de Santa Catarina

UDP *User Datagram Protocol*

SUMÁRIO

1 INTRODUÇÃO

A massiva difusão de dispositivos conectados à Internet culminaram na concretização do conceito da *Internet of Things* (IoT). Consequentemente a adoção dos dispositivos inteligentes no contexto industrial resultou na chamada Internet Industrial, ou *Industrial Internet of Things* (IIoT). A admissão destes dispositivos na indústria é motivada pelo ganho de produtividade gerado pelas tecnologias nestas atividades. A incorporação de dispositivos inteligentes em cenários complexos e críticos como a indústria tem uma série de requisitos de segurança, área de aplicação e outros (BATAGLINI et al., 2021). O controle de acesso e o gerenciamento de identidades é uma das principais formas de garantir a autenticidade de dispositivos em todos os contextos computacionais. Neste contexto crítico, aumenta a necessidade de estar aderente aos requisitos e políticas de segurança.

Os arranjos das redes IIoT devem obedecer padrões gerais da tecnologia da informação de entidades como *National Institute of Standards and Technology* (NIST), *International Organization for Standardization* (ISO) e *Institute of Electrical and Electronics Engineers* (IEEE). Além disso, devem observar padrões de entidades específicas para regulação da IIoT como o *Industrial Internet Consortium* (IIC).

O IIC define padrões para arquitetura da IIoT, análise de dados, conectividade e segurança para a IIoT. Os padrões designados pelo IIC estão em conformidade com os padrões referenciais das entidades NIST, ISO e IEEE. Estes padrões incluem recomendações de segurança relativas ao controle de acesso e especificamente do ciclo de vida de credenciais para IIoT.

Em termos das tecnologias de comunicação e os meios de transmissão utilizados na adoção de ambientes de IIoT podem significar riscos dependendo da escolha, e.g., as redes Low Power Wide Area Network (LPWAN) LoraWan, implementadas com servidores ChirpStack não necessariamente foram desenhadas para atender aos padrões de entidades reguladoras do setor industrial (STELLIOS; KOTZANIKOLAOU; PSARAKIS, 2019). Logo, isto significa que estas redes não atendem padrões como os do IIC em relação ao gerenciamento de identidades e ciclo de vida de credenciais. Este trabalho se justifica na análise do ciclo de vida de credenciais no contexto das redes *Low Power Wide Area Network* (LPWAN) LoraWan privadas providas por ChirpStack, no que se refere ao controle de acesso e gerenciamento de credenciais.

Para realização de tal análise, além de fazer um apanhado dos padrões envolvidos, foram necessários acercamentos sobre as tecnologias do ambiente. Uma primeira etapa da definição do ambiente foi analisar as implementações do protocolo

LoraWan. Em primeiro lugar foram analisadas as versões do protocolo LoraWan. A versão atual estável do protocolo é a versão 1.1. Porém, existem muitos dispositivos que usam a versão 1.0 em uso atualmente (LOUKIL et al., 2022). Mesmo com avanços em relação a características de segurança, não houveram avanços no sentido do gerenciamento de credenciais. As diferenças em relação a segurança são exploradas no trabalho e tendo em vista a melhor adequação de requisitos de segurança foi escolhida a versão 1.1 do protocolo LoraWan para o cenário deste estudo.

Tendo definida a versão do protocolo LoraWan, o trabalho analisa as as ferramentas mais relevantes da atualidade no quesito da implementação dos servidores LoraWan. Para tanto, foram analisadas ferramentas disponíveis no mercado e a ferramenta que obteve resultados positivos em relação a atualização, suporte da comunidade e código aberto foi o ChirpStack. Esta ferramenta oferece uma interface de gerenciamento de *gateways* e dispositivos. O ChirpStack fornece uma API baseada em gRPC que pode ser usada para integrar ou estender o ChirpStack. Esta API foi usada como meio de realizar os testes envolvidos nos experimentos.

O problema principal deste trabalho exige uma análise que possa testar a adequação do cenário LoraWan com servidores providos com ChirpStack em cenários industriais com o ciclo de vida recomendado pelas entidades padronizadoras da indústria. Neste contexto, são utilizados os padrões do IIC, em conformidade com os principais padrões de segurança da informação e de IoT de forma geral. O objetivo é analisar este ciclo de vida, identificando conformidades e problemas relacionados as diversas identidades/credenciais existentes em um sistema IIoT LoraWan ChirpStack. A análise se concentra em primeiro lugar a entender as relações entre os padrões de ciclo de vida de identidades e o protocolo LoraWan. Em seguida, se dedica a analisar por meio de critérios objetivos a adequação do cenário aos padrões.

A execução do trabalho se dá por meio de uma composição de pesquisa e revisão bibliográfica aliada a pesquisa de aplicada. As principais contribuições desta pesquisa são:

- Identificação do ciclo de vida de identidades aplicado a LoraWan 1.1; e
- Identificação e análise do ciclo de vida de identidades no LoraWan 1.1 quando empregada a soluções de código aberto ChirpStack.

As contribuições, conforme listadas são relativas principalmente a análise do ciclo de vida de credenciais. Em primeiro lugar fazendo uma análise do ciclo de vida aplicado a LoraWan 1.1. Contribuição relevante pois tal intersecção não é explorada conforme demonstra a revisão bibliográfica. Ademais, o presente trabalho contribui na análise do ciclo de vida no cenário estudado de acordo com os padrões industriais.

O processo de fundamentação, proposta, testes e contribuições são o fio condutor desta dissertação. Para tanto o texto está organizado como segue. O Capítulo 2 traz a fundamentação dos principais conceitos em IIoT e LoraWan. Traz o levantamento conceitual-teórico no capítulo ainda é realizada a definição do problema. O mesmo capítulo, ainda abarca uma pesquisa bibliográfica sistemática na qual a resolvibilidade prévia do problema não se mostrou verdadeira. No Capítulo 3 os detalhes da proposta de análise são descritos. Os critérios de análise, a arquitetura e demais detalhes do ambiente são apresentados. Além disto, abarca o planejamento dos testes. A proposta tem o propósito de alinhar os experimentos realizados com o objetivo, de forma que sejam desenhados testes que cumpram o objetivo de analisar o ciclo de vida de credenciais no ambiente. No Capítulo 4 são realizados os experimentos *de facto*. Neste capítulo são demonstradas a inicialização dos experimentos com as variáveis e dados necessários ao ambiente para a realização dos testes. O capítulo conta com uma organização na qual os tópicos representam os cenários de testes desenvolvidos no Capítulo 3. Além disto, são apresentadas as análises e comparações dos resultados obtidos nos experimentos.

Para realização do trabalho foi empregado o método de pesquisa aplicada. Desta forma, o trabalho sintetiza os conceitos obtidos na fundamentação com a proposta realizada no ambiente com os testes e experimentos.

2 FUNDAMENTAÇÃO

O conceito de *Internet of Things* (IoT) e a sua evolução para o arranjo industrial, denominado *Industrial Internet of Things* (IIoT), representam uma oportunidade de ganho de produtividade, segurança e eficiência nas operações de indústrias de diversas verticais. Entretanto, a adoção destas tecnologias depende de uma arquitetura de referência que possa trazer segurança e organização para a implementação desta tecnologia em conjunto com a tecnologia operacional presente nas indústrias. Outras preocupações pertinentes na adoção dessas tecnologias dizem respeito às tecnologias adotadas de acordo com os requisitos do contexto e a partir das suas implicações com a segurança. Dentre estas implicações de segurança, existem cuidados em se tratando do controle de acesso aos dispositivos presentes na borda do sistema IIoT. No gerenciamento do controle de acesso, um dos pontos cruciais é o ciclo de vida de credenciais que pode evitar ataques conhecidos e prover mais segurança ao sistema como um todo.

No presente capítulo, realiza-se uma fundamentação dos principais conceitos envolvidos no trabalho relacionando estes pontos de preocupação na adoção da IIoT. Para tanto, é realizado um estudo para a definição do conceito de IIoT com base na literatura disponível e a arquitetura referencial é apresentada com base em entidades padronizadoras. Além disto, este capítulo introduz os conceitos inerentes a garantia de requisitos gerais de segurança específicos a uma rede IIoT, realizando uma pesquisa dos principais ataques direcionados a este cenário. Em seu teor, o capítulo trata ainda do gerenciamento de identidades em IIoT baseados nas normativas das entidades reguladoras. O capítulo ainda elenca as tecnologias de comunicação disponíveis para compor a camada de borda desta aplicação, fazendo uma análise comparativa entre as mesmas. As tecnologias são comparadas de acordo com o cenário desenvolvido a partir dos requisitos gerais desenhados.

Ademais é feita uma análise da tecnologia que pode cumprir os requisitos levantados, neste caso, vem a ser a rede do tipo *Low Power Wide Area Network* (LPWAN) LoraWan. Ainda são descritas definições do protocolo LoRa, a sua arquitetura de camadas e as suas classes de dispositivos. Além de levantados os conceitos relativos ao protocolo *wide area* advindo do LoRa, o qual é denominado LoraWan. Ainda é descrita a arquitetura de rede protocolo de acordo com os idealizadores da mesma. Ainda no tópico relacionado ao LoraWan, são discutidos aspectos gerais de segurança no protocolo e aspectos do gerenciamento de credenciais e identidades.

A partir dos conceitos e do cenário levantado é feita uma uma definição do

problema circundado por este trabalho. Para tanto são levantados quais os atores envolvidos no problema, um levantamento das consequências relacionadas ao problema bem como um levantamento das limitações das abordagens tradicionais. O capítulo ainda realiza uma pesquisa de trabalhos relacionados sendo esta, uma busca de trabalhos relativos a IIoT, LoraWan, segurança e o ciclo de vida de credenciais. Esta busca dá-se por meio de uma designação de palavras-chave, uma estruturação da forma de seleção, o levantamento de fontes, a análise dos trabalhos com critérios de inclusão e exclusão levando a execução da busca. Os trabalhos ainda são categorizados por meio da sua elegibilidade, e os trabalhos mais pertinentes são agregados e discutidos individualmente. Este levantamento demonstra a lacuna que indica a não resolução na literatura, do problema proposto neste trabalho.

2.1 *INDUSTRIAL INTERNET OF THINGS* (IIOT)

Em 1991, Mark Weiser cunhou o termo computação ubíqua (WEISER,) como um nível dos três que caracterizam a evolução dos computadores. No primeiro nível, a computação compartilhada, com *mainframes* ou servidores compartilhados por vários usuários. No segundo nível, a computação pessoal, na qual existe, em média, um usuário por dispositivo. No terceiro nível, a computação ubíqua, em que os dispositivos passam a ter tamanho reduzido e cada usuário utiliza vários dispositivos presentes em diversos objetos do cotidiano de maneira onipresente (PIMENTEL; FUCKS, 2013).

Outra designação para a área do conhecimento que trata da difusão dos dispositivos conectados para além dos servidores e *desktops* é denominada Internet das coisas ou IoT. O conceito de IoT evoluiu e ganhou uma versão para a sua aplicação em ambientes industriais, a IIoT. Existem diversas definições que caracterizam a IIoT, cuja primeira articulação foi realizada pela General Eletrics (GE) (LEBER, 2012). A GE introduziu a definição de Internet industrial, que inclui dois componentes principais: (i) conexão de sensores e atuadores de máquinas industriais ao processamento local e à Internet e (ii) a conexão progressiva com outras redes industriais importantes que podem gerar valor de forma independente (FLOYER, 2013). Percebe-se que o conceito de Internet Industrial definido pela GE é similar à definição atual de IIoT.

A IIoT também pode ser definida como um sistema que conecta e integra sistemas de controle industrial com sistemas empresariais, e também como um conjunto de processos de negócios e análises, usando sistemas de controle industrial que contêm sensores e atuadores (LIN et al., 2017), os quais também são normalmente de considerável complexidade. O conceito pode ainda, de maneira geral, ser definido como o uso de tecnologias de IoT na manufatura (ABERLE, 2015), e pode ser empregado como uma abreviação para as aplicações industriais da IoT (KORSTEN et al., 2015).

A Internet Industrial também é comumente associada ao conceito de Indústria 4.0, no qual a utilização de IoT e outras tecnologias faz emergir a quarta revolução industrial. O conceito de Indústria 4.0 foi consolidado a partir do ano de 2011 numa iniciativa do governo alemão em parceira com as universidades e a indústria. O objetivo era de tornar o país um pioneiro na produção e utilização de tecnologia de informação industrial (PEREIRA; SIMONETTO, 2018).

Observa-se que existem diversas definições existentes para a IIoT. Neste trabalho, interpreta-se a definição de IIoT, baseando-se em (LIN et al., 2017), ou seja como um sistema no qual existe a conexão e a integração da infraestrutura industrial e de dispositivos finais, com uma infraestrutura de rede e controle para possibilitar a análise e visualização de dados, além de controle centralizado pelos sistemas empresariais de alto nível. Em consonância com esse conceito, faz-se necessário compreender a dinâmica entre os componentes da arquitetura de um sistema de IIoT.

2.1.1 Arquitetura IIoT

Desde que os objetos inteligentes começaram a ter certa popularidade em contextos industriais, surgiram iniciativas na tentativa de padronizar os ambientes de IoT e IIoT. Segundo (PAL; PURUSHOTHAMAN, 2016), a padronização fez-se importante em contextos e dispositivos inteligentes por razões como interoperabilidade e segurança. A interoperabilidade refere-se à possibilidade de comunicação de dispositivos de diferentes fabricantes com protocolos comuns. A segurança refere-se a manter os requisitos de confidencialidade, integridade, disponibilidade e autenticidade (SAMONAS; COSS, 2014).

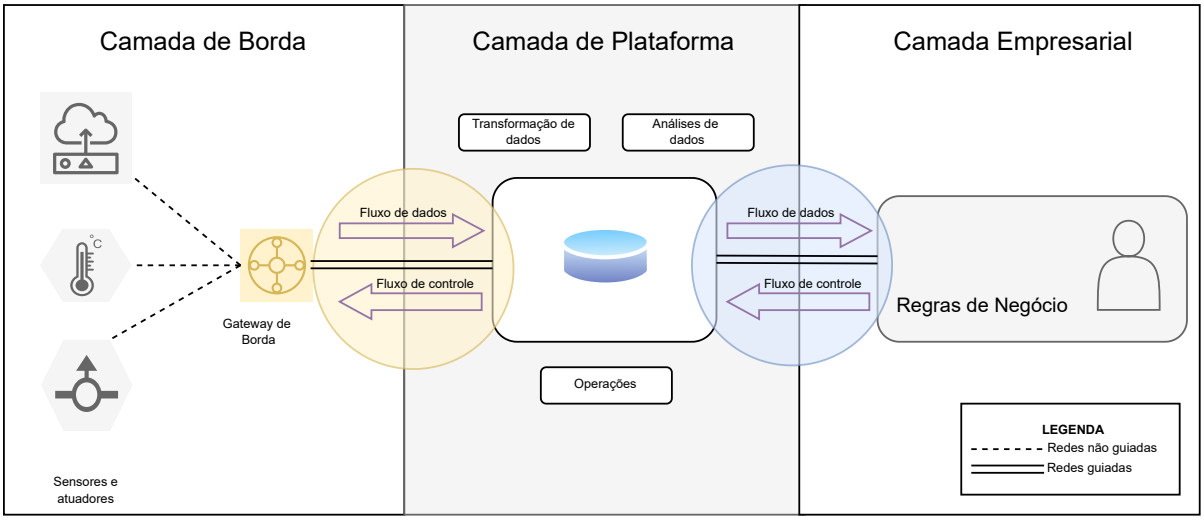
Apesar da carência de um órgão único para criar padrões para a IIoT, entidades reguladoras como o *International Telecommunication Union* (ITU) e *International Organization for Standardization* (ISO) possuem publicações que visam contribuir com a padronização de ambientes com dispositivos inteligentes como IoT e IIoT (WAGLE; PECERO, 2019; ZENNARO, 2017). Dentre estas, destaca-se a ISO 30141:2018, intitulada *"Internet of Things (IoT) — Reference Architecture"* que fornece uma arquitetura de referência com práticas recomendadas. O documento faz um apanhado das características mais importantes do cenário de IoT, faz um modelo conceitual cenário do contexto e oferece um modelo em cinco visualizações de arquitetura de diferentes perspectivas (SANTOS et al., 2020). Por outro lado, a norma Y.4460 do ITU *"Architectural reference models of devices for Internet of things applications"* é uma recomendação que define uma classificação dos dispositivos de IoT que baseia-se em características como a capacidade de processamento e conectividade destes. A partir desta classificação, são definidos os modelos de referência com os requisitos para as arquiteturas de software de cada uma das classes.

Estas recomendações têm importante papel no desenvolvimento no estabelecimento das linhas gerais dos contextos de IIoT porém existem ainda iniciativas conectadas com o mercado e a indústria. A inciativa que mais destaca-se neste sentido é o *Industrial Internet Consortium* (IIC), entidade que congrega grandes empresas do meio industrial, setores da academia além de entidades reguladoras (LIN et al., 2017).

O IIC em 2017 publicou o *"Volume G1: Reference Architecture"* documento no qual apresenta uma arquitetura referencial, em conformidade com os padrões ISO e ITU. O documento faz um compêndio de conceitos envolvidos com o conceito de IIoT, apresenta a estrutura da arquitetura de camadas para a IIoT, além de apresentar a arquitetura referencial sob o ponto de vista dos negócios, do ponto de vista funcional e do ponto de vista de implementação (LIN et al., 2017).

Segundo (BADER; MALESHKOVA; LOHMANN, 2019), que no seu trabalho fizeram uma classificação de propostas de arquitetura referencial de IIoT. A proposta de arquitetura do IIC tem o melhor desempenho no critério de relevância na área de pesquisas acadêmicas de acordo com ranking elaborado pelos autores. Esta posição é atingida devido ao fato de ser a arquitetura referencial com maior número de referências em trabalhos de pesquisa. O trabalho classifica as propostas de arquitetura referencial também por interesse geral, por meio de dados de mecanismos de busca. Portanto, esta será a arquitetura referencial utilizada neste trabalho. O IIC, em (LIN et al., 2017) apresenta como proposta um modelo de arquitetura referencial de IIoT, projetado com uma estruturação em três camadas. (Figura 1).

Figura 1 – Arquitetura IIoT de três camadas.



Adaptado de: (LIN et al., 2017)

Uma breve explicação das camadas listadas na Figura 1 (LIN et al., 2017):

1. Camada de Borda: formada pelos dispositivos físicos, sensores, atuadores e controladores que coletam dados ou executam funções específicas, dispositivos esses marcados pela pouca disponibilidade de recursos como memória, processamento e armazenamento;

2. Camada de Plataforma: camada formada pela infraestrutura de comunicação e controle dos dispositivos na rede. Recebe, processa e encaminha comandos de controle da camada empresarial à camada de borda; e

3. Camada Empresarial: recebe dados advindos da borda e da camada de plataforma, além de emitir comandos de controle à camada da plataforma e à camada de borda.

Na Figura 1, a Camada de Borda coleta dados dos sensores e atuadores, usando redes majoritariamente sem fio. Esses dispositivos costumam ter pouco poder de processamento e memória, de forma que as operações de agregação de dados são realizadas pelos *gateways* de borda (LIN et al., 2017; SILVA et al., 2019). O *gateway* pode ainda fazer a transformação dos protocolos de comunicação. Os sensores e atuadores podem usar protocolos de comunicação como *Long Range* (LoRa), SigFox, *Narrowband IoT* (NB-IoT), e os *gateways* podem realizar a transformação para protocolos *Internet Protocol* (IP) (KULIK; KIRICHEK, 2018). No que tange ainda à arquitetura referencial, a Camada de Plataforma recebe, processa e encaminha comandos de controle da Camada Empresarial à Camada de Borda, além de gerenciar processos e analisar fluxos de dados da Camada de Borda, bem como fornecer funções de gerenciamento a dispositivos e ativos.

A Camada de Plataforma oferece serviços não específicos do domínio, como consulta e análise de dados (LIN et al., 2017; RIBEIRO, 2021) A Camada de Borda (Figura 1) e a Camada de Plataforma são conectadas por meio de um *gateway*, o qual é responsável pela gestão dos dispositivos, pela agregação de dados e pelo encaminhamento de comandos da Camada de Plataforma e da Camada Empresarial. A Camada Empresarial é a camada de mais alto nível, e implementa aplicativos de visualização de dados, sistemas de suporte à decisão, além de fornecer interfaces a usuários finais, incluindo especialistas em operação. A Camada Empresarial também recebe fluxos de dados da camada de borda e plataforma e emite comandos de controle à Camada de Plataforma e à Camada de Borda (LIN et al., 2017).

Nessa arquitetura em camadas de (LIN et al., 2017), os fluxos de dados e controle, por meio dos *gateways*, formam a relação entre as camadas. Essa conectividade, que faz uso de fluxos bidirecionais, traz para a arquitetura um nível maior de

complexidade. No contexto de IIoT, geralmente existe uma quantidade considerável de elementos na rede (HEJAZI et al., 2018), o que representa um ponto adicional de complexidade, principalmente na Camada de Borda, potencialmente gerando desafios adicionais no que se trata do gerenciamento dessa rede. Os dispositivos finais, os quais se encontram na Camada de Borda, principalmente no que tange cenários de sensoriamento, podem-se dar em elevada quantidade. A Camada de Borda, além disso, geralmente tem por característica a heterogeneidade desses dispositivos (HEJAZI et al., 2018), os quais, de acordo com a arquitetura referencial apresentada, comunicam-se com as outras camadas de rede por meio do *gateway* que, por sua vez, age como um concentrador do fluxo de dados e de controle desses dispositivos heterogêneos. O fluxo de informações dá-se a partir da geração dos dados nos dispositivos da Camada de Borda, passando pela camada de plataforma e indo até a Camada Empresarial, na qual serão disponibilizados em aplicativos de visualização de dados. Eventualmente, a depender dos dados ou do acionamento manual, podem ser disparadas ações por meio dos sensores e atuadores. Por outro lado, o fluxo de controle dá-se no sentido inverso das informações. A partir da visualização dos dados, ou de gatilhos programados, são disparados comandos aos atuadores presentes na Camada de Borda, novamente tendo a Camada de Plataforma como infraestrutura (LIN et al., 2017).

Na IIoT, os dispositivos finais podem ser classificados, de maneira geral, entre dispositivos inteligentes e infraestrutura operacional. Os dispositivos inteligentes são dispositivos capazes de servir como um recurso numa rede, ou seja, são objetos conectados em rede servindo como um componente IoT e IIoT (FRANÇA et al., 2011).

Quando trata-se de IIoT, os dispositivos finais da Camada de Borda são intrinsecamente ligados à atividade industrial. A arquitetura de rede deve ter requisitos de modo a oferecer o pleno funcionamento de serviços de infraestrutura, como controle de movimento, robôs móveis, funções de segurança e monitoramento de processos, conforme ilustrado na Tabela 1.

Tabela 1 – Principais requisitos de desempenho em serviços de IIoT / automação industrial.

Caso de Uso (alto nível)		**Disponibilidade**	**Ciclos de tempo**	**Tamanho típico do payload**	**Número de dispositivos**	**Área típica de serviço**
Controle de movimento	Impressora industrial	>99,9999%	<2ms	20 bytes	>100	100mx100mx30m
	Máquina-ferramenta	>99,9999%	<0,5ms	50 bytes	$\sim$20	15mx15mx3m
	Máquina de embalagem	>99,9999%	<1ms	40 bytes	$\sim$50	10mx5mx3m
Robôs móveis	Controle de movimento cooperativo	>99,9999%	1ms	40-250 bytes	100	<1 km^2
	Controle remoto operado por vídeo	>99,9999%	10-100ms	15-150 bytes	100	<1 km^2
Painéis de controles móveis e com funções de segurança	Robôs de montagem ou fresadoras	>99,9999%	4-8ms	40-250 bytes	4	10m x 10m
	Guindastes móveis	>99,9999%	12ms	40-250 bytes	2	4m x 60m
Automação / Monitoramento de processos		>99,99%	>50ms	vários	10.000 dispositivos por km^2	

Adaptado de (BROWN et al., 2018)

A configuração de uma rede IIoT nas suas camadas pode ser definida a partir de requisitos dos seus dispositivos finais, conforme a Tabela 1. Requisitos tais como alta disponibilidade, ciclos de tempo e massividade podem resultar em diversos outros requisitos, os quais exigem, *e.g.*, a comunicação por algum meio físico de alta velocidade e uma infraestrutura que permita a comunicação em curtos intervalos de tempo e com disponibilidade para suportar muitos dispositivos.

Este trabalho tem foco nos requisitos de automação e monitoramento de processos. O que implica numa grande densidade de dispositivos, vários tipos de *payloads*, com ciclos de tempo e disponibilidade própria. Com base nestes requisitos, itens da arquitetura referencial, como o meio de transmissão e as tecnologias de comunicação devem ser especificados.

Partindo da descrição dos requisitos dos dispositivos de IIoT, existem diversas tecnologias que podem formar a arquitetura da rede dentro dessa referência em camadas. Essas variações de tecnologias para atendimento dos requisitos podem trazer desafios adicionais de complexidade, custo e segurança.

Em IIoT, mesmo quando essa arquitetura referencial é seguida, existem ainda diversos riscos envolvidos no que se refere à segurança dos sistemas e à criticidade do ambiente, podendo envolver maquinário complexo que oferece risco ao mundo físico (BOYE; KEARNEY; JOSEPHS, 2018). À medida que a implementação da IIoT populariza-se, além das preocupações com a arquitetura, crescem também as preocupações com as questões de segurança e comunicação (PANCHAL; KHADSE; MAHALLE, 2018a; NAKAMURA; RIBEIRO, 2018; YU; GUO, 2019).

Um ataque cibernético numa infraestrutura crítica pode ter consequências graves, fazendo-se necessário investigar a relação da arquitetura IIoT e a segurança desses contextos. Existem, na literatura, diversos casos de ataques a organizações que fazem uso dessa infraestrutura (SULLIVAN; KAMENSKY, 2017; HASSANZADEH; MODI; MULCHANDANI, 2015; KARNOUSKOS, 2011), o que reforça a preocupação com esses contextos computacionais.

2.1.2 Segurança em IIoT

Os sistemas de IIoT são caracterizados por conectar e integrar diferentes tipos de sistemas de controle e sensores com sistemas corporativos e de controle na camada empresarial, e diferem-se dos sistemas de controle industrial tradicionais por estarem conectados, o que aumenta a sua diversidade e a sua escala.

Tradicionalmente, a segurança em sistemas industriais dependia da separação física e do isolamento da rede de componentes vulneráveis e das regras de acesso para sistemas de controle críticos. Os criadores desses sistemas não conside-

ravam que estes podem ser expostos publicamente/Internet, tornando-os acessíveis tanto a usuários legítimos como também a usuários maliciosos e desautorizados (KOBARA, 2016).

Em contrapartida, na IIoT o poder de computação cada vez mais acessível, a conectividade onipresente abriu as portas para a convergência de sistemas de controle, sistemas de negócios e a Internet. Esta convergência teve a sua origem especialmente para o monitoramento e o gerenciamento remoto de sistemas, mas que também expandiu-se para atividades mais complexas como mineração e análise de dados. Estas atividades tem como fim de obter métricas de desempenho para prever falhas, otimizar frotas e realizar atualizações remotas de *software*. A IIoT também aumentou a produtividade, a eficiência e o desempenho dos processos operacionais existentes e permitiu a criação de novas formas de alavancar os dados das operações, entregando valor de negócios agora e no futuro (JAIDKA; SHARMA; SINGH, 2020).

Esses ganhos, porém, trazem riscos. Um ataque bem sucedido pode ter consequências tão grandes quanto um acidente industrial, possuindo o potencial de dano material e ambiental, de ferimentos ou até a perda de vidas humanas. Ademais, pode incorrer na divulgação de dados confidenciais, interrupção de operações e destruição de sistemas, além de dano à marca, à sua reputação, e consequentes danos econômicos. Apesar da recente convergência de parte das indústrias, já existem diversos registros de ataques nesses ambientes (Tabela 2).

Tabela 2 – Ataques em redes IIoT.

	Camada afetada	**Tipo do ataque**	**Tipo de rede**	**Tipo de aplicação**
(INGHAM; MARCHANG; BHOWMIK, 2020)	Borda	*Man-in-the-Middle*, Interferência de sinal	Redes não guiadas (LoraWan)	Redes LPWAN para sensoriamento industrial
(BERLANDA, 2021)	Borda / Plataforma	*Sniffing* - Captura de tráfego	Redes não guiadas	Dispositivos com comunicação não criptografada em geral
(PANCHAL; KHADSE; MAHALLE, 2018b)	Borda / Plataforma	Injeção de pacotes no tráfego	Redes não guiadas	Dispositivos com comunicação não criptografada e não autenticada em geral
(JR et al., 2017)	Borda	Engenharia Reversa para roubo de credenciais e injeção de código malicioso	Dispositivos em rede em geral	Dispositivos finais com *firmware*/*software* não atualizado

Fonte: O autor.

Na Tabela 2, pode-se observar que a complexidade da arquitetura referencial desdobra-se em relação aos ataques, que podem afetar todas as camadas da arquitetura. Os ataques podem ainda ser de tipo passivo, com técnicas que permitem capturar o tráfego da rede. Os ataques ainda podem ser ativos, com técnicas que podem interferir no sinal e injetar pacotes maliciosos. Os ataques podem até mesmo culminar no roubo de credenciais, que, aliado à injeção de códigos maliciosos, pode resultar em ataques especialmente danosos e na interrupção das aplicações finais do sistema.

Tendo em vista essa percepção de risco, os sistemas de IIoT devem-se pautar em garantir requisitos de resiliência devido à criticidade de processos industriais, tais como os processos relacionados a setores ligados à energia elétrica e à saúde, para citar alguns exemplos, cujo tempo ocioso deve ser evitado por questões econômicas.

Já os requisitos ligados à segurança tradicionalmente utilizados na computação têm que ser garantidos no que se refere ao roubo de informações e segredos, e sequestro de dados, entre outros ataques (BREIVOLD; SANDSTRÖM, 2015).

De forma geral, a justificada demanda por segurança e confiabilidade passa por temas chave na garantia desses requisitos, para a qual confiabilidade, confidencialidade, integridade, disponibilidade e autenticidade são caraterísticas básicas. Essa demanda pode ser encarada a partir do ponto de vista do negócio fazendo-se a gestão de riscos em alto nível, ou seja, criando-se programas de conscientização e segurança na organização, métricas e indicadores de segurança, e ainda considerando-se essa demanda no orçamento (SISINNI et al., 2018). Por outro lado, a demanda por segurança pode ser encarada do ponto de vista funcional e de implementação. Esse ponto de vista tem objetivos de segurança concretos, como a proteção da conectividade e das comunicações, a proteção dos *endpoints* e o monitoramento e a análises de segurança, bem como a configuração e o gerenciamento de segurança (SCHRECKER et al., 2016).

Em tratando-se destes dois últimos em particular, no contexto de IIoT, existem desafios adicionais como o tratamento da segurança operacional em contraposição às políticas tradicionais de segurança. Uma das particularidades do cenário de IIoT é que a comunicação comumente se dá por meio dos dispositivos e os *gateways* e não por usuários e os *gateways*. Esta condição traz prerrogativas e requisitos diferentes em relação as políticas tradicionais. A comunicação entre dispositivos tem especificidades tal qual a autenticação por credenciais cujo a comunicação dispositivo - *gateway* é autenticada por meio de certificados, códigos e outras maneiras de identificar uma entidade na autenticação. Concomitantemente, em um cenário de comunicação dispositivo-*gateway* a política de gerenciamento e configuração em segurança ganham traços próprios. Estes traços envolvem aspectos dentre os mais diversos, como a proteção da comunicação, *backups*, a atualização de *softwares*, a proteção de portas e também o controle de acesso (SCHRECKER et al., 2016).

O modelo de camadas em (LIN et al., 2017) indica ainda que a segurança deve ser pensada de acordo com a arquitetura referencial de IIoT, em que há a incidência de ataques voltados à camada empresarial. Uma contramedida a esses ataques pode ser, por exemplo, o aumento do nível de segurança das aplicações de alto nível. Já noutra camada, a de plataforma, em que prevalece a infraestrutura de rede, deve-se pensar em medidas de aumento da segurança na infraestrutura de rede, tais como *switches*, servidores locais e *gateways*. Por fim, na Camada de Borda, as contramedidas tem como objetivo garantir a comunicação segura dos seus dispositivos.

Requisitos específicos, como o número de dispositivos, a sua localização e a sua capacidade de reconfiguração, entre outros, podem influenciar nas políticas de

segurança. No presente trabalho, analisa-se especificamente o tema relacionado à segurança de contextos IIoT na sua Camada de Borda para o controle de acesso de dispositivos e no gerenciamento do ciclo de vida de credenciais desses dispositivos, garantindo autenticidade no controle de acesso de dispositivos em sistema industrial. Exclui-se deste escopo os acessos humanos, com foco em dispositivos.

2.1.2.1 Gestão de identidade e acesso em IIoT

O *National Institute of Standards and Technology* (NIST) define Controle de Acesso (CA) como um processo de concessão ou negação de solicitações específicas para obterem-se e usarem-se informações e serviços de processamento. (WHITE; SJELIN, 2022). A instituição ainda define CA como o processo de conceder acesso a recursos de Tecnologia da Informação (TI) apenas a usuários, programas, processos ou entidades autorizadas por meio de alguma credencial válida (DEMPSEY; PILLITTERI; REGENSCHEID, 2021).

De forma geral, o controle de acesso de um sistema computacional ocorre com entidades distintas. Primeiramente, existem dois componentes básicos: a entidade que solicita o acesso e o recurso que essa entidade deseja acessar, sujeito e objeto respectivamente. Para tanto, o sistema de controle de acesso deve verificar a identidade da entidade solicitante a fim de que ela seja autêntica e que tenha privilégios para acessar dito recurso (LEGAT et al., 2018).

O processo de verificação da identidade pode ser feito de diversas maneiras. A identidade da entidade pode ser verificada com algo que a entidade conheça, como, por exemplo, por meio de códigos e senhas. A identidade também pode ser avaliada por meio de algo que a entidade possua, ou seja, um *token*, uma credencial, um certificado ou um identificador de forma geral.

Segundo o (SCHRECKER et al., 2016) a definição de credencial se refere a uma evidência que dá suporte a uma revindicação de identidade ou de um atributo. Esta evidência, em tese é usada mais de uma vez. Esta definição está de acordo com a padronização descrita na revisão de número 4009 do glossário do **CNSS!** (**CNSS!**).

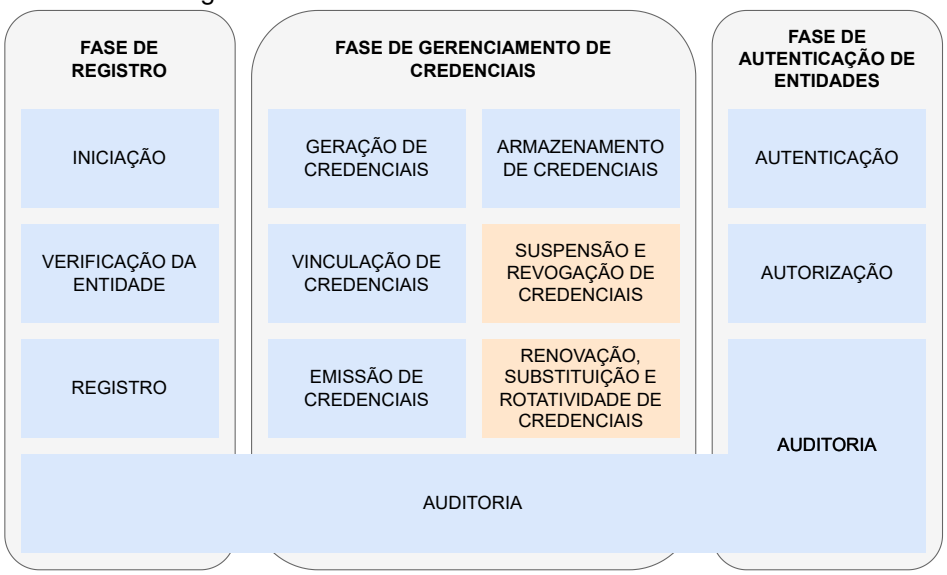
A identidade, além disso, pode ser mensurada por meio de algo que a entidade seja, ou, posto doutra forma, por meio da verificação de atributos. Por fim, esta pode ser verificada utilizando-se algo que esta faz, ou seja, esta pode ser identificada por um comportamento (LEGAT et al., 2018).

De forma geral, existe uma organização para gerir esse processo. Essa gestão dos processos no controle de acesso é denominada *Identity and Access Management* (IAM) ou ainda Gestão de Identidade e Acesso. A IAM tem a função de estabelecer o relacionamento entre um usuário ou dispositivo com os recursos que deve acessar

para exercer a sua função de trabalho (NIST, 2014). Para gerenciar o controle de acesso, o IAM disponibiliza identidades que são associadas a credenciais que, por sua vez, são utilizadas para controlar o acesso a recursos protegidos. Essas credenciais são o atributo de identificação dos usuários e dispositivos no controle de acesso.

Em (SCHRECKER et al., 2016), o gerenciamento de identidades tem três fases: a primeira, em que a entidade é verificada e registrada; a segunda, que é a fase de gerenciamento das credenciais associadas a essas entidades; e a terceira, a fase de autenticação, em que se dá a verificação da identidade propriamente dita, bem como a autorização da entidade para acessar os recursos solicitados. Esse processo é observado na Figura 2, com as fases e as suas etapas, além das etapas que perpassam todo o processo.

Figura 2 – Gerenciamento de identidades em IIoT.



Adaptado de: (SCHRECKER et al., 2016)

Conforme observa-se na Figura 2, a fase de registro garante que a entidade apropriada receba o material de identidade apropriado, ou seja, fornece a base para que se estabeleça a confiança numa entidade. Nesta fase é implementado um mecanismo para garantir que a entidade seja a correta antes que se emitam as credenciais. Em cenários em que exista uma quantidade de entidades em grande escala, uma abordagem automatizada é recomendada.

Existem três etapas para a fase de registro: a iniciação, a verificação da entidade e o registro. A iniciação declara o desejo de colocar-se a entidade sob gestão e obterem-se uma identidade e credenciais. A verificação da entidade envolve provar que ela é a entidade à qual a identidade deve ser criada e emitida. Na fase de registro, a entidade está ou pronta para ter as suas credenciais criadas e entregues ou para que a própria entidade gere as credenciais. Processos de validação após essa etapa são sempre indicados (SCHRECKER et al., 2016).

A fase de gerenciamento de credenciais é composta por seis etapas divididas em duas categorias, conforme a Figura 2. A primeira categoria compreende as etapas necessárias para gerar credenciais, vinculá-las a uma entidade e emiti-las à entidade à qual a credencial deve ser emitida. A segunda categoria compreende as etapas para armazenar credenciais e revogá-las, bem como estender a vida útil das credenciais. Após isso, espera-se que a entidade esteja num estado no qual as credenciais estão em vigor e prontas para uso. Em (SCHRECKER et al., 2016), a geração de credenciais inclui todas as etapas necessárias para criar a própria credencial ou para habilitar uma credencial. Em seguida, durante a vinculação da credencial, ela, ou o meio para criá-la, é associada à identidade atribuída à entidade. Finalmente, durante a sua emissão, a credencial, ou o meio ou diretriz para criá-la, é entregue à entidade usando um processo seguro e auditável. O processo específico depende da política organizacional para o ambiente e da implementação técnica.

A segunda categoria de etapas aborda o seu uso diário. A etapa de armazenamento de credenciais é implementada de acordo com a política da companhia com base nas autorizações que essa entidade possui. Quanto maior o nível de autorização, mais rigorosos devem ser os requisitos de armazenamento.

As etapas que concernem à suspensão e revogação de credenciais são essenciais para a sua remoção adequada do serviço. O processo de suspensão configura-se quando uma credencial é identificada para isso e é temporariamente impedida de ser usada para autenticação, o que se aplica a qualquer credencial ou processo de geração suspeito de comprometimento potencial. Se o comprometimento for provável para a credencial, esta deve ser revogada.

A revogação de credenciais em sistemas IIoT também pode ser acionada em caso de expiração ou como parte de um processo de rotação das credenciais. Este processo consistem em desassociar a credencial do dispositivo. Em alguns casos, uma credencial mais recente substitui a revogada. Em cenários menos críticos, pode-se considerar reutilizar as credenciais para prolongar a vida útil. Dentro desses termos, os processos dão-se de acordo com a política de rotação e desuso de credenciais da organização. Ademais, num cenário seguro, todas as operações de gerenciamento de credenciais devem ser rastreadas para fins de auditoria.

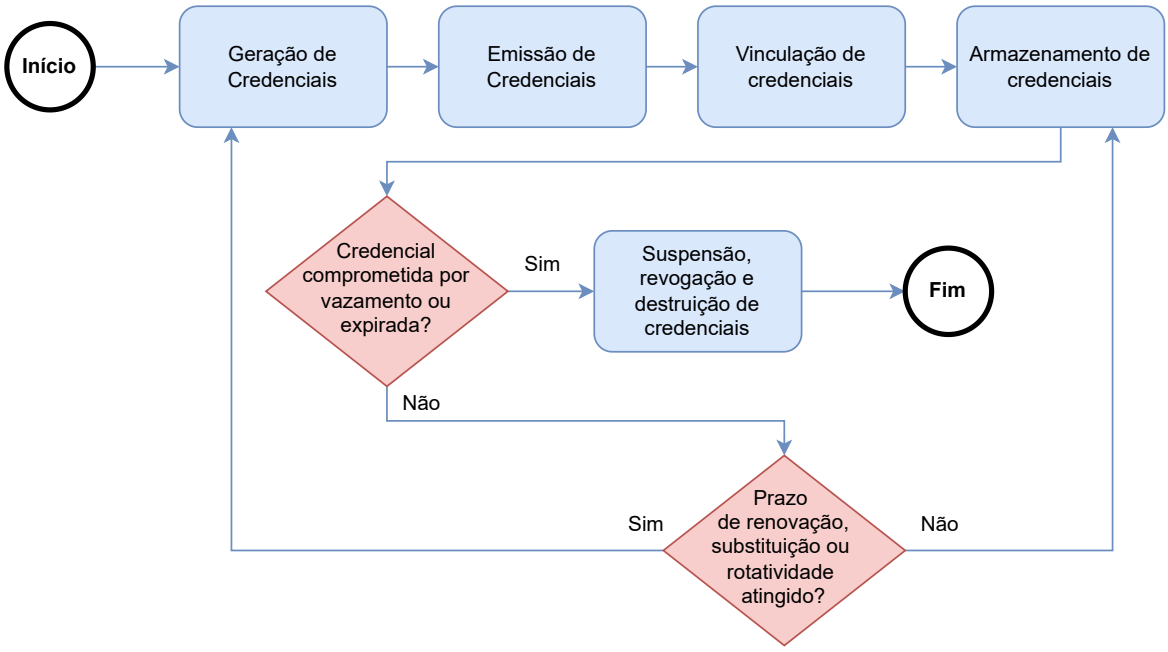
Se esse processo não for implementado e seguido corretamente, os resultados da autenticação do serviço podem não produzir o nível de confiança adequado. Tendo em vista requisitos próprios e o predomínio de autenticação de entidade não humanas, na sua maioria dispositivos, (SCHRECKER et al., 2016) defende que a aplicação de uma perspectiva de IIoT às recomendações de gerenciamento de identidade existentes gera uma variante do processo de ciclo de vida, o que torna necessário avaliar-se um ciclo de vida de identidades próprio de IIoT.

2.1.2.2 Ciclo de vida de identidades em IIoT

O gerenciamento das identidades e credenciais numa organização é fundamental para a garantia dos requisitos de segurança, o qual, além de proteger dados importantes, exige um sistema de controle de acesso. O gerenciamento adequado de credenciais e o uso da técnica de expiração de identidades podem evitar ataques utilizando credenciais indevidamente expostas anteriormente. Além disso, a rotatividade de identidades pode ao mesmo tempo prolongar o seu uso e evitar ataques que tenham como premissa a identidade antes da sua renovação.

O requisito de massividade de dispositivos que pode existir num contexto IIoT exige, segundo (SCHRECKER et al., 2016), uma forma organizada e ordenada de gerenciar o ciclo de vida de identidades num contexto IIoT. Na Figura 3, esse ciclo de vida é ilustrado por meio de um fluxograma.

Figura 3 – Ciclo de vida de identidades em IIoT.



Adaptado de: (SCHRECKER et al., 2016).

Na Figura 3, observa-se o ciclo de vida de identidades de acordo com a recomendação contida em (SCHRECKER et al., 2016). O ciclo começa na geração de credenciais, que cria credenciais novas para que sejam atribuídas a uma entidade. Já a emissão de credenciais é o processo de emanar a credencial ao sistema em geral a fim de que a entidade possa autenticar-se. Em sequência, o processo de vinculação associa a chave gerada e emitida à entidade correta para receber a credencial. Feitos esses processos, a credencial pode ser armazenada para uso.

No decorrer do processo de utilização cotidiana das identidades digitais, as credenciais podem ser comprometidas por vazamentos ou sofrer expiração pelo seu

tempo de vida útil. Nesses casos, é acionada a etapa de suspensão e revogação da credencial, que incorre no fim do seu ciclo de vida.

Conforme exemplificado na Figura 3, se a credencial não está comprometida ou expirada, esta deve ser verificada de acordo com o seu prazo de renovação, substituição e rotatividade. Segundo (SCHRECKER et al., 2016), a organização deve ter uma política de renovação e rotatividade de credenciais, pois isso minimiza as chances de ataque com credenciais vazadas.

Essa prática de implementar um ciclo de vida para as identidades e credenciais é um importante item para a definição dos papéis de uma determinada entidade dentro das propriedades de segurança essenciais para a arquitetura referencial de IIoT. Além disso, (LIN et al., 2017) define que a verificação de credenciais é um processo essencial na arquitetura no que se refere ao fluxo seguro de dados da Camada de Borda e da Camada Empresarial.

Esta pesquisa analisa o ciclo de vida de identidades em cenários de IIoT, conforme a recomendação da entidade padronizadora do IIC, elabora questões acerca do controle de acesso relacionado à arquitetura referencial apresentada na Seção 2.1.1, e considera o cenário de monitoramento de processos e dos dispositivos de IoT presentes na Camada de Borda da arquitetura referencial. Nesta interseção, elaboram-se os questionamentos acerca das melhores práticas de gerenciamento do ciclo de vida de credenciais dos dispositivos de borda na arquitetura referencial de IIoT. Para tanto, faz-se necessário o estudo do meio de transmissão e das tecnologias de comunicação que são utilizadas na implementação da arquitetura.

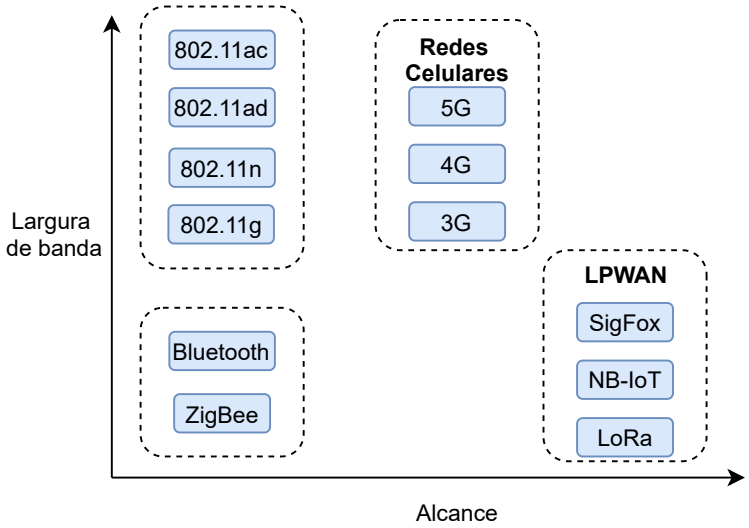
2.2 TECNOLOGIAS DE COMUNICAÇÃO EM IIOT

Em cenários IIoT, um aspecto relevante para a implementação de um projeto é a escolha das tecnologias de comunicação e o meio de transmissão, especialmente na borda da arquitetura referencial. A depender dos requisitos dos dispositivos conectados, é necessário adotar meios de transmissão de redes guiadas ou não guiadas da camada física (ZIMMERMANN, 1980).

Os meios de transmissão guiados, ou seja, aqueles que requerem um condutor físico (par trançado, coaxial, entre outros) (FOROUZAN; COOMBS; FEGAN, 2001) são adequados para a infraestrutura operacional de controle de movimento e robôs estacionários. Porém, as aplicações de robôs móveis e automação e monitoramento de processos exigem mobilidade e quantidade considerável de dispositivos com uma área típica de serviço acima de 10 quilômetros quadrados (Km^2) (BROWN et al., 2018), o que são requisitos melhor atendidos pelos meios de transmissão não guiados, tais como redes sem fio. Existem diversas soluções entre as tecnologias de comunicação

com meios de transmissão não guiados, as quais, mas podem divergir de acordo com a velocidade dos dados trafegados e no seu alcance. A Figura 4 relaciona tecnologias de redes não guiadas de acordo com o seu alcance e a sua largura de banda.

Figura 4 – Largura de banda vs. distâncias suportadas de redes não guiadas.



Adaptado de: (HAIDINE et al., 2016)

Com base na Figura 4, observam-se as características de alcance e largura de banda de alguns grupos de tecnologias. No grupo que suporta largura de banda e alcance limitados, encontram-se as tecnologias ZigBee e Bluetooth. Em contrapartida, os protocolos de comunicação da família IEEE 802.11 têm um alcance superior, enquanto que os grupos de redes celulares têm consideráveis largura de banda e alcance em comparação com os demais. Já os protocolos LPWAN destacam-se pelo longo alcance, mas baixa largura de banda, devendo-se ressaltar que estes protocolos LPWAN caraterizados pelo baixo consumo de energia.

Os requisitos de largura de banda e de alcance são importantes, mas existem outros fatores que podem impactar na tomada de decisão de uma determinada abordagem para conexão e integração de dispositivos de borda num sistema de IIoT, tais como entidades padronizadoras, espectro de banda, custo, latência e segurança. Essas características são listadas na Tabela 3.

Tabela 3 – Características de tecnologias de comunicação para IoT e IIoT.

Padrão	**IEEE802.11**	**BLE**	**ZigBee Pro**	**SigFox**	**LoraWan**	**LTE**	**NB-IoT**	**5G**
Usa 3GPP?	Não	Não	Não	Não	Não	Sim (Release 13)	Sim (Release 13)	Sim (Release 15)
Área de cobertura	17–30 m	∼1–50 m²	∼1–250 m²	<12 km²	<10 km²	<10 km²	<15 km²	<12 km²
Espectro / largura de banda	2,4 GHz (802.11)	2,4 GHz (802.15.1)	2,4 GHz (802.15.4)	900 MHz	900 MHz	7–900 MHz	8–900 MHz	5–900 MHz (entre outras)
Taxa de comunicação	450 Mbps (802.11n)	1 (Mbps)	250 kbps	∼100–600 bps	∼200–50 kbps	<1 Mbps	<144 kbps	∼10 Gbps
Custo	4,00 USD (2016)	4,00 USD (2016)	3,00 USD (2016)	4,00 USD (2015) 2,64 USD (2020)	4,00 USD (2015) 2,64 USD (2020)	5,00 USD (2015) 3,30 USD (2020)	4,00 USD (2015) 2,00–3,00 USD (2020)	<2,00 USD
Latência	20–40 ms	6 ms	40 ms	1–30 s	61–371 ms	50–100 ms	1,6–10 s	5–50 ms
Segurança	256 bits	128 bits AES	128 bits	16 bits	32 bits AES-128	128–256 bits	128–256 bits	256 bits

Fonte: (CARVALHO; MIERS, 2021)

Avaliando os dados contidos na Tabela 3, observa-se que tecnologias que usam redes celulares, como *Long Term Evolution* (LTE), NB-IoT e *Fifth Generation Networks* (5G), são padronizadas pela sua entidade reguladora, e que o *3rd Generation Partnership Project* (3GPP) tem alcance maior que 10 quilômetros e variadas taxas de comunicação. As redes IEEE 802.11, Bluetooth e Zigbee, embora tenham uma frequência que gera uma estabilidade com relação ao sinal em torno de 2,4GHz de frequência, não têm aderência aos requisitos do cenário de automação e monitoramento de processos. Ou seja, não atende a área e a quantidade de dispositivos. Por sua vez, as redes SigFox atendem ao requisito de longa distância, mas apresentam uma estrutura de segurança bastante frágil, contando com apenas 16 bits de criptografia, além de não suportar a criação de redes privadas. Assim, a partir da análise das tecnologias presentes na Tabela 3 no que tange os requisitos de monitoramento ambiental constantes nas aplicações típicas de IIoT e presentes na Tabela 1, identifica-se a necessidade de uma rede de longo alcance e com baixo custo. As redes LPWAN, por outro lado, possuem um alcance elevado, como pode ser visto na Figura 4. Portanto, tendo em vista os quesitos de alcance e custo em conformidade com o requisitado, a tecnologia de comunicação escolhida para a solução proposta por este trabalho, para a integração de dispositivos foi o LoraWan. Uma tecnologia baseada em LPWAN desenvolvida pela entidade reguladora LoRa Alliance e que utiliza comunicação por modulação LoRa(KIM; KIM, 2015). Na Subseção 2.2.1, realiza-se uma análise preliminar da modulação LoRa e, na Subseção 2.2.2, um apanhado sobre o protocolo LoraWan, bem como os seus aspectos de segurança e de identidades.

2.2.1 LoRa

As redes *Long Range*, ou simplesmente LoRa, são uma técnica proprietária de modulação de espectro criada pela empresa Semtech promovida pela instituição sem fins lucrativos LoRa Alliance, derivada da tecnologia *Chirp Spread Spectrum* (CSS) desenvolvida para aplicações de radar na década de 1940 e tradicionalmente usada em aplicações militares (SEMTECH, 2015). A LoRa oferece soluções a problemas de comunicação sem fio com transmissões de longo alcance, baixo consumo de energia em sistemas de borda e transmissão segura de dados (BOR; VIDLER; ROEDIG, 2016), operando em várias frequências de rádio *Industrial Scientific Medical* (ISM) — frequências estas não restritas — em diferentes regiões do planeta. No Brasil, a Agência Nacional de Telecomunicações (ANATEL) publicou o ato 14.448 regulamentando a tecnologia LoRa no país e lhe designando frequências entre 915 a 928 Mega Hertz (MHz) (ANATEL, 2017) de uso não restrito, o que significa que, ao contrário de redes celulares como 4G ou 5G, não é necessário obter-se uma licença ou pagar-se uma taxa para utilizá-las.

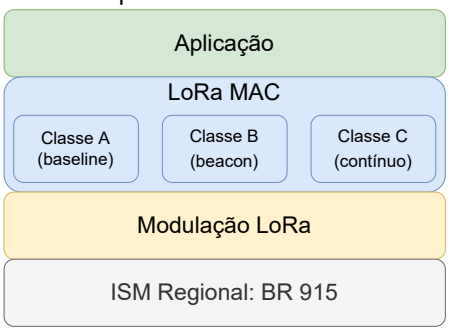
A LoRa também é uma tecnologia que permite comunicação a longas distâncias, alcançando, em áreas urbanas, de 3 a 4 quilômetros, enquanto que, em áreas rurais, pode alcançar 12 quilômetros ou mais, com um consumo mínimo de energia. As suas principais aplicações são em sistemas de IoT, como sensores e monitores remotos, sobretudo aqueles que fazem uso de mensagens curtas e, em alguns casos, de locais com uma ampla área de cobertura. Recentemente, essa técnica tem visto uma adoção crescente em várias aplicações de comunicação de dados devida aos seus requisitos de potência de transmissão relativamente baixos e a robustez inerente de mecanismos de degradação de canal e de interferência em banda (TOZETTO; SILVA; FREIRE, 2021).

Aqui, o espalhamento do espectro é obtido pela geração de um sinal de *chirp* CSS que varia continuamente em frequência, e um benefício desse método é que os deslocamentos de tempo e frequência entre o transmissor e o receptor são equivalentes, reduzindo bastante a complexidade do projeto do receptor. Além disso, a largura de banda de frequência do *chirp* é equivalente à largura de banda espectral do sinal. A LoRa é ainda um tecnologia com alta robustez, escalável em largura de banda e com sinal resistente ao desvanecimento e à degradação. De forma pura, este sinal não implementa um protocolo de comunicação, o qual exige, entre dispositivos, que se tenha um protocolo implementado para além da modulação utilizada na camada física. O protocolo mais utilizado é o LoraWan.

2.2.2 LoraWan

O protocolo LoraWan é definido pela Lora Alliance — grupo integrado por diversas empresas do ramo como IBM, Actility, Semtech e Microchip — como um protocolo de redes LPWAN que foi projetado para conectarem-se dispositivos operados à bateria numa rede sem fio de longo alcance. A versão 1.0 do protocolo foi liberada no ano de 2015, passando por atualizações nos anos seguintes até a sua atualização para versão 1.1 em 2017. Esse protocolo tem foco em IoT, oferecendo serviços de comunicação bidirecional, mobilidade e localização (ALLIANCE, 2017). Este possui uma arquitetura de rede aberta, a qual pode ser vista na Figura 5.

Figura 5 – Arquitetura em camadas LoraWan.



Adaptado de: (ALLIANCE, 2017)

A arquitetura do LoraWan é dividida em camadas. A camada de aplicação, de LoRa *Medium Access Control* (MAC), a camada de modulação (física) e, na base, o meio de transmissão de ISM. A camada de aplicação compreende o fornecimento de serviços ao utilizador, através dos dados armazenados, e a infraestrutura da rede. A camada MAC estabelece a conexão entre o *gateway* LoRa e o dispositivo final e trata da transmissão e recepção de comandos e dados MAC da camada de aplicação (FEHRI et al., 2018). A presente arquitetura é a mesma, independentemente da versão se 1.0 ou 1.1. Nesta, os dispositivos que se comunicam com os *gateways* são categorizados em classes para utilização adequada em cada projeto. As categorias classificam os dispositivos conectados pelo seu fluxo de comunicação e pelo gasto de energia, sendo estas:

- Classe A: dispositivos bidirecionais, mas com possibilidade de receber mensagens (*downlink*) restritas logo após o envio (*uplink*). Nesta forma de operação, reduz-se o consumo de energia, uma vez que, durante o restante do tempo, o módulo LoRa estará desligado.
- Classe B: dispositivos bidirecionais, mas que têm a possibilidade de receber mensagem com tarefas sincronizadas de tempo em tempo. Ao enviar-se uma mensagem, pode ser criada uma tarefa para sincronizar o próximo horário em que o dispositivo estará ouvindo.
- Classe C: dispositivo bidirecional, mas com a possibilidade de recebimento de mensagens. Esses dispositivos possuem praticamente uma janela de recepção aberta, estando fechada somente quando está transmitindo. Esta é indicada a dispositivos conectados diretamente à rede elétrica, devido ao seu expressivo consumo de energia, e a aplicações que necessitam de uma menor latência no *downlink*.

No cenário de IIoT, em aplicações de automação e monitoramento de processos, podem ser implementados dispositivos de todas as classes, cuja escolha depende

dos requisitos de energia, fluxo de dados e latência (PÖTSCH; HAMMER, 2019). Por sua vez, a camada de modulação LoRa implementa o meio de comunicação sem fio.

Além das camadas de arquitetura, o LoraWan pode ser descrito por meio da sua topologia de rede em estrela, na qual os *gateways* retransmitem mensagens entre os dispositivos finais e um servidor de rede central. Esses *gateways* são conectados ao servidor de rede por meio de conexões IP padrão e atuam como uma ponte transparente, simplesmente convertendo pacotes de radiofrequência em pacotes IP e vice-versa.

Todos os modos são capazes de comunicação bidirecional, e há suporte a grupos de endereçamento *multicast* para fazer-se o uso eficiente do espectro durante tarefas como atualizações de *Firmware Over-The-Air* (FOTA) ou outras mensagens de distribuição em massa (ALLIANCE, 2017).

Embora a arquitetura em camadas, o padrão de dispositivos e alguns componentes mínimos sejam comuns as versões 1.0 e 1.1, existem diferenças substanciais. Entre outras dissimilitudes estão as chaves de sessão, comandos MAC, quantidades de entidades e questões relacionadas ao *roaming* conforme relacionado na Tabela 5.

Tabela 4 – LoraWan 1.0 *vs.* LoraWan 1.1.

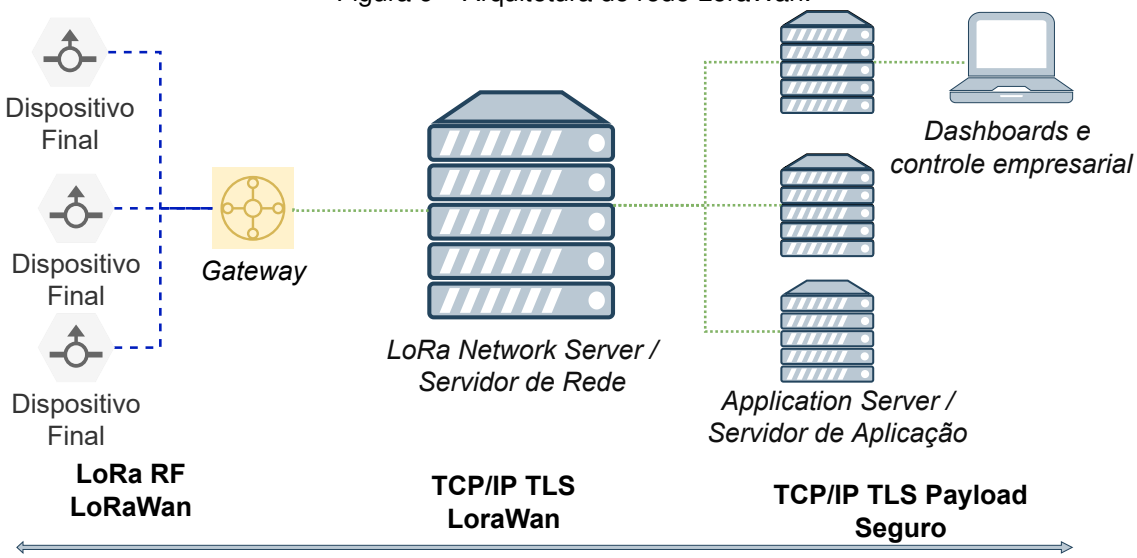
	LoraWan 1.0	**LoraWan 1.1**
Chaves de Sessão	AppSkey e NwkSkey	FNwkSIntKey, SNwkSIntKey, NwkSEncKey e AppSKey
Comandos MAC	Não cifrados	Cifrados
Entidades	Dispositivos finais, *Network Server* e *Application Server*.	Dispositivos finais, *Join Server*, *Network Server* e *Application Server*
Roaming entre redes LoraWan	Não suporta *roaming* entre redes	Suporta *roaming* entre redes

Adaptado de: (ALLIANCE, 2017).

Conforme relacionado na Tabela 5, além das diferenças em relação a chaves de sessão, comandos MAC e *roaming* existem diferenças na definição da topologia do protocolo. A entidade *Join Server* é adicionada a arquitetura e tem responsabilidade de fazer a autenticação dos dispositivos finais, responsabilidade que na versão 1.0 era do *network server*.

Este trabalho, em um primeiro momento, se vale de uma definição geral do protocolo LoraWan, traçando comparações entre as versões em quesitos genéricos e em relação a estrutura de segurança. Porém, este estudo, encontra-se focado na versão 1.1 por ser a versão mais recente e recomendada para ambiente com requisitos de segurança (DöNMEZ; NIGUSSIE, 2018). Entretanto, mesmo esta versão atualizada, permanece sem uma solução adequada de gerenciamento seguro de credenciais (CHEN; WANG; WANG, 2019; TSAI et al., 2020; TSAI et al., 2019). A topologia de rede do LoraWan 1.1 pode ser observada na Figura 6.

Figura 6 – Arquitetura de rede LoraWan.



Adaptado de: (ALLIANCE, 2017).

Como observado na Figura 6, de maneira geral, a topologia é composta por dispositivos finais, um ou mais *gateways*, servidores de rede e servidores de aplicação. Os dispositivos finais são os elementos básicos da rede, como sensores de temperatura, sensores de umidade, leitores de consumo, entre outros, além de compreenderem atuadores. Estes não têm a capacidade de roteamento de pacote entre si, de forma que todas as mensagens que são trocadas são enviadas diretamente ao *gateway*.

Os *gateways*, também denominados concentradores, são estações rádio base que fazem o intermédio e a retransmissão das mensagens entre os dispositivos finais e os servidores de rede, os quais são os responsáveis por receber as informações enviadas pelos *gateways*, processá-las e enviá-las à aplicação. Por fim, os servidores de aplicação, ao receberem as informações do servidor, podem tomar ações específicas de atuação na camada de borda. Na camada de aplicação existem também os aplicativos de visualização de dados e de controle da rede (SEMTECH, 2019).

A integração de dispositivos nessa rede, bem como o tráfego seguro de dados, são importantes requisitos a serem considerados nessa topologia. Na Subseção 2.2.2.1, os principais tópicos de segurança em redes LoraWan são discutidos.

2.2.2.1 Segurança no LoraWan

Existem dois elementos-chave para a segurança de uma rede LoraWan: o procedimento de junção de dispositivos à rede, o chamado *join procedure*, e a autenticação da mensagem. O procedimento de junção estabelece a autenticação mútua entre um dispositivo final e a rede LoraWan à qual este está conectado, sendo que somente dispositivos autorizados têm permissão para ingressar na rede. As mensagens Lo-

raWan MAC e de aplicativos são autenticadas na origem, protegidas por integridade e cifradas de ponta a ponta com o algoritmo *Advanced Encryption Standard* (AES). Esses recursos de segurança garantem que o tráfego de rede não será alterado e que apenas dispositivos legítimos estão conectados à rede LoraWan. Além disso, com esses recursos, o tráfego de rede não pode ser ouvido por atacantes e não pode ser capturado e reproduzido.

A segurança em LoraWan tem distinções em relação a versão utilizada. Na versão 1.0 do protocolo, existiam brechas que em tese poderiam ser explorada por atacantes. A versão 1.1 teve atualizações em relação a estas vulnerabilidades além de aprimorar a designação de responsabilidades de segurança de cada entidade da arquitetura (ALLIANCE, 2017). Na Tabela 5 observa-se as principais diferenças entre as versões em termos de segurança.

Tabela 5 – Segurança em LoraWan 1.0 e LoraWan 1.1.

	LoraWan 1.0	**LoraWan 1.1**
Entidade responsável pelo processo de junção	Network Server	Join Server
Reutilização de contadores de mensagens	Por não persistir contadores pode comprometer a criptografia e comunicação.	Corrigido armazenando o último valor do contador em memória não volátil.
Repetição de mensagem de solicitação mal sucedida	Um invasor deve esperar apenas N mensagens antes de reproduzir uma mensagem de solicitação de junção.	A repetição de mensagens do tipo *join request* é impedida pela rede não incrementais.
Sem proteção de ponta a ponta	Como o AppKey é compartilhado tanto pelo servidor de rede quanto pelos servidores de aplicativos está sujeito a interceptação.	Proteção de ponta a ponta para OTAA é alcançada fornecendo duas chaves diferentes para a rede (NwkKey) e para o aplicativo (AppKey), das quais todas as outras chaves são derivadas.

Adaptado de: (LOUKIL et al., 2022).

Na Tabela 5 pode-se observar que houveram melhorias em termos da arquitetura de segurança, prevenção de ataques de repetição e proteção ponta a ponta. Entretanto, não houveram atualizações sobre o gerenciamento de identidades. Neste trabalho detalha-se as especificações referentes á versão 1.1 do protocolo e emprega-se esta versão nos experimentos de pesquisa.

De forma geral, esta seção dedica-se a entender aspectos gerais de segurança em LoraWan. Em sequência, na Subseção 2.2.2.2, o trabalho dedica-se a entender mais detalhadamente o processo de *join procedure*. Isto nos permite compreender melhor o processo de autenticação, identidades e credenciais envolvidos nos processos de controle de acesso na arquitetura do LoraWan.

2.2.2.2 Controle de acesso e gestão de identidades no LoraWan

O processo de gestão de identidade no LoraWan agrega à sua arquitetura uma nova entidade, o *Join Server*, responsável por gerenciar o processo de ativação dos dispositivos finais na rede e que contém a informação necessária para processar as solicitações de incorporação à rede e também os *frames* de aceitação das operações de *downlink* (SEMTECH, 2019). O *Join Server* sinaliza ao servidor de rede qual servidor de aplicativos deve ser conectado ao dispositivo final e executa as derivações de chave de criptografia de sessão de rede e de aplicativo. O *Join Server* ainda comunica a chave de sessão de rede do dispositivo ao servidor de rede e a chave de sessão do aplicativo ao servidor de aplicativos correspondente. Para isso, o servidor de aplicação correspondente deve contar com as seguintes informações para cada dispositivo final sob o seu controle: *End-device serial Unique Identifier* (DevEUI), *Application encryption Key* (AppKey), *Network Encryption Key* (NwkKey), *application server identifier*, que serve como identificador do servidor de aplicação, e, por fim, um perfil de serviço do dispositivo final, o *end-device service profile*.

Contando com o *Join Server* e as suas funcionalidades, é possível fazer o comissionamento de dispositivos. Estes devem ser comissionados e ativados na rede no início da operação do dispositivo para a garantia da sua autenticidade na rede, processo esse que relaciona, com segurança, cada dispositivo e a rede, tendo em vista os seus parâmetros essenciais de provisionamento. A especificação LoraWan permite dois tipos de ativação: (i) *Over-the-Air Activation* (OTAA), modo colocado como preferencial; e (ii) *Activation by Personalization* (ABP). A Tabela 6 lista as diferentes características de cada um desses tipos de ativação.

Tabela 6 – Tipos de ativação no LoraWan.

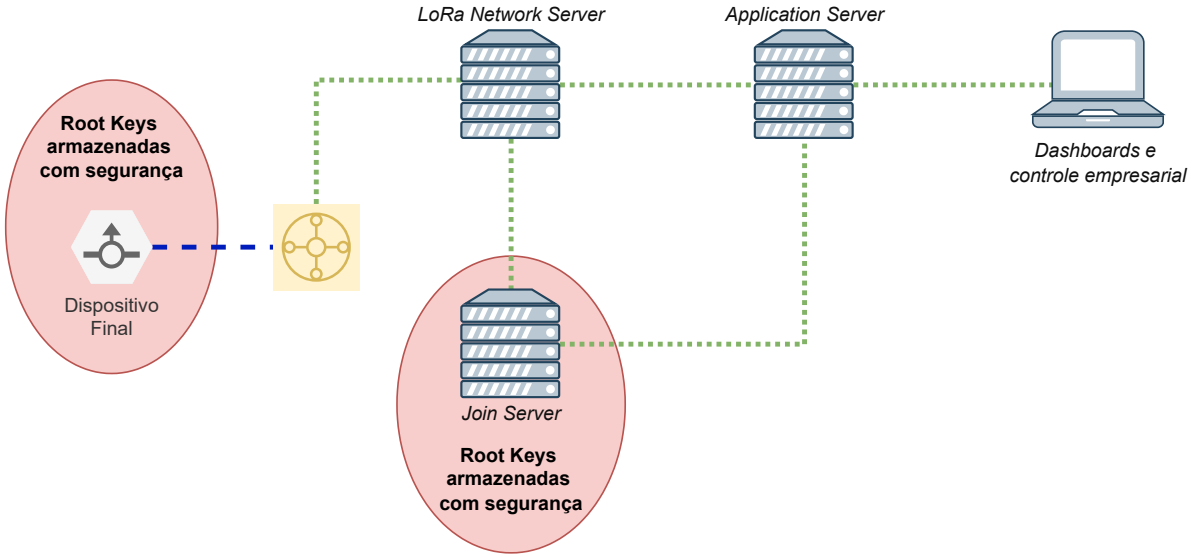
	Over-the-air activation (OTAA)	**Activation by personalization (ABP)**
Chaves no processo de comissionamento	Chaves seguras (de duração de sessão e derivadas) podem ser renovadas regularmente.	Um processo de comissionamento simplificado (menos seguro).
Atribuição de Identidades para os dispositivos	Os dispositivos podem armazenar múltiplas identidades para alternar redes e operadoras de forma dinâmica e segura durante o seu ciclo de vida.	IDs e chaves são personalizados na fabricação.
Parâmetros de provisionamento	Os fabricantes de dispositivos geram de forma autônoma parâmetros de provisionamento essenciais.	Os dispositivos tornam-se imediatamente funcionais ao serem ligados; o procedimento de comissionamento é ignorado.
Opções a prova de adulteração	Opções de segurança de alta qualidade e à prova de adulteração estão disponíveis.	Os dispositivos estão vinculados a uma rede/serviço específico; o NetID é uma parte do endereço de rede do dispositivo.

Adaptado de: (SEMTECH, 2019).

A ABP usa um método de ativação em que são usadas credenciais definidas na fabricação, é mais simples e menos seguro, já que o processo de comissionamento é ignorado. O ABP tem como limitação, o fato de que os dispositivos ficam vinculados a uma rede ou serviço específico, comprometendo a sua versatilidade. Por sua vez, a OTAA usa um método cujas chaves podem ser renovadas regularmente, favorecendo a sua rotatividade. Além disso, nesse método de ativação, os dispositivos podem armazenar diversas identidades, podendo operar em diversas redes durante o seu ciclo

de vida na rede. De modo geral, os itens elementares do processo de comissionamento são as chaves usadas para autenticar os dispositivos na rede. As chaves de segurança, conforme ilustrado na Figura 7, devem ser armazenadas com segurança nos dispositivos finais, e as chaves correspondentes são armazenadas com segurança no *Join Server*.

Figura 7 – Chaves geradas no processo de *join procedure*.

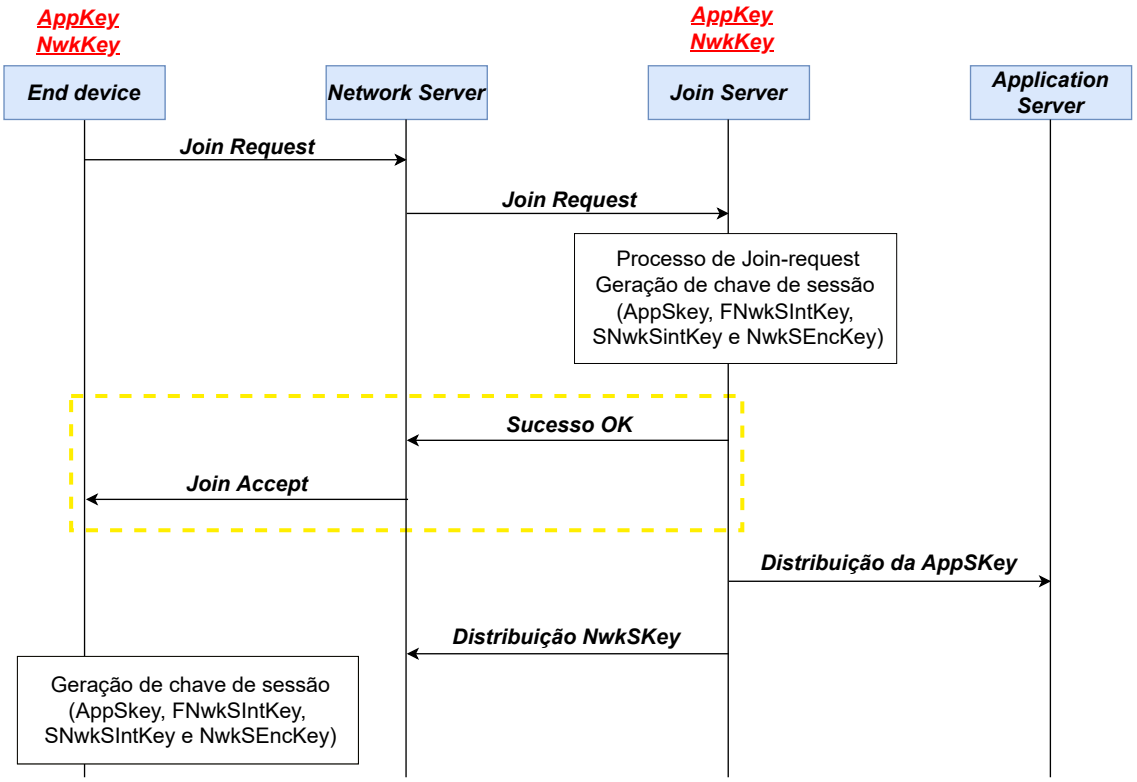


Adaptado de: (SEMTECH, 2019).

O *Join Server* é uma nova entidade introduzida na especificação o versão 1.1 do LoraWan responsável por lidar com a autenticação do dispositivo final. Este verifica a autenticidade do dispositivo final e usa as chaves-raiz e os parâmetros relacionados para derivar as chaves de sessão e o endereço do dispositivo final (SEMTECH, 2019).

No LoraWan, o procedimento de comissionamento requer que duas mensagens MAC sejam trocadas entre o dispositivo final e o *Network Server*: o *join request*, do dispositivo final ao servidor de rede, e o *join accept*, do servidor de rede ao dispositivo final. Antes da ativação, as chaves JoinEUI, DevEUI, AppKey e NwkKey devem ser armazenados no dispositivo final. A AppKey e a NwkKey são chaves simétricas (AES) conhecidas como chaves-raiz. O AppKey, NwkKey e o DevEUI correspondentes devem ser provisionados no *Join Server*, que ajudará no processamento do procedimento de integração do dispositivo e na derivação da chave de sessão. O O JoinEUI e o DevEUI, por sua vez, não são secretos e visíveis a todos. Na Figura 8, observa-se o procedimento de adesão de um dispositivo em redes LoraWan.

Figura 8 – Fluxo de mensagens OTAA no LoraWan 1.1.



Adaptado de: (DANISH et al., 2020).

O procedimento de comissionamento é sempre iniciado pelo dispositivo final. O dispositivo final envia a mensagem de *join request* à rede que será ingressada. Essa mensagem consiste dos seguintes campos:

- JoinEUI (8 *bytes*): um identificador de aplicativo global de 64 *bits* no espaço de endereço IEEE EUI64 que identifica exclusivamente o *Join Server* que pode auxiliar no processamento da *join request* e na derivação das chaves de sessão;
- DevEUI (8 *bytes*): um identificador de aplicativo global de 64 *bits* no espaço de endereço IEEE EUI64 que identifica exclusivamente o *Join Server* que pode auxiliar no processamento da *join request* e derivação das chaves de sessão; e
- DevNonce (2 *bytes*): um contador de 2 *bytes* que inicia em 0 quando o dispositivo é inicialmente ligado e incrementa com cada *join request*. O valor DevNonce é usado para evitar ataques de repetição.

O *Message Integrity Code* (MIC) é calculado sobre todos os campos na mensagem de *join request* usando o NwkKey. O MIC calculado é então adicionado à mensagem de *join request*. A mensagem de *join request* pode ser transmitida usando qualquer taxa de dados e um dos canais de junção específicos da região. Por exemplo, na rede com definições para o Brasil, um dispositivo final pode transmitir a mensagem

de *join request* escolhendo aleatoriamente entre 915 a 928 MHz. A mensagem de solicitação de associação passa por um ou mais *gateways* para o servidor de rede, o qual encaminha a mensagem de *join request* ao *Join Server* correspondente. O *Join Server* processa a mensagem de *join request*, e gerará todas as chaves de sessão (AppSKey, FNwkSIntKey, SNwkSIntKey e NwkSEncKey) se o dispositivo final tiver permissão para ingressar na rede. Se a mensagem for autêntica e consequentemente for bem sucedida, o *network server* gerará a mensagem de *join accept*, que consiste dos campos:

- `JoinNonce`: um valor de contador específico do dispositivo fornecido pelo *Join Server* e usado pelo dispositivo final para derivar as chaves de sessão FNwkSIntKey, SNwkSIntKey, NwkSEncKey e AppSKey;
- `NetID`: um identificador de rede exclusivo de 24 *bits*;
- `DevAddr`: um endereço de dispositivo de 32 *bits* atribuído pelo servidor de rede para identificar o dispositivo final na rede atual;
- `DLSettings`: um campo de 1 *byte* que consiste em configurações de *downlink* que o dispositivo final deve usar;
- `RxDelay`: contém o atraso entre o TX (transmissor) e o RX (receptor); e
- `CFList`: uma lista opcional de frequências de canal para a rede na qual o dispositivo final está conectando-se. Essas frequências são específicas da região.

O MIC é calculado sobre todos os campos na mensagem de *join accept* usando NwkKey (para dispositivos LoraWan 1.0) ou JSIntKey (para dispositivos LoraWan 1.1). O MIC computado é, então, adicionado à mensagem de *join accept*, a qual é cifrada com a NwkKey. O servidor de rede usa uma operação de descriptografia AES no modo *Electronic CodeBook* (ECB) para cifrar essa mensagem e, em seguida, a envia, cifrada, de volta ao dispositivo final como um *downlink* normal.

Dentre os modos de ativação de dispositivos no LoraWan e levando-se em conta os requisitos do contexto da IIoT, considera-se o método de ativação OTAA como adequado para o caso de uso de monitoramento ambiental, tendo em vista os fatores de segurança e flexibilidade para comissionarem-se dispositivos em diversas redes. No método de ativação OTAA, existe a possibilidade de estender-se e personalizar-se a ativação de dispositivos por meio de um *join procedure* personalizado. A personalização faz-se necessária para o caso da gestão de credenciais personalizada. A garantia de aplicação das boas práticas relacionadas ao gerenciamento de credenciais abordadas na Seção 2.1, correlacionada aos procedimentos nativos de ativação

de dispositivos finais e as suas credenciais, é discutida como uma problemática deste trabalho.

2.3 CHIRPSTACK

O ChirpStack é solução de software para prover um servidor LoRaWAN, que pode ser usado para configurar redes LoRaWAN públicas ou privadas. O ChirpStack fornece uma interface web para o gerenciamento de *gateways* e dispositivos bem como para configurar integrações de dados com os principais provedores de nuvem, bancos de dados e serviços comumente usados para lidar com dados de dispositivos. O ChirpStack fornece uma *Application Programming Interface* (API) baseada em *Google Remote Procedure Call* (gRPC) que pode ser usada para integrar ou estender o mesmo. O servidor contempla também uma API JSON REST Full. O ChirpStack é o servidor de rede para LoraWan *open source* mais largamente utilizado, com a maior comunidade ativa e com atualizações periódicas e atuais (LUND, 2022). A primeira versão foi disponibilizada no ano de 2019 e a versão atual (no momento da escrita deste texto no ano de 2022) é a “V4”, disponibilizado sob a licença MIT.

2.3.1 Arquitetura

O ChirpStack funciona como um ambiente completo para *setups* de LoraWan. Sua arquitetura é semelhante a arquitetura LoraWan citada na Figura 6. A arquitetura básica do ChirpStack está dividida em seis componentes básicos:

(i) Dispositivos Finais: são os dispositivos que enviam dados para o ChirpStack Network Server, através de um ou vários LoRa Gateways. Esses dispositivos podem ser, e.g., sensores que medem a qualidade do ar, temperatura, umidade, localização, etc.

(ii) Lora Gateway: monitora, normalmente, 8 ou mais canais simultaneamente e encaminha os dados recebidos (de dispositivos) para um servidor de rede LoRaWAN (neste caso o servidor de rede ChirpStack). O software em execução, responsável pelo recebimento e envio, é chamado de *Packet Forwarder*.

(iii) ChirpStack Gateway Bridge: posicionado entre o Packet Forwarder e o Broker *Message Queuing Telemetry Transport* (MQTT). Este transforma o formato Packet Forwarder (como o protocolo Semtech UDP Packet Forwarder) em um formato de dados usado pelos componentes ChirpStack. Este também fornece integrações com nuvens, e.g., Google Cloud Plataform IoT Core e Azure IoT Hub.

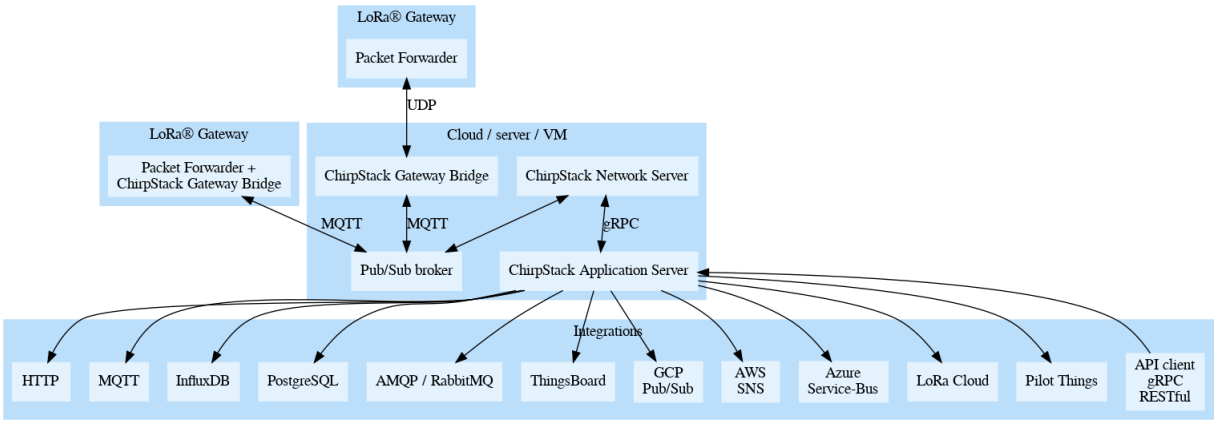
(iv) Servidor de rede ChirpStack: é um servidor de rede LoRaWAN, responsável por gerenciar o estado da rede. Este tem conhecimento das ativações de dispositivos

na rede e é capaz de lidar com solicitações de junção quando os dispositivos desejam ingressar na rede. Quando os dados são recebidos por vários *gateways*, o ChirpStack Network Server eliminará as duplicações desses dados e os encaminhará como uma carga útil para o ChirpStack Application Server. Quando um servidor de aplicativos precisa enviar dados de volta para um dispositivo, o ChirpStack Network Server manterá esses itens na fila, até que possa enviar para um dos *gateways*.

(v) Servidor de aplicação ChirpStack: é um servidor de aplicativos LoRaWAN, compatível com o servidor de rede ChirpStack. Este fornece uma interface web e APIs para gerenciamento de usuários, organizações, aplicativos, *gateways* e dispositivos.

(vi) Aplicação final: recebe os dados do dispositivo por meio de uma das integrações configuradas, pode usar a API ChirpStack Application Server para encaminhar um *payload* de *downlink* para os dispositivos. O objetivo de um aplicativo final pode ser análise, alerta, visualização de dados, ações de acionamento entre outras formas de consumo de dados e controle.

Figura 9 – Arquitetura ChirpStack.



Adaptado de: (CHIRPSTACK, 2022).

No primeiro nível superior da Figura 9 é representado o Lora Gateway, que tem a função de encaminhador de pacotes. O componente de software *Packet Forwarder* é o responsável por operacionalizar o encaminhamento dos pacotes. Esta comunicação é feita com protocolo *User Datagram Protocol* (UDP). Em termos de IIoT, conforme a Figura 1 o *gateway* faz a ligação entre a Camada de Borda e a Camada de Plataforma.

O segundo nível da Figura 1 tem representados uma versão do Lora Gateway com o software ChirpStack Gateway Bridge e as entidades da infraestrutura do ChirpStack. Esta versão do Lora Gateway permite o encaminhamento de pacotes usando a tecnologia MQTT a um *broker*. As outras entidades deste nível são as responsáveis por operacionalizar o sistema LoraWan, por meio do servidor de rede e aplicação.

No terceiro nível da Figura 1 observa-se as integrações disponíveis no ChirpStack por meio de diversos protocolos como *HyperTextTransferProtocol* (HTTP) e MQTT, plataformas de nuvem e bancos de dados. Este nível, em termos de IIoT se refere a camada empresarial.

Esta representação de arquitetura detalha em alto nível o funcionamento geral da implementação de rede LoraWan privada. Levando em conta o objetivo deste trabalho de analisar o ciclo de vida das credenciais neste contexto, faz-se necessário o detalhamento da utilização de credenciais neste cenário. A Subseção 2.3.2 detalha as credenciais usadas bem como seu emprego em relação ao seu ciclo de vida.

2.3.2 Credenciais no ChirpStack

O ChirpStack já possui implementada o LoraWan versão 1.1. Portanto, o processo de ativação de dispositivos se dá pelo método OTAA ou ABP. Em ambos os métodos, na autenticação de dispositivos, pode-se categorizar as credenciais primeiramente pela ordem de geração e armazenagem no dispositivo. Primeiramente são listadas as credenciais geradas antes da ativação (ALLIANCE, 2017):

- JoinEUI: é um ID de aplicação global mantido pela IEEE, sendo um identificador exclusivo denominado *Extended Unique Identifier* (EUI) de 64 bits. Isto determina sua intitulação usual, EUI64. Esta credencial identifica exclusivamente o *Join Server* que é capaz de auxiliar no processamento do *join procedure* e na derivação das chaves de sessão. Para dispositivos OTAA, o JoinEUI deve ser armazenado no dispositivo final antes que o *join procedure* seja executado. O JoinEUI não é necessário apenas para dispositivos finais que implementam o ABP.

- DevEUI: é um identificador global de dispositivos finais no espaço de endereço IEEE EUI-64 que identifica o dispositivo final de forma única globalmente. DevEUI é o identificador de dispositivo exclusivo recomendado pelo(s) servidor(es) de rede, independentemente do procedimento de ativação usado, para identificar um dispositivo em *roaming* entre redes. Para dispositivos OTAA, o DevEUI deve ser armazenado no dispositivo final antes que o procedimento de *join* seja executado. Os dispositivos ABP não precisam que o DevEUI seja armazenado no próprio dispositivo, mas é recomendado fazê-lo. Ademais, é uma prática recomendada que o DevEUI também seja disponível em uma etiqueta do dispositivo, para administração do dispositivo.

- AppKey: é uma chave AES-128 específica para o dispositivo final que é atribuída ao dispositivo final durante a fabricação. Sempre que um dispositivo final

ingressa em uma rede por meio de ativação sem fio o AppKey é usado para derivar a chave de sessão AppSKey. O provisionamento e armazenamento seguro desta credencial dependem da implementação geral da solução. Esta implementação pode incluir *Secure Elements* (SE) e *Hardware Security Modules* (HSM). A AppKey deve obrigatoriamente ser armazenada nos dispositivos no método de ativação OTAA, não sendo necessária no método ABP. Esta credencial ainda deve ser armazenada de forma a evitar a extração e o uso por atacantes.

- NwkKey: é uma chave AES-128 do dispositivo final, atribuída ao dispositivo final durante a fabricação. Sempre que um dispositivo final ingressa em uma rede por meio de ativação OTAA, o NwkKey é usado para derivar as chaves de sessão FNwkSIntKey, SNwkSIntKey e NwkSEncKey. Assim como a AppKey, esta credencial deve ser armazenada e provisionada com segurança de acordo com a implementação geral de segurança do sistema computacional. Quando utilizada a versão 1.0 do LoraWan, a NwkKey serve para derivar as chaves de sessão AppSKey e FNwkSIntKey. A NwkKey deve ser armazenada em um dispositivo final que pretenda usar o procedimento OTAA. Não sendo necessária apenas para dispositivos finais ABP.
- JSIntKey: para dispositivos OTAA, duas chaves de vida útil específicas são derivadas da chave raiz NwkKey, a JSIntKey é usada para gerar a mensagem de *rejoin-request*, além de mensagens e respostas *join-accept*.
- JSEncKey: é usada para cifrar o *join-accept* acionado por um *rejoin-request*.

Após a ativação, as seguintes informações adicionais são armazenadas no dispositivo final (ALLIANCE, 2017):

- DevAddr: consiste em 32 bits e identifica o dispositivo final dentro da rede atual, sendo alocado pelo *Network Server* do dispositivo final. Seu formato é apresentado na Tabela 7, sendo *n* é um número inteiro no intervalo entre 7 e 24:

Tabela 7 – Campos DevAddr.

Bit#	[31..32-*n*]	[31-*n*..0]
DevAddr bits	AddrPrefix	NwkAddr

Fonte: (ALLIANCE, 2017).

O campo AddrPrefix de tamanho variável é derivado do identificador exclusivo NetID do *Network Server*, que é alocado pela LoRa Alliance quando não se trata de rede experimental/privada. O campo AddrPrefix permite a descoberta do *Network Server* que está gerenciando o dispositivo final durante eventuais trocas de rede (*roaming*). Os bits menos significativos (32-*n*), o endereço de

rede (NwkAddr) do dispositivo final, podem ser atribuídos arbitrariamente pelo gerenciador de rede.

- FNwkSIntKey: A Forwarding Network Dession Integrity key (FNwkSIntKey) é uma chave de sessão de rede específica para o dispositivo final. Este é usado pelo dispositivo final para calcular o MIC[1] ou parte do MIC (código de integridade da mensagem) de todas as mensagens de dados de *uplink* para garantir a integridade dos dados. O FNwkSIntKey deve ser armazenado de forma a evitar a extração e reutilização por agentes mal-intencionados.

- SNwkSIntKey: é uma chave de sessão de rede específica para o dispositivo final. Esta é usada pelo dispositivo final para verificar o MIC de todas as mensagens de dados de *downlink* para garantir a integridade dos dados e para calcular as mensagens de *uplink* MIC. O SNwkSIntKey deve ser armazenado de forma a evitar a extração e reutilização por agentes mal-intencionados.

- NwkSEncKey: é uma chave de sessão de rede específica para o dispositivo final. Este é usado para cifrar e decifrar comandos MAC de *uplink* e *downlink* transmitidos como carga útil.

- AppSKey: é uma chave de sessão de aplicação específica para o dispositivo final. Esta é usada pelo servidor de aplicação e pelo dispositivo final para cifrar e decifrar o campo de carga útil das mensagens de dados específicas do aplicativo. Os *payloads* de aplicativos são cifradas de ponta a ponta entre o dispositivo final e o servidor de aplicativos, mas são protegidas por integridade apenas de um modo “salto a salto”: um salto entre o dispositivo final e o servidor de rede e o outro salto entre o *Network Server* e o *Application Server*. Isso significa que um *Network Server* malicioso pode ser capaz de alterar o conteúdo das mensagens de dados em trânsito, o que pode até ajudar o *Network Server* a inferir algumas informações sobre os dados observando a reação dos terminais da aplicação aos dados alterados. Servidores de rede são considerados confiáveis, mas os aplicativos que desejam implementar confidencialidade de ponta a ponta e proteção de integridade podem usar soluções de segurança de ponta a ponta adicionais.

As credenciais usadas no processo de comissionamento, utilização e descomissionamento de dispositivos tem papeis diferentes em referência ao seu escopo de utilização e armazenamento. As credenciais são usadas para finalidades diferentes e estão envolvidas em etapas distintas do ciclo de vida de credenciais (Subseção

[1] O MIC é responsável por calcular a integridade da mensagem e o cálculo do MIC de *uplink* depende de duas chaves (FNwkSIntKey e SNwkSIntKey) para permitir que um servidor de rede de encaminhamento em uma configuração de *roaming* possa ser verificado.

2.1.2.2) e na Tabela 8 são relacionadas as credencias identificadas em relação ao ciclo de vida.

Tabela 8 – Credenciais ChirpStack LoraWan e ciclo de vida.

	Ciclo de vida	Etapas
JoinEUI	Credencial fixa, permeia todo o processo de autenticação.	Vinculação, Armazenamento, Suspensão e Revogação.
DevEUI	Credencial fixa, permeia todo o processo de autenticação.	Vinculação, Armazenamento, Suspensão e Revogação.
AppKey	Credencial fixa, permeia todo o processo de autenticação.	Vinculação, Armazenamento, Suspensão e Revogação.
NwkKey	Credencial fixa, permeia todo o processo de autenticação.	Vinculação, Armazenamento, Suspensão e Revogação.
JSIntKey	Credencial derivada da NwkKey usada durante a vida útil do dispositivo nas mensagens de *join* e *rejoin*.	Vinculação e Armazenamento.
JSEncKey	Credencial usada para cifrar o *join-accept* acionado por um *rejoin-request*.	Vinculação.
DevAddr	um endereço de dispositivo de 32 bits atribuído pelo servidor de rede para identificar o dispositivo final na rede atual.	Vinculação e Armazenamento.
FNwkSIntKey	Chave de sessão de rede específica para o dispositivo final calcular a mensagem de integridade nos *uplinks*.	Armazenamento, suspensão e revogação.
SNwkSIntKey	Chave de sessão de rede específica para o dispositivo final calcular a mensagem de integridade nos *downlinks*.	Armazenamento, suspensão e revogação.
NwkSEncKey	Chave de sessão de rede específica para o dispositivo final. Este é usado pelo servidor de rede para cifrar e decifrar comandos MAC de *uplink* e *downlink* transmitidos como carga útil.	Vinculação, armazenamento, suspensão e revogação.
AppSKey	Chave de sessão de rede específica para o dispositivo final. Este é usado pelo servidor de aplicação para cifrar e decifrar comandos MAC de *uplink* e *downlink* transmitidos como carga útil.	Vinculação, armazenamento, suspensão e revogação.

Fonte: O autor.

Na Tabela 8 verifica-se qual a funcionalidade e as etapas do ciclo de vida envolvidas. Com isto, se estabelece uma correlação entre a arquitetura de uma implementação de LoraWan usando o servidor ChirpStack com o ciclo vida de credenciais me IIoT objeto de análise deste trabalho.

Partindo deste ponto de intersecção entre a implementação das credenciais no ChirpStack Lorawan e o ciclo de vida de credenciais abre-se espaço para problemáticas relativas a aderência desta implementação aos padrões estabelecidos e explorados neste trabalho.

2.4 DEFINIÇÃO DO PROBLEMA

O desenvolvimento da IIoT é cercado de diversos desafios em se tratando da sua implementação de forma segura, estável e rentável às organizações industriais. Os desafios de construção de um sistema de IIoT seguro, confiável e estável dizem respeito à implementação de uma arquitetura referencial e políticas de segurança baseados em normas e padrões de entidades reguladoras. Outras características do caso de uso das aplicações de IIoT também trazem considerações adicionais a serem feitas na implementação de uma rede IIoT confiável e estável. Além disso, a escolha

de tecnologias de comunicação para compor esse sistema de IIoT são determinantes na definição do contexto final.

O cenário apresentado neste trabalho é o de uma arquitetura de camadas para IIoT com o caso de uso de automação e monitoramento de processos. Esse padrão arquitetural e esse caso de uso final resultam num sistema IIoT com requisitos de uma ampla área de distribuição dos dispositivos finais no espaço físico da organização. Para a concretização desse cenário de IIoT, ainda faz-se necessário definir uma tecnologia de comunicação para os dispositivos na camada da borda. Neste trabalho, sob critérios de baixo custo dos dispositivos finais, de longo alcance, de economia de energia e de flexibilidade, o protocolo de LPWAN LoraWan foi estabelecido como a tecnologia de comunicação de redes não guiadas no cenário proposto. Em termos de LoraWan, muitas organizações necessitam de implantação como rede privada, garantindo isolamento de possíveis atacantes. Ademais, o ambiente utilizado nos procedimentos do trabalho fará uso do componente de software ChirpStack para implementar as entidades do protocolo. Ainda que o LoraWan com ChirpStack satisfaça esses requisitos, a abordagem tradicional do protocolo não prevê um ciclo de vida definido para as credenciais usadas para verificar-se a autenticidade dos dispositivos.

Diante do cenário de estudo proposto, há a preocupação de como gerenciar-se seguramente dispositivos de IIoT que necessitam possuir credenciais seguras e que possam ser gerenciadas de forma que estejam em conformidade com os padrões internacionais referidos na Subseção 2.1.2. Levanta-se, também, a questão de como implementar-se o ciclo de vida recomendado pelas entidades padronizadoras de IIoT nas redes LoraWan, a qual não foi originalmente pensada para o cenário industrial.

Esses problemas podem ter possíveis consequências sensíveis a ambientes como os de IIoT, além de limitarem as possibilidades de crescimento e competitividade geradas com a IIoT em geral. Em um cenário em que a sua quantidade é expressiva, a gestão indevida/maliciosa dos dispositivos pode fazer com que dispositivos autênticos não sejam bem utilizados e dispositivos não autênticos possam ingressar na rede. As consequências da não implementação de um ciclo de vida de credenciais organizado pode comprometer a segurança do sistema como um todo, abrindo a possibilidade de dispositivos com chaves comprometidas continuarem a integrar a rede e comprometer os dados enviados à camada empresarial ou, ainda, possibilitar ataques de interceptação de informações.

Os critérios de avaliação para verificar-se se existem, na literatura, soluções a esse problema devem ser pautados em basicamente quatro questões motivadoras:

- Q1: Propõe a utilização um ciclo de vida para garantir a segurança do controle de acesso de dispositivos?

- Q2: Propõe uma solução para o gerenciamento de credenciais em ambientes com requisitos de ampla área com configuração remota?
- Q3: Utiliza o protocolo LPWAN LoraWan na comunicação?
- Q4: Voltado a contextos industriais, ou seja, a IIoT?

Partindo-se desses critérios e métricas, pode-se avaliar se o problema é resolvido ou não, os quais servirão de base para compararem-se os trabalhos relacionados identificados na revisão bibliográfica.

2.5 TRABALHOS RELACIONADOS

A presente seção tem como objetivo levantar os trabalhos relacionados com o problema da pesquisa, e, nesse sentido, busca-se também analisar tendências em relação aos problemas propostos. Esse tipo de mapeamento também tem como finalidade identificar lacunas em quantidade de publicações em determinado assunto por meio da produção de uma classificação estruturada. Tem foco na coleta de dados, categorização e frequência de publicação e visa produzir gráficos para o melhor entendimento das tendências de publicação nessas áreas.

2.5.1 Termos de pesquisa definidos

O fraseamento, com o estabelecimento de palavras-chave, foi estabelecido para a seleção dos trabalhos usando-se os termos: `(Industrial Internet of Things OR IIoT) AND LoraWan AND (Control Access OR Identity Management OR life cycle management)`.

2.5.2 Seleção

O processo de seleção é uma das principais etapas para um levantamento de trabalhos relacionados adequado, levando-se em consideração a recomendação em (GALVÃO; PANSANI; HARRAD, 2015). Há nele quatro fases relevantes:

- Identificação: nesta etapa da seleção, são elencados o número de relatos identificados nos banco de dados de buscas e também os relatos identificados em outras fontes;
- Seleção: aqui, primeiramente, são excluídos os trabalhos idênticos, mesmo que coletados em mecanismos de busca diferentes ou até mesmo publicados em veículos diferentes. Na seleção, também já é possível obter-se um número inicial de trabalhos elegíveis à próxima etapa e os trabalhos já excluídos (idênticos);

- Elegibilidade: este mapeamento dedica-se a trabalhos relativos à última década (2011-2022). O seu foco dar-se-á em artigos primários e secundários, podendo ser um *full* ou *short paper*. Os trabalhos poderão ser tanto de língua portuguesa quanto de língua inglesa; e
- Inclusão: depois de coletados os trabalhos, levando-se em conta as especificidades mencionadas na elegibilidade, serão elencados critérios de inclusão e exclusão, a fim de elaborar-se uma tabela comparativa com os trabalhos e obtendo-se, por fim, o número de estudos incluídos em síntese quantitativa.

2.5.3 Fontes

As fontes de busca do mapeamento serão feitas em Mecanismo de Busca Acadêmica (MBA) que atendam aos seguintes critérios: mecanismos de busca voltados à tecnologia da informação e engenharias ou mecanismos de busca gerais, dentre os quais, os selecionados foram IEEE Xplore, Science Direct e o Science.gov, devido à sua disponibilidade de recursos como filtros avançados e melhores visualizações dos dados.

2.5.4 Análise

Para a análise dos trabalhos que passaram da primeira etapa de seleção, elegendo os critérios de inclusão e exclusão.

Critérios de Inclusão (CI):

- **Critério de Inclusão 01 (CI01)**: é um trabalho da última década, ou seja, publicado entre 2012 e 2022;
- **Critério de Inclusão 02 (CI02)**: o trabalho aborda o controle de acesso em redes LoraWan para IoT e/ou IIoT;
- **Critério de Inclusão 03 (CI03)**: o trabalho concentra-se na gestão de identidade e credenciais de dispositivos em redes LoraWan; e
- **Critério de Inclusão 04 (CI04)**: o trabalho avalia a autenticação de dispositivos industriais (IIoT).

Critérios de Exclusão (CE):

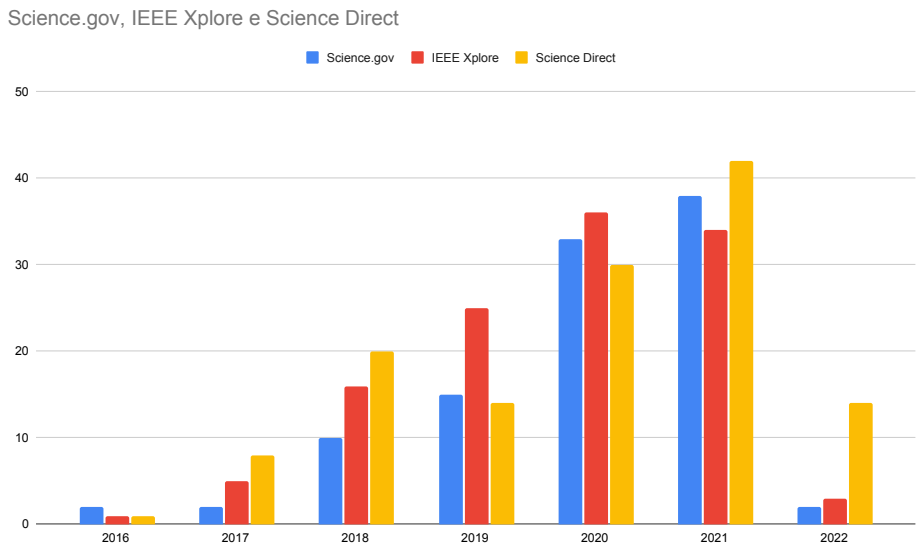
- **Critério de Exclusão 01 (CE01)**: o trabalho é publicado em revistas comerciais ou não acadêmicas, ou ainda não foi revisado;

- **Critério de Exclusão 02 (CE02)**: o trabalho foi publicado há mais de uma década, ou seja, é anterior a 2012;
- **Critério de Exclusão 03 (CE03)**: o trabalho não aborda, ou aborda apenas indiretamente, o tema de gerenciamento de credenciais no LoraWan; e
- **Critério de Exclusão 04 (CE04)**: o trabalho não aborda, ou aborda indiretamente, a utilização de dispositivos IoT ou IIoT.

2.5.5 Execução

Na busca geral dos mecanismos eleitos como fontes, foram encontrados um total de 332 artigos que atendem à pesquisa usando o fraseamento definido. A listagem do quantitativo relacionado a cada mecanismo de busca está na Figura 10.

Figura 10 – Gráfico de artigos encontrados nos mecanismos de busca, por ano.



Fonte: O autor.

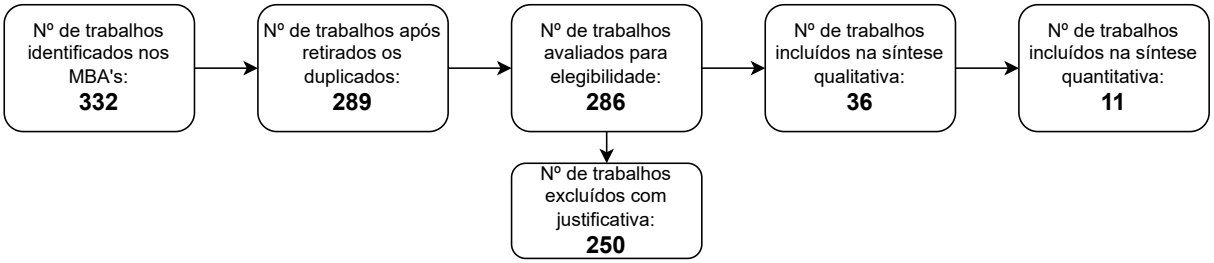
Como a Figura 10 indica, a maior parte dos trabalhos foi publicada entre 2018 e 2021, apresentando uma tendência de aumento no número de publicações nos últimos anos até 2021, o que assevera que o tema está em ascensão. Essa tendência de crescimento nas publicações permeia todos os mecanismos de buscas utilizados, embora faça-se uma ressalva de que, como a presente pesquisa foi realizada no ano de 2022, os trabalhos desse ano não tenham sido todos ainda contabilizados.

2.5.6 Elegibilidade

Para determinar-se a elegibilidade dos trabalhos, foram aplicados os passos de filtragem dos trabalhos, tais como como retirarem-se os trabalhos repetidos e aplicarem-se

os critérios de inclusão (CI) e os critérios de exclusão (CE). A Figura 11 relaciona o processo, desde a identificação inicial dos trabalhos nos MBAs até a síntese final.

Figura 11 – Fluxograma do processo de avaliação dos trabalhos.



Fonte: O autor.

Conforme a Figura 11, os trabalhos passam por um filtro de remoção de trabalhos duplicados e, posteriormente, são submetidos à avaliação de elegibilidade de acordo com os critérios de inclusão e exclusão. Então, os trabalhos são incluídos numa síntese qualitativa, cujo conteúdo do resumo e principais tópicos da publicação são avaliados. Assim, chega-se ao número de trabalhos incluídos na síntese quantitativa final.

2.5.7 Agregação

Neste MBA, chegou-se à apreciação de 11 trabalhos que se tornaram elegíveis à pesquisa. A Tabela 9 lista os trabalhos categorizadas por autor e ano.

Tabela 9 – Trabalhos relacionados identificados.

	Q1: propõe ciclo de vida de credenciais?	Q2: requisitos de ampla área e configuração remota?	Q3: utiliza redes LPWAN e protocolo LoraWan?	Q4: é voltado a cenários industriais IIoT?
(RIBEIRO et al., 2020)	Sim	Não	Sim	Não
(SANCHEZ-IBORRA et al., 2018)	Sim	Não	Sim	Não
(NAOUI; ELHDHILI; SAIDANE, 2016)	Sim	Não	Sim	Não
(RALAMBOTIANA, 2018)	Sim	Não	Sim	Não
(MCPHERSON; IRVINE, 2020)	Sim	Sim	Sim	Não
(WANG, 2021)	Sim	Sim	Sim	Não
(NAOUI; ELHDHILI; SAIDANE, 2017)	Não	Não	Sim	Sim
(XING et al., 2019)	Sim	Não	Sim	Não
(SANCHEZ-GOMEZ et al., 2020)	Sim	Sim	Não	Não
(FELLI; GIULIANO, 2021)	Sim	Sim	Sim	Não

Fonte: O autor.

Conforme observado na Tabela 9, na pesquisa de trabalhos relacionados, não houve trabalhos que preencham todos os requisitos derivados do problema de pesquisa colocado neste trabalho, de forma que a proposta é lastreada nos requisitos levantados no problema e não resolvidos nos trabalhos relacionados. Além disso, relaciona-se a qual versão do protocolo essas propostas levantadas pertencem.

2.5.8 Discussão

Os trabalhos relacionados para discussão apresentam similaridades com a proposta desta pesquisa e são analisados de acordo com os critérios advindos do problema de pesquisa. Também é realizada uma síntese dos objetivos e contribuições dos trabalhos:

1. (RIBEIRO et al., 2020) O trabalho é uma proposta de arquitetura de chaves baseada em contratos inteligentes e *blockchain* autorizado para aumentar a segurança de redes LoraWan. É realizada uma análise de desempenho com um protótipo que apresentou tempo de execução e valores de latência semelhantes quando comparado com um sistema tradicional. Os autores ainda discutem a utilização em ambientes com um grande número de dispositivos finais. O trabalho não faz qualquer menção ao contexto de IIoT e não aborda os requisitos de área ampla e configuração remota;

2. (SANCHEZ-IBORRA et al., 2018) O trabalho tem por objetivo o aprimoramento da segurança do LoraWan por meio de uma abordagem de gerenciamento de chaves leve e autenticada, o que busca trazer flexibilidade quando uma atualização de chave é necessária. Além disso, avaliaram-se as vulnerabilidades de segurança do LoraWan na área de gerenciamento de chaves, e o presente trabalho propõe diferentes esquemas alternativos. Ele também não menciona IIoT e não aborda os requisitos de área ampla e configuração remota;

3. (NAOUI; ELHDHILI; SAIDANE, 2016) O trabalho investiga os protocolos de gerenciamento de chaves existentes propostos para a IoT e, em seguida, propõe uma solução para aumentar a segurança da arquitetura LoraWan da IoT, solução essa inspirada numa de gerenciamento de chaves existente que usa nós *proxy* para aliviar a computação no lado do nó restrito;

4. (RALAMBOTIANA, 2018) Neste trabalho, o protocolo LoraWan e o gerenciamento de chaves em geral foram analisados para encontrarem-se os possíveis ataques por meio da exploração das suas vulnerabilidades através de uma árvore de ataques, os quais foram simulados para definirem-se as suas consequências no sistema. De acordo com eles, foram propostas melhorias de segurança na arquitetura com base em trabalhos anteriores sobre o tema e a exploração de possíveis contramedidas. Foi observado que o LoraWan ainda está em estágio inicial e tem sido usado principalmente em casos em que o servidor de rede gerenciava as chaves, garantindo a confidencialidade e a integridade dos dados. O trabalho ainda levantou a questão dos conflitos de interesse no caso em que a operadora de rede e o provedor de aplicativos são duas entidades

distintas. Para manter-se a confidencialidade, foi proposto um gerenciamento de chaves com uma *third party* confiável usando o protocolo LoraWan, o que resultou num caso de estudo para segurança *end-to-end*;

5. (MCPHERSON; IRVINE, 2020) Este trabalho aponta uma barreira a implantação de redes LPWAN como LoraWan que concerne aos problemas causados por interrupção dos serviços de rede para reconfiguração segura de dispositivos. Ele utiliza-se de *smartphones* para a transferência de credenciais de forma segura aos dispositivos, personalizando, assim, o esquema de distribuição das credenciais;

6. (WANG, 2021) O trabalho levanta que a versão mais recente do LoraWan v.1.1 forneceu uma estrutura de segurança que inclui proteção de confidencialidade de dados, verificação de integridade de dados, autenticação de dispositivos e gerenciamento de chaves. No entanto, os autores argumentam que a sua parte de gerenciamento de chaves é definida apenas de forma ambígua, e um esquema completo de gerenciamento de chaves é proposto para o LoraWan. O esquema proposto pelos autores aborda a atualização de chave, a geração de chave, o *backup* de chave e a compatibilidade com versões anteriores de chave. O esquema proposto foi desenvolvido não apenas para aprimorar o atual padrão LoraWan, mas também para atender o requisito mais básico do LoraWan, a saber, o baixo consumo de energia;

7. (NAOUI; ELHDHILI; SAIDANE, 2017) Este estudo tem como objetivo desenvolver um mecanismo de segurança para melhorar a resiliência da segurança de dados em aplicações LoraWan que utilizam servidores de rede de terceiros. Uma avaliação de risco foi executada para demonstrarem-se os perigos de servidores de rede de terceiros na segurança de dados. Com a constatação dos riscos, foi criada uma solução que criptografa toda a carga útil pela rede, que pode ser implementada por meio de uma biblioteca criada pelos autores. A solução pode ser implementada em contextos que incluem o industrial;

8. (XING et al., 2019) No trabalho, os autores levantam que não há nenhum mecanismo de gerenciamento e atualização de chaves definido para LoRa e propõem um esquema aprimorado de gerenciamento de chave segura para o sistema de comunicação LoRa para distribuir e atualizar a chave estática remotamente. Na proposta, o dispositivo e o servidor usam a carteira Hierarchical Deterministic (HD) para o gerenciamento de chaves. Eles realizam o acordo de chave por meio do algoritmo de troca de chave *Elliptic Curve Diffie-Hellman* (ECDH) e atualizam periodicamente a chave estática remotamente. Além disso, o algoritmo

Hash bidirecional é usado para minimizar o custo computacional e a sobrecarga de comunicação;

9. (SANCHEZ-GOMEZ et al., 2020) O trabalho inicialmente analisa o processo de *bootstrap*, que é necessário para estabelecer uma troca de dados segura entre dispositivos IoT e plataformas orientadas por dados, o que envolve, entre outras etapas, a autenticação, a autorização e o gerenciamento de credenciais. No entanto, argumentam os autores, há poucos esforços dedicados a fornecer a autenticação de acesso ao serviço na área de dispositivos IoT restritos conectados a redes sem fio recentes, como IoT de banda estreita NB-IoT e 5G. O trabalho apresenta a adaptação de protocolos de *bootstrapping* para serem compatíveis com as especificações 3GPP a fim de habilitar o recurso 5G de autenticação secundária para dispositivos IoT restritos. A solução proposta é avaliada para comprovar-se a sua eficiência e viabilidade, sendo um dos primeiros esforços para dar suporte à autenticação segura de serviços e estabelecimento de chaves a dispositivos IoT restritos em ambientes 5G; e
10. (FELLI; GIULIANO, 2021) O trabalho lida com o problema relacionado ao controle de acesso em áreas remotas com redes LoraWan. Os autores levantam o uso de memória e processamento exigidos por sensores locais para processos de autenticação e autorização e propõem um sistema capaz de verificar a acessibilidade de pessoas ou veículos em áreas restritas por meio da tecnologia LoraWan e registrar os dados relevantes usando um Ethereum Blockchain privado. Para isso, foi implementado um ambiente de teste numa floresta testando-se e avaliando-se os potenciais da tecnologia LoRa e Ethereum em *hardware* limitado. O banco de ensaio é montado numa área rural com diferentes densidades e alturas de árvores, o que foi levado em consideração durante a campanha de medição.

Assim, analisando-se os trabalhos identificados, percebe-se a existência do problema e também a ausência de trabalhos que o resolvam, e faz-se relevante a proposição de uma solução que o atenda e também a definição dos requisitos que delimitem a ele e ao cenário de aplicação.

2.6 CONSIDERAÇÕES DO CAPÍTULO

Neste capítulo, apresentam-se conceitos fundamentais ao entendimento do cenário no qual se encontra a problemática que o trabalho pretende resolver. Os conceitos elementares da IIoT, a sua arquitetura e a sua segurança permitem que se situem os principais tópicos para o entendimento desse conceito emergente. De maneira mais específica, este capítulo apresenta as práticas recomendadas no que tange

a gestão de identidade e o controle de acesso, para, então, analisar os ciclos de vida de identidades ideal e recomendado para contextos IIoT.

A discussão acerca das tecnologias de comunicação para IIoT compara as diversas tecnologias disponíveis como meio de transmissão não guiado para ligar os dispositivos aos *gateways* da camada de borda. Tendo em vista os requisitos do cenário estudado de monitoramento de processos de ampla área, para além do custo e da flexibilidade, o Lora e o LoraWan são designados como o melhor meio para compor-se o cenário IIoT estudado. Partindo-se, então, da seleção dessas tecnologias, são analisados os seus principais tópicos em segurança, e, sem seguida, apresentam-se os tópicos relativos ao controle de acesso e gestão de identidades no protocolo.

Com os conceitos fundamentais de arquitetura, segurança e controle de acesso tanto de IIoT quanto do LoraWan possibilitam relacionar os problemas ligados ao ciclo de vida de identidades e credenciais neste cenário concreto. Além disso, os trabalhos relacionados justificam uma proposta para a resolução dos problemas fundamentados e evidenciados neste capítulo. As considerações inerentes à proposta de resolução do problema podem ser encontradas no Capítulo 3.

3 PROPOSTA & AMBIENTE DE TESTES

A presente proposta de análise se estrutura em critérios empíricos, que são compilados em uma proposta coesa que serve de base para os testes e experimentos. Além disto, a presente proposta engloba a definição do ambiente de experimentação, os planos de testes baseados nos critérios de análise, a arquitetura do ambiente de teste bem como considerações parciais.

Neste capítulo, a Seção 3.1 estabelece critérios da análise e a Seção 3.2 faz as proposições gerais deste trabalho. Os critérios são uma peça importante para a guiar a consolidação da proposta. A Seção 3.3 faz um detalhamento do cenário de testes da proposta. A arquitetura do ambiente na Seção 3.4 faz um detalhamento da arquitetura e por fim, os planos de testes na Seção 3.5 concretizam a testagem dos itens propostos para análise. O ambiente, a arquitetura e os planos de testes são as bases para a concretização das testagens necessárias para guiar os experimentos.

3.1 CRITÉRIOS DE ANÁLISE

A principal proposição desta pesquisa é realizar uma análise do ciclo de vida de credenciais em redes LPWAN LoraWan 1.1 de acordo com os padrões recomendados para IIoT. Para concretização desta análise, elabora-se critérios que, a partir de testes e experimentos para indicar possíveis resoluções para o problema apontado por este trabalho. Cada critério é fundamentado nos padrões de gestão de identidades:

- Critério 1 (C1): Fases do ciclo de vida das credenciais: Identificar as fases do ciclo de vida das credenciais de uma rede LoraWan implementada com servidores ChirpStack para IIoT. Verificar se os procedimentos estão categorizados em fases de registro, gerenciamento e autenticação. Conforme os padrões de gerenciamento de identidades descrito por (SCHRECKER et al., 2016) baseados em normas.

- Critério 2 (C2): Etapas do ciclo de vida das credenciais: Identificar quais etapas estão envolvidas nas fases do ciclo de vida de identidades do contexto LoraWan ChirpStack. A fase de registro deve conter etapas de iniciação, verificação e registro de entidades. A fase de gerenciamento deve agregar a geração, vinculação, emissão e armazenamento de credenciais. Além disto, a fase de gerenciamento deve abranger medidas de contingência como suspensão, renovação e substituição de credenciais. De acordo com padrão estabelecido por (SCHRECKER et al., 2016).

- Critério 3 (C3): Mecanismos de auditoria: Verificar, no contexto LoraWan ChirpStack, os mecanismos de auditoria no ciclo de vida de credenciais. O processo de auditoria é uma etapa presente em todas as fases do ciclo de vida, conforme (SCHRECKER et al., 2016). Neste critério é avaliado se a auditoria neste cenário transpassa estas fases.

Estes aspectos e critérios permitem concretizar um escopo para a proposta de análise deste trabalho. A Seção 3.2 descreve a proposta para a resolução das problemáticas do trabalho e o seu cenário de aplicabilidade é delimitado.

3.2 PROPOSTA

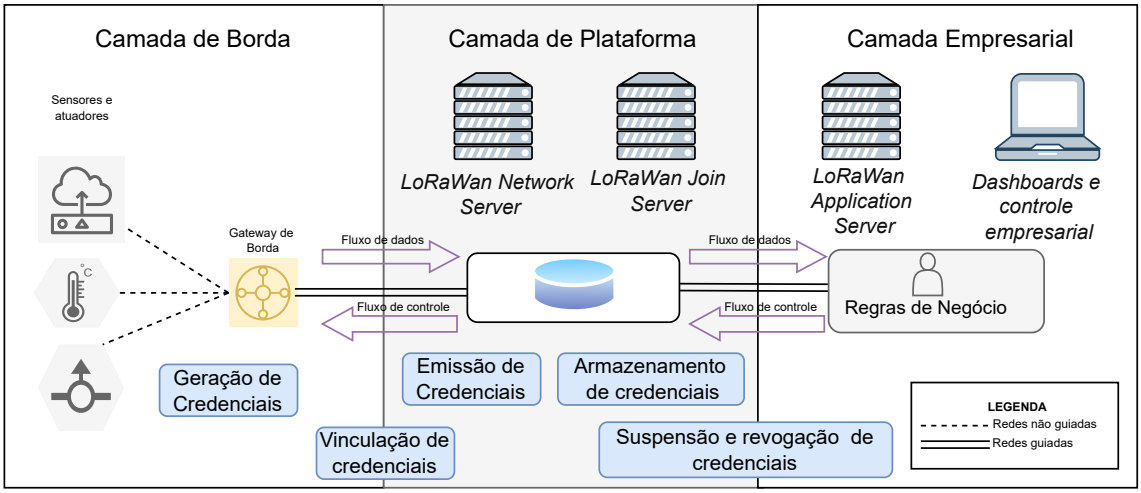
A proposta consiste em uma análise do ciclo de vida de credenciais para redes privadas de redes LPWAN, implementadas com com o protocolo LoraWan com servidores usando o componente ChirpStack. O cenário se baseia em uma arquitetura de camadas para IIoT. Para tanto, o cenário segue o padrão proposto em (LIN et al., 2017), descrito no Capítulo 2.

Neste contexto, delimitando a categorização por camadas atribui a Camada de Borda, os dispositivos finais, sensores e atuadores. Estes componentes da arquitetura são responsáveis por fornecer as credenciais usadas para a emissão das credenciais, bem como realizar sua vinculação. Já no contexto LoraWan, isto se dá com o dispositivo final fornecendo sua credencial ao processo de *join procedure*.

Para mais, a etapa de vinculação entre a credencial e a entidade do IAM é realizada em uma operação entre a camada de borda e de plataforma. No LoraWan, a vinculação acontece no procedimento de *join procedure*, antes do envio da mensagem de *join accept*. Outrossim, a emissão e armazenamento das credenciais ocorre na camada de plataforma da arquitetura referencial, responsável pelas operações de infraestrutura e rede. Em termos do protocolo LoraWan as entidades envolvidas são o *Join Server* e o servidor de rede. As operações são realizadas após o procedimento de *join procedure*.

Por fim, a suspensão e revogação de credenciais é uma operação que envolve duas camadas da arquitetura, a camada de plataforma e empresarial. Em paralelo, no LoraWan, esta operação pode ser acionada pelo *Join Server* ou pelo servidor de aplicação LoraWan através de sistemas de controle ou *dashboards*. A arquitetura em questão pode ser observada na Figura 12.

Figura 12 – Arquitetura do LoraWan usando IIoT.



Fonte: O autor.

Conforme ilustrado na Figura 12 a arquitetura referencial genérica de IIoT pode servir como uma estruturação para redes LoraWan no contexto industrial. Este cenário de análise é justificado pelos requisitos em serviços de IIoT. A implementação de redes LoraWan está diretamente ligada aos requisitos de caso de uso, número de dispositivos e área típica de serviço conforme (BROWN et al., 2018). Em concomitância, a utilização dos componentes para ChirpStack para servidores LoraWan é justificado por ser largamente utilizado, ter a maior comunidade ativa e ter atualizações periódicas e atuais (LUND, 2022).

O domínio de aplicação da proposta pode ser definido como um contexto industrial com uma grande área de serviço, disponibilidade de baixa potência e gerenciamento remoto. Entretanto, o correto gerenciamento de segurança no cenário desta proposta aponta para o problema apontado neste trabalho na Seção 2.4.

Esta problemática tem especial importância para uma aplicação segura do protocolo LoraWan em contextos industriais, de acordo com a arquitetura descrita. Principalmente, no que se refere a mitigação de vulnerabilidades, em contextos da internet industrial. As vulnerabilidades relativas ao processo de autenticação, especialmente em relação ao ciclo de vida de credenciais. É especificamente este contexto que a proposta visa atender, e estas vulnerabilidades que a proposta visa identificar e analisar. Dentro deste contexto computacionais são definidos as tecnologias utilizadas, a arquitetura do ambiente no qual são realizados os testes da proposta.

3.3 AMBIENTE DE EXPERIMENTAÇÃO

Os conceitos e definições relacionados ao ambiente de experimentação são apresentados nesta seção. Estes tem especial importância para definição do contexto computacional em que são realizados os testes. Ainda, tem relevância para estabelecer a correlação dos testes com o ambiente real, além de pontos de medição e os meios de comunicação.

O ambiente de experimentação, de forma geral é simulado em um sistema operacional GNU/Linux Ubuntu, versão 18.04. Neste contexto, os componentes da arquitetura são simulados em consonância com a Figura 12, seguindo uma estrutura de camadas. A camada de borda conta com dispositivos finais simulados, especificamente medidores de temperatura. Esta camada de borda ainda conta com o *gateway* LoraWan simulado. Ambos os componentes são simulados em um ecossistema ChirpStack versão 3.16.3. Versão lançada em julho de 2021. Estes componentes são equivalentes aos dispositivos finais do chão de fábrica do cenário industrial, e o *gateway* equivalente a um *gateway* físico. Estes componentes são simulados sob um computador com hardware integrado por, processador Intel Core I3 64bits, 8GB de memória DDR4 e disco rígido de 1TB. Os servidores são instalados sob um servidor *web* Apache, banco de dados Postgresql Server diretamente no sistema operacional nativo. Sem utilização tecnologias de virtualização, e.g., máquinas virtuais.

Na camada de plataforma, o ambiente conta com um servidor de rede e um *join server* LoraWan 1.1, ambos implementados também pelo ecossistema ChirpStack. O servidor de rede e o *join server* são os mesmos usados em ambientes reais, portanto bem próximos aos que são usados em ambiente fabril real.

Na camada empresarial, são utilizados o servidor de aplicação e a API também do ChirpStack. O servidor de aplicação e a interface de programação também são idênticos aos utilizados no contexto real. O gerenciamento da API é realizado pelo o cliente de API Insomnia[1], em sua versão 2022.2.1.

Com o detalhamento do ambiente de experimentação, é garantida a reprodutibilidade dos experimentos. Com isto, se faz possível o detalhamento da arquitetura de testes, bem como do plano concreto de testes relativos a análise. Na Subseção 3.4 é apresentada a arquitetura construída para a elaboração dos testes e a Subseção 3.5 o plano de testes.

[1] Informações em: <https://insomnia.rest/>

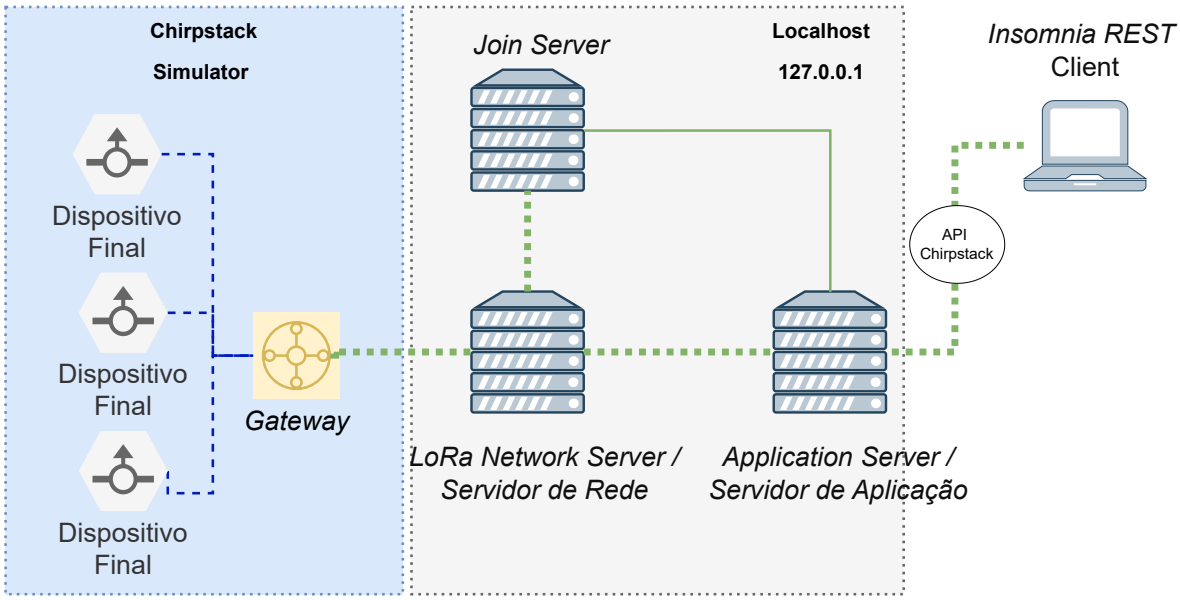
3.4 ARQUITETURA DO AMBIENTE DE TESTE

Conforme especificado no ambiente de experimentação, o mesmo foi construído sob a plataforma ChirpStack. A abordagem foi escolhida por permitir a implementação de uma arquitetura Lorawan em ambiente local, controlado, de forma simples, com custo e utilização de recursos reduzidos.

A arquitetura tem os componentes básicos de uma rede LoraWan: dispositivos finais, *join server*, *network server*, e *application server*. A arquitetura usada no ambiente de teste é ilustrada na Figura 12.

Nesta arquitetura, na qual são ambientados os testes, tem-se que os componentes da infraestrutura são implementados por um ambiente ChirpStack. No qual os dispositivos finais e *gateways* são simulações que interagem com um *Network Server*, um *Join Sever* e um *Application Server* convencional instalado em um servidor local. Este arranjo é ilustrado na Figura 13.

Figura 13 – Entidades simuladas e instaladas localmente no experimento.



Fonte: O autor.

Nesta arquitetura, não foram implementadas *dashboards*, softwares de visualização ou controle empresarial.

Seguindo esta mesma arquitetura, houve a utilização de um componente para simulação de dispositivos e *gateways*. Este é um simulador de código aberto para o servidor de rede LoRaWan ChirpStack. Simula um número configurável de dispositivos e gateways, que serão criados automaticamente ao iniciar a simulação. O simulador é disponibilizado em um respositório de um servidor GIT [2].

[2] https://github.com/brocaar/chirpstack-simulator

Nestes testes o servidor é inicializado, e tem uma duração da simulação configurada. Na sequencia, é possibilitada a leitura de métricas e a conclusão da simulação. Após esta duração, o simulador pode ser encerrado e os dispositivos, *gateways*, aplicativos e perfil de dispositivo criados são reiniciados.

No ambiente de experimentação, além dos componentes reais como Servidor de Rede, Sevidor de Aplicação e componentes do ChirpStack, o uso do simulador possibilita o teste com dispositivos da versão 1.1 do LoraWan. Para tanto, tanto os componentes reais, quantos os componentes simulados são agregados ao plano de testes. Possibilitando assim a observabilidade e reprodutibilidade dos experimentos.

Foram utilizadas as interfaces do servidor de aplicação e a API do ChirpStack para fazer a visualização e gerenciamento dos dispositivos e coleta de dados. Isto não interfere na relação com um cenário real, pois é estritamente igual ao funcionamento em ambiente de produção. Neste ambiente são realizados os testes descritos na Seção 3.5.

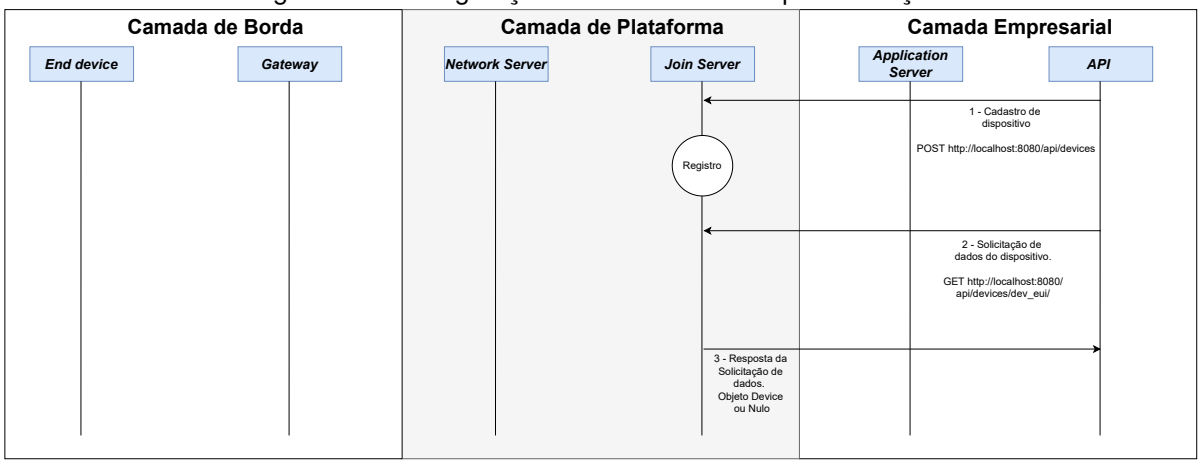
3.5 PLANO DE TESTES

De forma a realizar a análise do ciclo de vida de credenciais, foram levantados na Seção 3.1, critérios de análise. O plano de testes discutido, a partir dos critérios, são a referência para elaboração de cenários em que os testes são realizados. Ainda, ressalta-se que os experimentos seguem as definições do ambiente de experimentação e do ambiente de testes.

No plano de testes, cada cenário, é a base para os experimentos envolvidos na análise. Bem como, cada um deles possui um objetivo específico. Os cenários podem ter diferentes configurações pontuais de ambiente. Ainda, cada cenário é listado com seus métodos de coleta e análise de dados. Posteriormente aos testes nos cenários, os dados de retorno das operações do ciclo de vida são compilados e analisados.

- Cenário 1: Este cenário possui como configuração, um sensor dispositivo final e um *gateway* simulados. Ainda, utiliza do servidor de rede, do *Join Server* e do servidor de aplicação para realização das operações. A iniciação dos testes neste cenário se dá com o cadastro de um novo dispositivo no servidor de aplicação. Nesta iniciação é enviada a credencial DevEUI. Este processo de junção equivale aos processos de iniciação, verificação e registro do dispositivo. Por fim, faz-se a solicitação dos dados relativos ao dispositivo cadastrado, validando o sucesso ou fracasso da operação. Em consequência, a coleta de dados se dá pela respostas da realização desta operação de gerenciamento de credenciais pela API do ChirpStack. Estas operações são ilustradas na Figura 14.

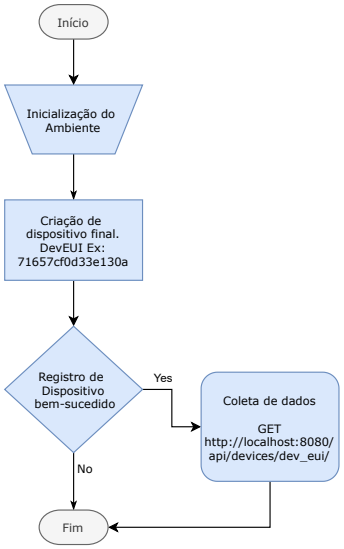
Figura 14 – Configuração do Cenário 1 de experimentação.



Fonte: O autor.

O objetivo desta experimentação é investigar se o protocolo LoraWan no cenário de testes implementa as etapas de iniciação, verificação e registro da fase de registro do ciclo de vida de credenciais. Portanto, este experimento visa avaliar os critérios C1 e C2. Isso para testar primeiramente as etapas e a fase de registro. A métrica do teste consiste no recebimento das chaves geradas na vinculação e o status positivo. Os passos do cenário são listados no fluxo da Figura 15.

Figura 15 – Fluxo do Cenário 1 de experimentação.



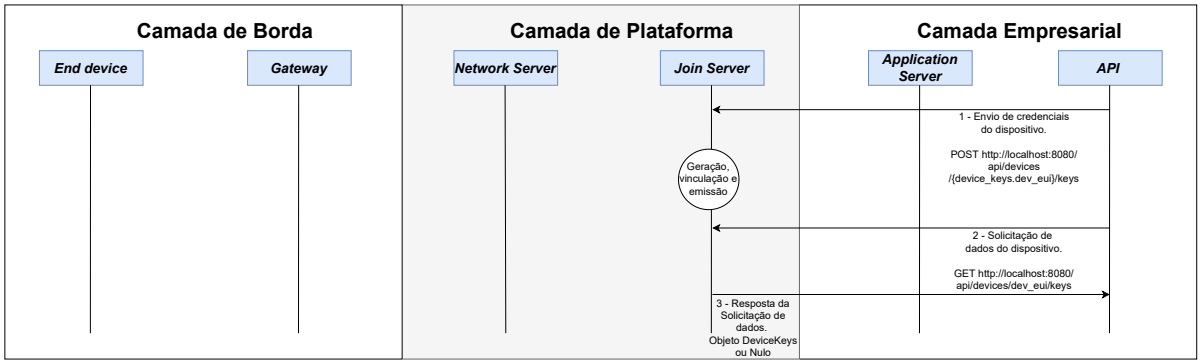
Fonte: O autor.

De acordo com a Figura 15 é possível verificar que a iniciação, verificação e registro de credenciais podem assumir dois estados diferentes ao depender do sucesso da operação. Posteriormente, a execução da solicitação dos dados valida o retorno do objeto. O resultado esperado é que o sistema responda com as credenciais do objeto *DeviceKeys*.

- Cenário 2: Este cenário possui como configuração, um sensor dispositivo final

e um *gateway* simulados. Ainda, utiliza do servidor de rede, do *join server* e do servidor de aplicação para realização das operações. A iniciação dos testes neste cenário se dá com um dispositivo já registrado no servidor de rede e aplicação. No teste, é realizado o envio de credenciais do dispositivo para geração, vinculação e emissão de novas credenciais do dispositivo nos servidores. Por fim, faz-se a solicitação das credenciais cadastradas para o dispositivo usado na operação. Esta operação tem como resposta um objeto denominado DeviceKeys em caso de sucesso ou nulo, Em consequência, a coleta de dados se dá pela respostas da realização desta operação de gerenciamento de credenciais pela API do ChirpStack. Estas operações são ilustradas na Figura 16.

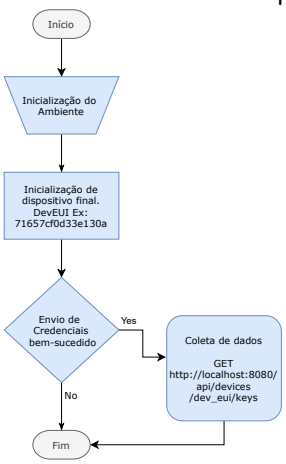
Figura 16 – Configuração do Cenário 2 de experimentação.



Fonte: O autor.

O objetivo desta experimentação é investigar se o protocolo LoraWan no cenário de testes implementa a fase de gerenciamento do ciclo de vida de credenciais. Concomitantemente, serve para testar a realização das etapas de geração, vinculação, emissão e armazenamento. Este experimento visa avaliar o critério C1 no que se refere a validação da fase de gerenciamento. Ademais, visa avaliar o critério C2 no que se refere as etapas de geração, vinculação e armazenamento. A métrica do teste consiste no recebimento das chaves de geradas no processo de registro do dispositivo. Isto indica que o dispositivo foi gerado, vinculado, emitido e armazenado. Os passos do cenário são ilustrados no fluxo da Figura 17.

Figura 17 – Fluxo do Cenário 2 de experimentação.



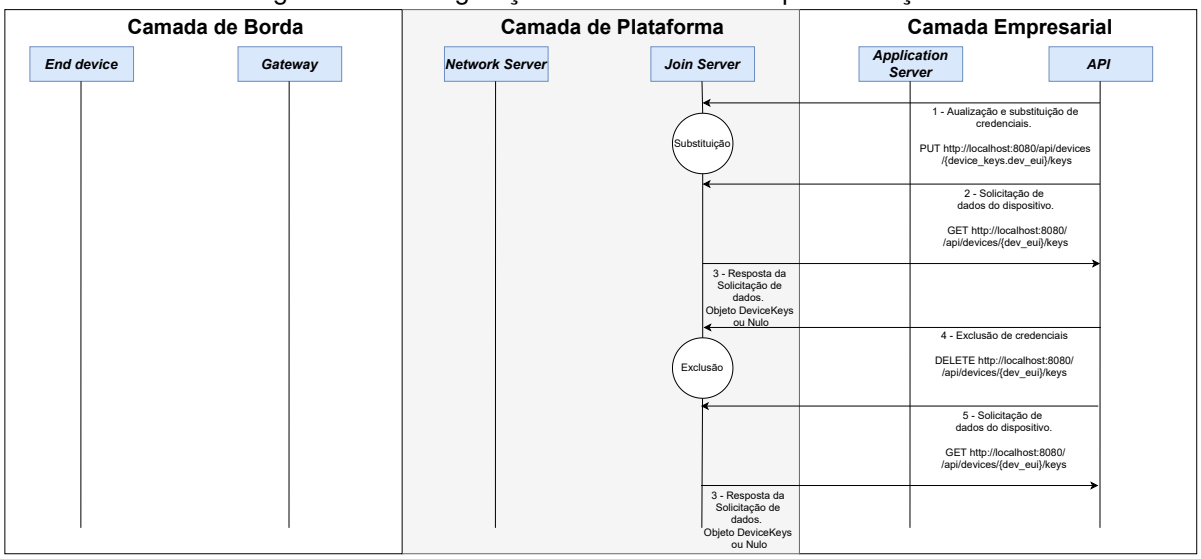
Fonte: O autor.

De acordo com a Figura 17 é possível verificar que o dispositivo podem assumir dois estados diferentes ao depender do seu processo de envio das credenciais para geração, vinculação emissão e armazenamento das mesmas. Posteriormente, a execução da solicitação das credenciais do dispositivo valida o retorno do objeto com as credenciais de sessão usadas na métrica. O resultado esperado é que o sistema responda com as credenciais do objeto *DeviceKeys*, com especificamente as credenciais Nwkkey e appKey.

- Cenário 3: Este cenário conta com um sensor dispositivo final e um *gateway* simulados. O mesmo, utiliza do servidor de rede, do *join server* e do servidor de aplicação para realização das operações. A iniciação dos testes neste cenário se dá com um dispositivo já registrado no servidor de rede e aplicação e que constam nos cenários C1 e C2 do plano de teste. Em primeiro lugar, neste cenário de teste, é realizado o envio de credenciais do dispositivo para substituição de credencias e exclusão de credenciais. O teste se dá em primeiramente enviando uma credencial ao *join server* e solicitando os dados das credenciais para conferência. Os dados esperados são atributos do objeto DeviceKeys em caso positivo, ou nulo. A finalidade é para atualização e substituição da credencial do dispositivo final usado no teste. Em segundo lugar, este teste faz a exclusão de uma credencial e também solicitando os dados das credenciais para conferência desta operação. Os dados esperados estão contidos no objeto DeviceKeys se a operação tiver sucesso ou nulo. A finalidade é fazer remoção desta credencial dos servidores. Haja vista os testes realizados, a coleta de dados se dá pela respostas da realização destas operações de gerenciamento de credenciais pela API do ChirpStack. Estas operações são ilustradas na Figura 18.

O objetivo desta experimentação é investigar se o protocolo LoraWan no cenário de testes implementa a fase de gerenciamento do ciclo de vida de credenciais.

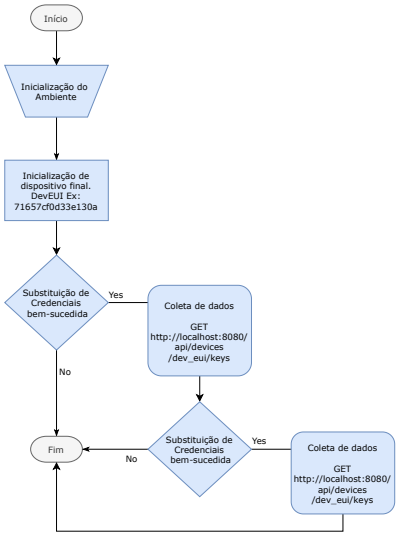
Figura 18 – Configuração do Cenário 3 de experimentação.



Fonte: O autor.

Assim como, serve para testar a realização das etapas de renovação, substituição, rotatividade, suspensão e revogação de credenciais. Este experimento visa avaliar o critério C1 no que se refere a validação da fase de gerenciamento. Ademais, visa avaliar o critério C2 no que se refere as etapas de renovação, substituição, rotatividade, suspensão e revogação de credenciais. A métrica do teste consiste no recebimento das chaves de geradas no processo de registro do dispositivo. Isto pode indicar que as credenciais do dispositivo tiveram exito ou falharam ao executar as operações. Os passos do cenário são ilustrados no fluxo da Figura 19.

Figura 19 – Fluxo do Cenário 3 de experimentação.



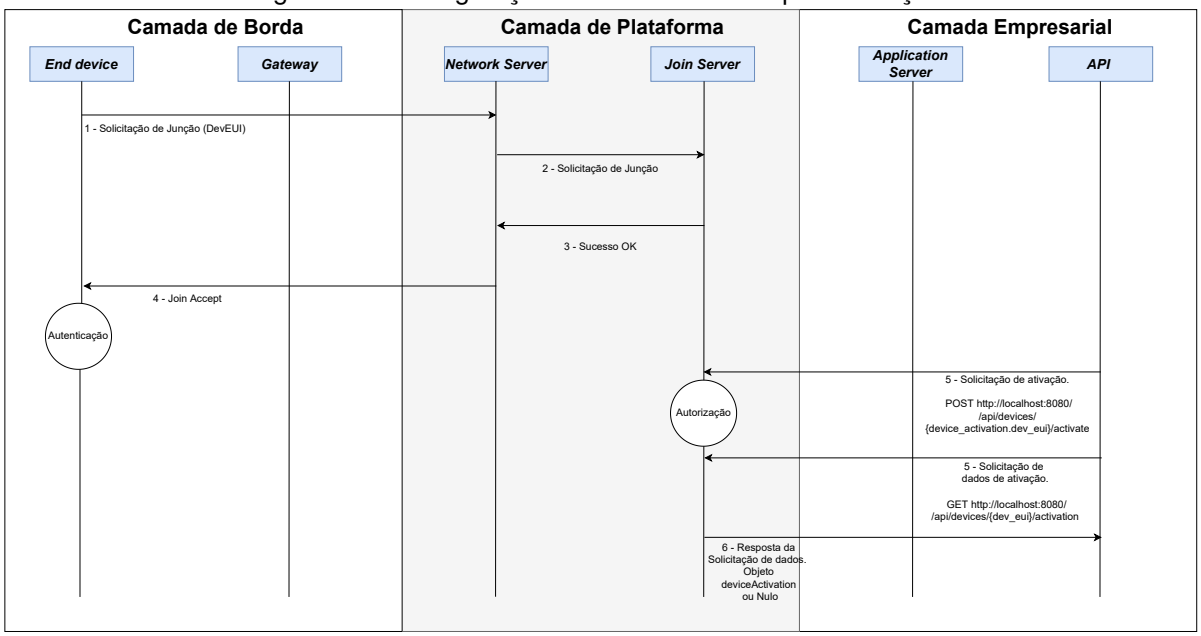
Fonte: O autor.

Analisando a Figura 19, percebe-se que é possível verificar que o dispositivo po-

dem assumir dois estados diferentes ao depender do seu processo de atualização das credenciais. Posteriormente, a execução da solicitação das credenciais do dispositivo valida o retorno do objeto com as credenciais de sessão usadas na métrica. Adiante, tendo a substituição retornado sucesso, é realizada o teste relativo a exclusão de credenciais, na sequência são solicitadas novamente as credenciais. O resultado esperado é que o sistema responda com as credenciais do objeto *DeviceKeys*, com especificamente as credenciais Nwkkey e appKey.

- Cenário 4: Este cenário (assim como Cenário 1, 2 e 3) conta com um sensor dispositivo final e um *gateway* simulados. O mesmo, utiliza do servidor de rede, do *join server* e do servidor de aplicação para realização das operações de junção e ativação. Este cenário visa testar os experimentos com base nos critérios de análise C1 e C2. O mesmo visa testar a adequação do protocolo objeto de estudo em relação a fase de autenticação de entidades. Relativo às fases, este cenário objetiva testar as etapas de autenticação e autorização. A iniciação dos testes neste cenário se dá com um dispositivo já registrado no servidor de rede e aplicação e com as credenciais do Cenário 1. Em primeiro lugar, neste cenário de teste, é realizado o procedimento de junção de um dispositivo no *network server*. A solicitação então é encaminhada ao *join server* que responde esta solicitação de junção que equivale a autenticação do dispositivo neste sistema. Após este processo o dispositivo está pronto para ser ativado. O processo de ativação se dá a partir de uma solicitação de ativação, no qual são enviadas informações inerentes ao dispositivo como contadores e suas credenciais de sessão. Em sequência, neste cenário é enviada uma solicitação dos dados inerentes a esta solicitação de ativação. A resposta desta solicitação serve como ponto de coleta de dados. Com estes testes realizados, a coleta de dados se dá pela respostas da realização destas operações de gerenciamento de credenciais. Neste caso especificamente as etapas de autenticação e autorização. Estas operações são ilustradas na Figura 20.

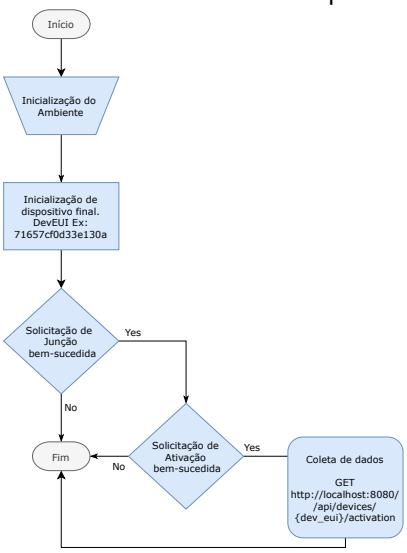
Figura 20 – Configuração do Cenário 4 de experimentação.



Fonte: O autor.

O objetivo desta experimentação é investigar se o protocolo LoraWan no cenário de testes implementa a fase de autenticação de entidades. Assim como, serve para testar a realização das etapas de autenticação e autorização de dispositivos usando as credenciais enviadas no experimento do Cenário 2. Este experimento visa avaliar o critério C1 no que se refere a validação da fase de autenticação. Ademais, visa avaliar o critério C2 no que se refere as etapas de autenticação e autorização. A métrica do teste consiste no recebimento do status de autenticação e autorização do dispositivo após a realização das operações. Isto pode indicar que as credenciais do dispositivo tiveram êxito ou falharam ao executar as operações. Os passos do cenário são ilustrados no fluxo da Figura 21.

Figura 21 – Fluxo do Cenário 4 de experimentação.

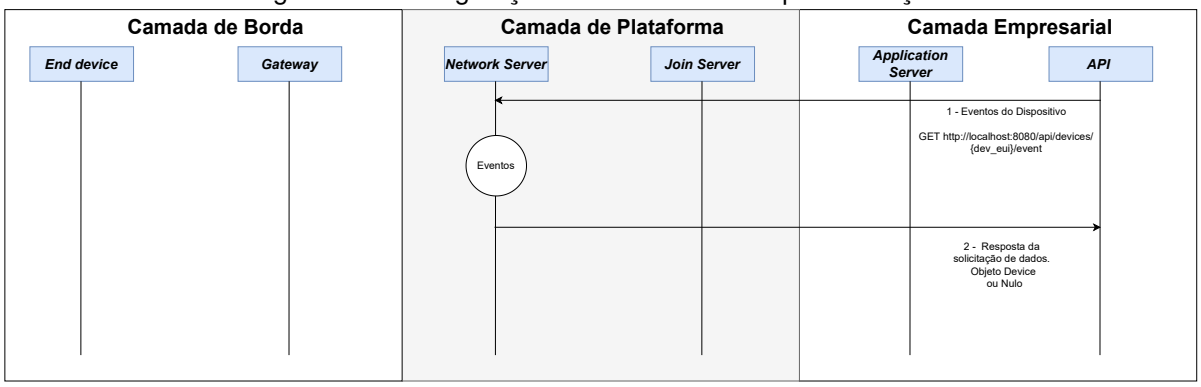


Fonte: O autor.

De acordo com a Figura 21 é possível verificar que após a inicialização do dispositivo e do ambiente é realizada a solicitação de junção e posteriormente a solicitação de ativação. Em primeiro lugar é realizada solicitação de junção, que retorna uma mensagem de sucesso ou falha. Este dado é coletado como uma primeira métrica para medição. Adiante, tendo a junção retornado sucesso, é realizado o teste relativo a autorização do dispositivo. Caso a junção não tenha sucesso, o teste é encerrado. Posteriormente, após a solicitação de junção bem sucedida, é realizado o teste de ativação. Caso a ativação não tenha sucesso o teste é também encerrado. Se a solicitação de ativação tem sucesso é feita uma coleta de dados do dispositivo para nova métrica e o experimento encerrado.

- Cenário 5: Este cenário possui como configuração, um sensor dispositivo final e um *gateway* simulados. Ainda, utiliza do servidor de rede, do *Join Server* e do servidor de aplicação para realização das operações. A iniciação dos testes neste cenário se dá com um dispositivo já registrado no servidor de rede e aplicação, conforme os Cenários 2, 3 e 4. No teste, é realizada uma solicitação de informações em relação aos eventos acontecidos no dispositivo. Estes eventos incluem cargas de *uplink*, ACKs, junções, erros. Por fim, é recebida a resposta da solicitação com as informações relativas aos eventos acontecidos no dispositivo. Em consequência, a coleta de dados se dá pela respostas da realização desta operação de auditoria pela API do ChirpStack. Estas operações são ilustradas na Figura 22.

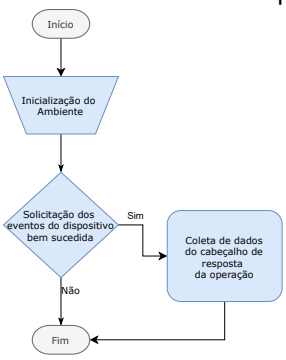
Figura 22 – Configuração do Cenário 5 de experimentação.



Fonte: O autor.

O objetivo desta experimentação é investigar se o protocolo LoraWan no cenário de testes implementa o processo de auditoria. Este processo deve estar de acordo com o padrão exposto em (SCHRECKER et al., 2016), que define o processo de auditoria como uma etapa transversal a todas as fases. Assim como, serve para testar a realização das etapas de autenticação e autorização de dispositivos usando as credenciais enviadas no experimento do Cenário 2. Este experimento visa avaliar o critério C1 no que se refere a validação da fase de autenticação. Ademais, visa avaliar o critério C2 no que se refere as etapas de autenticação e autorização. A métrica do teste consiste no recebimento do status de autenticação e autorização do dispositivo após a realização das operações. Isto pode indicar que as credenciais do dispositivo tiveram exito ou falharam ao executar as operações. Os passos do cenário são ilustrados no fluxo da Figura 23.

Figura 23 – Fluxo do Cenário 5 de experimentação.



Fonte: O autor.

De acordo com a Figura 23 é possível verificar que após inicializado o ambiente é realizada a solicitação dos eventos do dispositivo. Posteriormente, a execução da solicitação dos dados valida o retorno do objeto. O resultado esperado é que o sistema com o detalhamento de eventos como as cargas de *uplink*, ACKs, junções, erros.

Conforme observado nos cenários de testes, as fases e etapas do ciclo de vida de credenciais são testados de forma agrupada. Os mesmos são realizados com as operações disponíveis na API do servidor ChirpStack. Na Seção 3.6 os testes são observados a partir de suas categorização e características técnicas operacionais.

3.6 PLANO DE TESTES DE CONSOLIDADO

O plano de testes consolidado tem como objetivo sintetizar as informações relativas aos testes deste trabalho. As informações são categorizadas por meio de cada operação realizada na API do servidor ChirpStack.

Reforça-se que os critérios de análise são todos analisados neste ambiente de testes em seus respectivos cenários. Outrossim, os ambientes de testes são plausíveis para realização dos experimentos elencados pelo trabalho. Os cenários de testes consolidados são listados na Tabela 10.

Tabela 10 – Cenários de testes consolidados.

Fase	Etapa	Critério de Análise	Cenário	Operação HTTP	Endpoint da API ChirpStack
Registro	Iniciação	C1 e C2	Cenário 1	POST	/api/devices
Registro	verficação	C1 e C2	Cenário 1	POST	/api/devices
Registro	registro	C1 e C2	Cenário 1	POST	/api/devices
Gerenciamento de Credenciais	geração	C1 e C2	Cenário 2	POST	/api/devices/{device_keys.dev_eui}/keys
Gerenciamento de Credenciais	vinculação	C1 e C2	Cenário 2	POST	/api/devices/{device_keys.dev_eui}/keys
Gerenciamento de Credenciais	emissão	C1 e C2	Cenário 2	POST	/api/devices/{device_keys.dev_eui}/keys
Gerenciamento de Credenciais	armazenamento	C1 e C2	Cenário 2	GET	/api/devices/{dev_eui}/keys
Gerenciamento de Credenciais	renovação, substituição e rotatividade.	C1 e C2	Cenário 3	PUT	/api/devices/{device_keys.dev_eui}/keys
Gerenciamento de Credenciais	suspensão e revogação	C1 e C2	Cenário 3	DELETE	/api/devices/{dev_eui}/keys
Autenticação de Entidades	Autenticação	C1 e C2	Cenário 4	POST	/api/devices/{device_activation.dev_eui}/activate
Autenticação de Entidades	Autorização	C1 e C2	Cenário 4	POST	/api/devices/{device_activation.dev_eui}/activate
Autenticação de Entidades	Auditoria	C3	Cenário 5	GET	/api/devices/{dev_eui}/events

Fonte: O autor.

Na Tabela 10 foram sumarizados os cenários de testes com as informações relativas as fases e etapas do ciclo de vida. Além disto, são elencados quais critérios de análise são atendidos para cada operação de teste bem como qual cenário estas operações pertencem. Por fim, são especificadas que tipo de operações são realizadas e os respectivos *endpoints* utilizados.

3.7 CONSIDERAÇÕES PARCIAIS

Neste capítulo foram desenvolvidos aspectos da proposta e do ambiente de testes deste trabalho. Para o desenho da proposta, foram delineados critérios de análise na Seção 3.1 que servem de base para a concretização do analise objetivo do trabalho. Os critérios levantados dão base a uma análise relativa as fases, as etapas e os mecanismos de auditoria do ciclo de vida de credenciais.

A partir dos critérios de análise foi definida a proposta (Seção 3.2). A mesma consiste na análise do ciclo de vida de credenciais para redes privadas de redes LPWAN, implementadas com com o protocolo LoraWan com servidores usando o componente ChirpStack. Para melhor definição do ambiente de experimentação, a Seção 3.3 deu detalhes relacionados ao ambiente físico onde os experimentos são realizados. Detalhes do hardware do software utilizados, como versões capacidade e características são aprofundadas. Em sequência foi delineada a arquitetura de testes que cita os principais componentes da arquitetura que recebe viabiliza os testes. Esta arquitetura do ambiente de testes se desenvolve, Seção 3.4.

Com a fixação dos critérios de análise, o desenho da proposta, a designação do ambiente de experimentação e a arquitetura do ambiente de testes se tornou possível a criação do plano de testes (Seção 3.5). O plano de testes é estruturado em cenários de testes onde cada um conta com uma introdução do cenário com seus componentes e suas relações com os critérios de análise. Ademais, os cenários de testes contam com esquemas de configuração do cenário, ilustrando as operações desenvolvidas no cenário. Ainda, são expostos os objetivos, ponto de medição e métricas de testes. Por fim, nos cenários de análise, são ilustrados os fluxos dos experimentos e sua respectiva descrição com detalhes inerentes ao procedimento realizado nos testes. A Seção 3.6 foi realizada uma ajuntamento de informações com intenção de consolidar e compilar as informações relativas aos planos de testes.

Informações necessárias para a coleta, análise, interpretação e tratamento das informações coletadas para os experimentos que constam no Capítulo 4.

4 EXPERIMENTOS

Este capítulo explora os procedimentos realizados para concretização dos testes. A inicialização, execução e os resultados dos experimentos são analisados e descritos nesse capítulo. O objetivo, detalhes de implementação dos testes também são explicitados. Além disto, eventuais problemas encontrados são relatados.

O capítulo está organizado pelos cenários do plano de testes elaborado na Seção 3.5. Além disto, apresentam-se os resultados obtidos e os mesmos são discutidos, analisados e comparados na Seção 4.7.

4.1 INICIALIZAÇÃO DOS EXPERIMENTOS

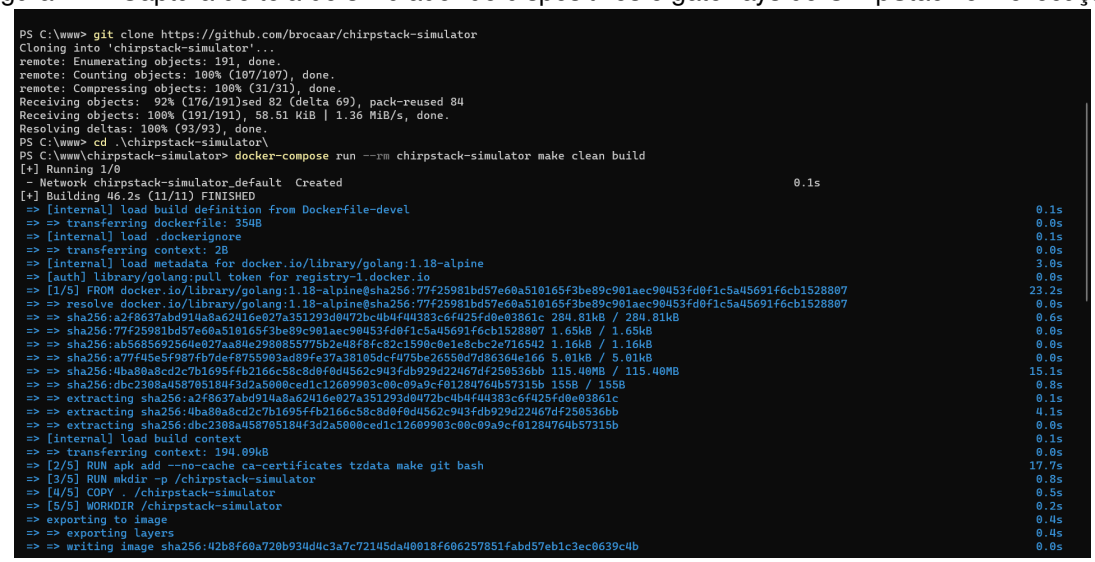
Em se tratando da inicialização do ambiente, foi necessária uma preparação e criação de variáveis necessárias para realização dos testes. Isto se deu inicialmente com o registro de um *Network Server*, de um perfil de serviço, de uma organização e uma aplicação. As especificações usadas nos registros são:

- *Network Server*: A configuração do *Network Server* é realizada inserindo o nome designado para esta entidade, além do endereço do *Network Server* instalado no sistema operacional. A instalação do *Network Server* serve de base para a criação dos perfis de serviço, organizações e aplicações.
- Perfil de Serviço (*Service Profile*): A configuração do *Network Server*, é usada como base para criação dos perfis de serviço. Os dados a serem introduzidos são inerentes a identificação do perfil, ao *Network Server* usado e a ativação de itens opcionais. Estes itens opcionais são relativos a ativação de meta dados do *gateway*, geolocalização da rede, taxa de dados mínima e máxima, além da possibilidade de usar um *gateway* privado apenas para um perfil de serviço.
- Organização (*Organization*): A configuração da organização é realizada inserindo o nome designado para esta organização bem como um nome de exibição. Organizações podem ter *gateways* suplementares adicionados por administradores do sistema, caso essa opção seja definida.
- Aplicação (*Application*): A criação de aplicação no servidor de aplicação é realizada inserindo o nome designado para a mesma bem como um nome de exibição. As aplicações devem usar um perfil de serviço cadastrado previamente.

Assim, com os registros realizados no ChipStack é possível inicializar o simulador, conforme a designação dos componentes reais e simulados descritos na Seção

3.4. Em primeiro lugar, o simulador é devidamente obtido do repositório em que se encontra disponibilizado. Posteriormente, acedendo a pasta do mesmo, é realizada uma instalação usando Docker Compose[1]. Com esta instalação os arquivos são devidamente descompactados tornando o programa apto para as configurações iniciais. Após a instalação, é realizada uma configuração em um arquivo de definições e a execução no console do sistema operacional. Esta execução (Figura 24) é realizada com o comando *./build/chirpstack-simulator -c chirpstack-simulator.toml*.

Figura 24 – Captura de tela do simulador de dispositivos e gateways do ChirpStack em execução.



Fonte: O autor.

Após a instalação, configuração e execução os dispositivos e *gateways* simulados são adicionados ao Application Server do ChirpStack. O ambiente, com os registros realizados, e com o simulador em execução fica pronto para efetuação dos primeiros testes. De acordo com cenários levantados no plano de testes da Seção 3.5 ambiente de experimentação.

4.2 CENÁRIO 1

O objetivo deste experimento é verificação de etapas e fases do ciclo de vida acontecem no cenário computacional deste trabalho. Este contexto é composto por uma rede LoraWan privada proporcionada pelo ChirpStack. A fase analisada é a fase de registro e as etapas examinadas são as de iniciação, verificação e registro.

Com o ambiente preparado, foi realizada uma operação POST para o servidor de aplicação do ChirpStack. A solicitação enviou as informações contidas no Código 1.

[1] https://docs.docker.com/compose/

Código 1: Experimento 1 Cenário 1.

```
1  # Solicitação POST em http://localhost:8080/api/devices
2
3  {
4       "device": {
5            "applicationID": "1",
6            "description": "DEVICE",
7            "devEUI": "f8033201000288e1",
8            "deviceProfileID": "1",
9            "isDisabled": false,
10           "name": "MyDevice",
11           "referenceAltitude": 0,
12           "skipFCntCheck": true,
13           "tags": {},
14           "variables": {}
15           }
16 }
```

Neste experimento inicial, é criado um dispositivo no servidor de aplicação do ChirpStack. Este dispositivo faz uso de uma aplicação de teste criada para o experimento, a mesma leva o número de identificação *applicationID* de valor 1. O dispositivo cadastrado ainda leva uma descrição genérica, neste teste com o valor "DEVICE". Neste cadastro é usada a DevEUI de um dispositivo simulado com o valor f8033201000288e1. Além disto, foram definidos o *deviceProfileID* criado com valor 1. O dispositivo ainda é definido como inicialmente ativado para o servidor, com a opção *isDisabled* marcada como *false*.

Além disto, o mesmo é de identificado por sua DevEUI e tem definido um nome, neste caso estipulado o nome de "MyDevice". Por fim, o dispositivo ainda tem um atributo de *referenceAltitude* fixado em valor inicial de 0. O atributo skipFCntCheck, relativo a validação do contador de quadros, é um propriedade do dispositivo que protege de ataques de repetição. Portanto, o mesmo foi habilitado para criação do dispositivo. Por fim, as *tags* podem ser usadas como filtros de dispositivo não foram determinadas, bem como as variáveis.

Com os atributos definidos, foi realizada a solicitação para o *endpoint* definido na linha 1 do Código 1. Esta operação retornou positivo, indicando preliminarmente que o dispositivo foi criado corretamente. Para validação da criação do dispositivo final é realizada a requisição que consta no Código 2.

Código 2: Experimento 1 Cenário 1.

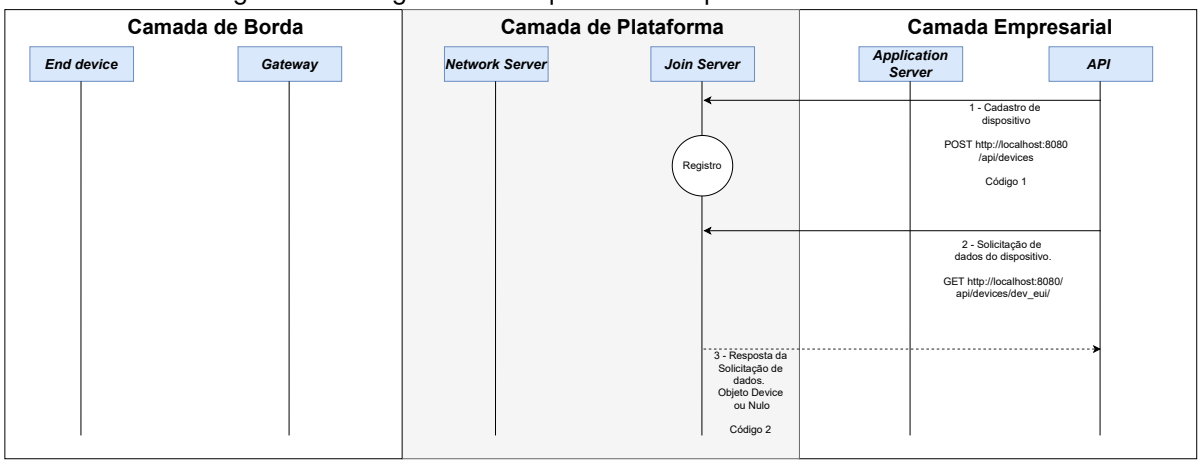
```
# Solicitação GET em http://localhost:8080/api/devices/dev_eui/

{
    "device":
        {
            "devEUI": "f8033201000288e1",
            "name": "MyDevice",
            "applicationID": "1",
            "description": "DEVICE",
            "deviceProfileID": "017d73bb-d7da-4b0e-aa2a-ff05a3cb51cb",
            "skipFCntCheck": false,
            "referenceAltitude": 0,
            "variables": ,
            "tags": ,
            "isDisabled": false
        },
    "lastSeenAt": null,
    "deviceStatusBattery": 256,
    "deviceStatusMargin": 256,
    "location": null
}
```

Conforme a resposta da solicitação GET realizada no Código 2 pode-se verificar que o dispositivo criado na solicitação do Código 1 foi bem-sucedida. Foram retornados as informações inseridas na solicitação inicial como a DevEUI, nome, identificador da aplicação, descrição, variáveis, *tags*, o contador de repetição (*skipFCntCheck*) e o controle do estado de habilitação (*isDisabled*). Além destas propriedades foram agregadas particularidades como o identificador do perfil do dispositivo (*deviceProfileID*) e os *logs* de visualização, bateria e localização. Estes *logs* são representados pelas propriedades de última visualização *lastSeenAt*, estado da bateria *deviceStatusBattery*, estado da margem de modulação que diz respeito a variação da modulação *deviceStatusMargin* e por fim a localização.

A credencial envolvida nesta operação é a DevEUI, que é detalhada na Subseção 2.2.2.2. Esta credencial é um identificador único global. Esta credencial é o principal identificador do dispositivo dentro de uma rede LoraWan. Neste caso, como o dispositivo é simulado, esta DevEUI é gerada de forma aleatória dentro do formato tradicional.

Em resumo, são realizadas duas requisições ao *join server*. A primeira requisição é responsável pelo registro deste dispositivo. A segunda requisição coleta informações do dispositivo cadastrado e verifica o sucesso da primeira operação. Estas requisições são ilustradas na Figura 25.

Figura 25 – Diagrama de sequência do experimento do Cenário 1.



Fonte: O autor.

Na Figura 25 foram ilustradas as solicitações feitas ao ChirpStack. A operação POST, representa a solicitação realizada pelo Código 1. Esta solicitação teve sua resposta positiva tendo um retorno de estado código 200 do protocolo HTTP. A solicitação GET, presenta a solicitação realizada pelo Código 2. Esta solicitação teve resposta positiva com retorno dos dados referente ao dispositivo cadastrado na operação POST.

Com estas respostas positivas pode-se concluir que por meio desta solicitação foram realizadas as etapas de iniciação, verificação e registro de dispositivos. A etapa de iniciação foi realizada com sucesso, pois por meio da solicitação é declarada a intenção do dispositivo se colocar sob gestão do *join server*, obtendo uma identidade. A etapa de verificação foi realizada com êxito pois o sucesso da operação indica que houve sucesso em provar que esta é a entidade à qual a identidade deve ser criada e emitida. Por fim, a fase de registro teve resultado positivo levando em conta a correta subscrição do dispositivo no *joinserver* e o recebimento de identificadores como o *deviceProfileID*.

Consequentemente esta operação comprova de maneira satisfatória a realização da fase de registro do ciclo de vida de credenciais. Isto se deve ao fato da todas as etapas da fase de registro terem sido exitosas no teste. Portanto, o Cenário 1 atesta a realização das etapas e da fase analisadas. A análise do experimento deste cenário é consolidada na Tabela 11.

Tabela 11 – Consolidação do Cenário 1.

	Código da Execução	Credenciais envolvidas	Adequação aos requisitos	Critérios de Análise	Resultados
Experimento 1	Código 1 Código 2	devEUI	Cumpridos totalmente	Critério 1	A operação executa as etapas de iniciação, verificação e registro.
				Critério 2	A operação executa a fase de registro.

Fonte: O autor.

A Tabela 11 reúne informações inerentes ao experimento realizado neste cenário. Em primeiro lugar são apontados os códigos nos quais o experimento foi realizado, de maneira a detalhar a execução do mesmo. Neste cenário foram trabalhados os Códigos 1 e 2. As credenciais envolvidas nessa operação são evidenciadas para análise concreta de quais as credenciais usadas no experimento. Credenciais detalhadas na Tabela 2.3.2. Além disto, é mostrada a conclusão sobre a adequação do experimento aos requisitos apontados na Subseção 2.1.2.2. Ademais, o experimento é analisados de acordo com os critérios de análise apresentados na Seção 3.1. Por fim, uma descrição dos resultados obtidos são explanadas de forma a especificar os resultados obtidos. Ressaltando, neste cenário, os resultados foram positivos no sentido que o experimento atesta a adequação do ciclo de vida detalhado na Capítulo 2.

4.3 CENÁRIO 2

O objetivo deste cenário de experimentos é a verificação das etapas de geração, vinculação, emissão e armazenamento de credenciais. Consequentemente, objetiva também analisar a fase de gerenciamento de credenciais. Esta análise ciclo de vida acontece no ambiente de testes do trabalho desenhado no Capítulo 3. Assim como, o contexto é composto por uma rede LoraWan privada proporcionada pelo ChirpStack. A fase analisada é a fase de gerenciamento de credenciais e as etapas examinadas são as de geração, vinculação, emissão e armazenamento.

Com o ambiente preparado, foi realizada uma operação POST para o servidor de aplicação do ChirpStack. A solicitação enviou as informações contidas no Código 3.

Código 3: Experimento 1 do Cenário 2.

```
1  # Solicitação POST http://localhost:8080/ api/devices/device_keys.dev_eui/keys
2
3  {
4      "deviceKeys": {
5          "appKey": "c89009a6fdb9752f553e6f41e6ca659e",
6          "devEUI": "f8033201000288e1",
7          "genAppKey": ,
8          "nwkKey": "03a9694378866ef2e798b88c1b0206e5"
9          }
10 }
```

De acordo com a requisição de método POST realizada no Código 3 pode-se observar a imputação de credenciais a um dispositivo. As informações enviadas são necessárias para a geração e vinculação de credenciais com a entidade do dispositivo.

Das credenciais envolvidas no processo, a appKey é uma credencial raiz usada na derivação de outras credenciais, a mesma é codificada em hexadecimal. A credencial DevEUI é enviada no corpo da solicitação, como identificador universal do dispositivo ao qual serão atribuídas as outras credenciais enviadas. A genAppKey é uma credencial opcional que implementa especificações de configuração remotas de *multicast*. Esta credencial é definida apenas em dispositivos LoraWan 1.0, versão não utilizada no ambiente deste trabalho. Por este motivo, a credencial não é enviada nesta requisição. Por fim, nesta requisição é enviada a chave raiz de rede, denominada nwkKey. Esta chave é utilizada para derivação de outras credenciais de rede como FNwkSIntKey, SNwkSIntKey e NwkSEncKey.

A execução da solicitação realizada com o corpo demonstrado no Código 3 obteve êxito. A mesma retornou uma resposta de sucesso na operação conforme esperado. A verificação da efetividade desta operação e o armazenamento das credenciais são testados no Código 4.

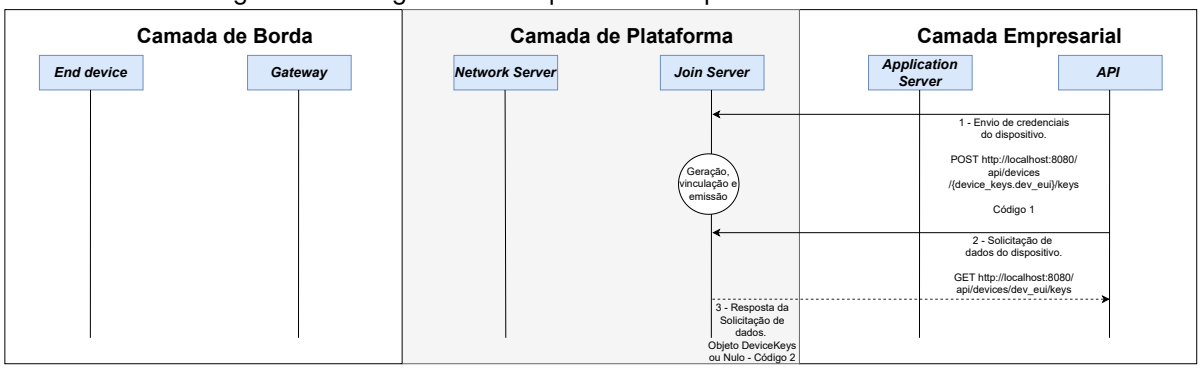
Código 4: Experimento 1 do Cenário 2.

```
1  # Solicitação GET http://localhost:8080/ api/devices/device_keys.dev_eui/keys
2
3  {
4       "deviceKeys": {
5            "devEUI": "f8033201000288e1",
6            "nwkKey": "03a9694378866ef2e798b88c1b0206e5"
7            "appKey": "c89009a6fdb9752f553e6f41e6ca659e",
8            "genAppKey": ,
9            }
10 }
```

O Código 4 é explicitada a requisição método GET para obtenção de credenciais atuais do dispositivo registrado. No corpo dessa resposta existem as credenciais cadastradas no experimento prévio, exemplificado no Código 4. Experimento este no qual foram enviadas as credenciais appKey, DevEUI, nwkKey e genAppKey. Com este resultado da solicitação inferiu-se que a operação gerou, vinculou e imitiu as credenciais do dispositivo registrado.

Em resumo neste cenário são realizadas duas requisições ao *join server*. Em primeiro lugar é realizada uma requisição para solicitar a vinculação das credenciais envolvidas na requisição, especificamente no Código 3. Posteriormente é feita uma solicitação de dados do dispositivo, especificamente suas credenciais, para análise do teste. Estas operações são ilustradas na Figura 26.

Figura 26 – Diagrama de sequência do experimento do Cenário 2.



Fonte: O autor.

No diagrama de sequência que consta na Figura 26 observam-se as requisições envolvidas no cenário. No passo 1 é representada a solicitação do Código 3. Esta requisição tem o sentido partindo da API do ChirpStack para o *join server*. Após essa operação observou-se a ocorrência da geração, vinculação e emissão das credenciais. No passo 2 do diagrama, ainda, são solicitados dados do dispositivo que responde com o objeto DeviceKeys no Código 4.

Tendo em vista o sucesso das etapas de geração, vinculação e emissão das credenciais atesta-se que as respectivas etapas do ciclo de vida são cumpridas satisfatoriamente. Consequentemente a fase de gerenciamento de credenciais do ciclo de vida é parcialmente atendida com o resultado destes experimentos. A análise do experimento deste cenário é consolidada na Tabela 12.

Tabela 12 – Consolidação do Cenário 2.

	Código da Execução	**Credenciais envolvidas**	**Adequação aos requisitos**	**Critérios de Análise**	**Resultados**
Experimento 1	Código 3 Código 4	devEUI, appKey, nwkKey, genAppKey	Cumpridos totalmente	Critério 1	A operação executa as etapas de geração, vinculação, emissão e armazenamento.
				Critério 2	A operação executa a fase de gerenciamento de credenciais.

Fonte: O autor.

A Tabela 12 faz uma consolidação dos resultados do Cenário 2 de experimentação. O experimento usa as credenciais devEUI, appKey, nwkKey, genAppKey. O experimento teve os requisitos cumpridos, nos dois critérios critérios analisados.

O experimento tem este cenário positivo pois a operação executa as etapas de geração, vinculação, emissão e armazenamento. O experimento ainda executa corretamente a fase de gerenciamento de credenciais nesse quesito. Os Códigos 3 e 4 expressam o procedimento desta testagem.

O experimento, tem resultado afirmativo pois executa as etapas de geração, vinculação, emissão e armazenamento de acordo com o Critério 1. Além disto, este

experimento comprova a adequação ao Critério 2 que denota que a operação executa a fase de gerenciamento de credenciais. De forma geral, este cenário mostra a adequação do ambiente de testes em relação as etapas e fase analisados.

4.4 CENÁRIO 3

Este cenário tem como finalidade é a análise das etapas de renovação, substituição e rotatividade bem como suspensão e revogação. Estas etapas são pertencentes a fase de gerenciamento de credenciais do ciclo de vida de credenciais apresentado na Subseção 2.1.2.2.

Esta análise do ciclo de vida acontece no ambiente de testes detalhado no Capítulo 3. Além disto, o experimento se vale de dispositivos simulados criados no ambiente nos Cenários 1 e 2. Com o ambiente definido, foi enviada uma solicitação PUT como um teste relativo á capacidade do ambiente fazer renovação, substituição e rotatividade. O corpo da solicitação é listado no Código 5.

Código 5: Corpo da solicitação do Experimento 1 do Cenário 3.

```
1  # Solicitação PUT http://localhost:8080/ api/devices/device_keys.dev_eui/keys
2
3  {
4       "deviceKeys": {
5            "appKey": "71E10DF3782E42FEA9D3601BE76CA4EA",
6            "devEUI": "71657cf0d33e130a",
7            "genAppKey": ,
8            "nwkKey": "F3A53E80E69F5CBA7613DB62B1E77B9E"
9            }
10 }
```

No Código 5, no objeto deviceKeys foram enviadas novas credenciais appKey e nwkKey. A credencial devEUI permanece a mesma dos experimentos anteriores por se tratar do identificador do dispositivo conforme a Tabela 8. O resultado esperado deste teste é que as credenciais sejam substituídas pelas novas credenciais. Este processo equivale a etapa de substituição das credenciais no ciclo de vida analisado.

O Código 6 faz uma solicitação GET ao *join server* para obtenção das credenciais do dispositivo usado no teste. Espera-se que esta operação comprove a substituição das credenciais.

Código 6: Corpo da solicitação do Experimento 1 do Cenário 3.

```
# Solicitação GET http://localhost:8080/ api/devices/device_keys.dev_eui/keys

{
    "deviceKeys": {
        "appKey": "71E10DF3782E42FEA9D3601BE76CA4EA",
        "devEUI": "71657cf0d33e130a",
        "genAppKey": ,
        "nwkKey": "F3A53E80E69F5CBA7613DB62B1E77B9E"
        }
}
```

Conforme observado no Código 6 a solicitação retorna um resultado positivo. Isto se deve ao fato de que as credenciais appKey e nwkKey foram sido substituídas pelos valores imputados no Código 5. Entretanto, após uma análise sobre a API do ChirpStack foi constatado que existe a possibilidade da substituição e renovação de credenciais, mas não de rotatividade das mesmas. Neste caso, o experimento atesta que a operação executa as etapas de forma parcial e consequentemente a fase de gerenciamento de credenciais é também executada de forma incompleta.

Ainda na fase de gerenciamento de credenciais, as etapas de suspensão e revogação tem papel central no ciclo de vida. No cenário atual, foram feitas execuções para testagem destas etapas, com um código usado para requisição DELETE no *endpoint* api/devices/device_keys.dev_eui/key. Este código valida a execução das etapas de suspensão e revogação de credenciais.

A solicitação de DELETE ao *endpoint* api/devices/device_keys.dev_eui/key é empregada para solicitar ao *join server* a exclusão de credenciais do dispositivo usado no teste. A execução retornou status 200, indicando confirmação de sucesso. A solicitação não conta com um corpo, e nem com resposta no formato texto. Nesta requisição o campo vazio significa sucesso na operação. Espera-se com essa operação atingir o objetivo de revogar a credencial. Além disto, busca-se suspender a credencial.

Os testes de suspensão e revogação são validados com a requisição presente no Código 7. Neste, espera-se que o dispositivo tenha como vazio e inexistente o objeto deviceKeys.

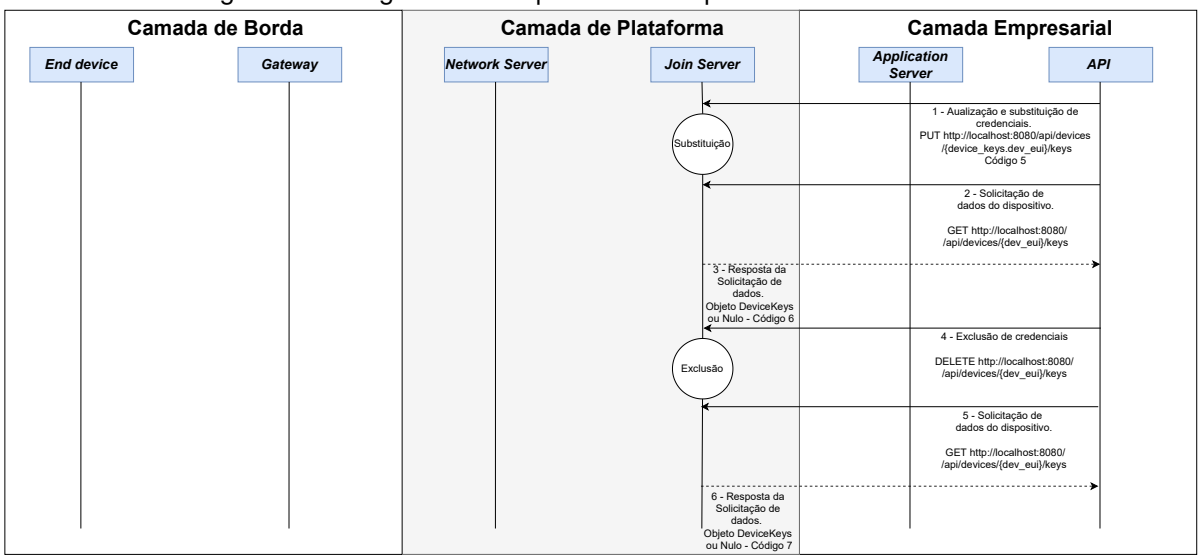
Código 7: Corpo da solicitação do Experimento 2 do Cenário 3.

```
# Solicitação GET http://localhost:8080/ api/devices/device_keys.dev_eui/keys

{
    "error": "object does not exist",
    "code": 5,
    "message": "object does not exist",
    "details": []
}
```

Este resultado, embora conste como uma mensagem de erro indicando valor nulo, significa a obtenção do resultado esperado. Isto se deve ao fato do dispositivo não contar com o objeto device keys. Este resultado atesta que a operação de revogação foi bem sucedida. Em contrapartida, a API do ChirpStack não oferece suporte para operações de suspensão e rotatividade de credenciais. Observa-se que esta não é uma limitação do protocolo LoraWan em si, mas uma funcionalidade não implementada no servidor de aplicação do ChirpStack.

De forma resumida, os experimentos do cenário fizeram duas operações objetivando testar etapas da fase de gerenciamento de credenciais do ciclo de vida. As operações se dão exclusivamente entre a API do ChirpStack e o *Join Server*. Estas requisições podem ser observadas na Figura 27.

Figura 27 – Diagrama de sequência dos experimentos do Cenário 3.



Fonte: O autor.

Na Figura 27 as operações representam as requisições da API feitas ao *Join Server*. A operação de número 1 é uma operação como o método DELETE para realizar a substituição das credenciais. Esta requisição é detalhada no Código 5. A operação 2 é uma solicitação de dados do dispositivo de método GET, detalhado no Código 6. A resposta, operação 3 da Figura 27 foi a solicitação que denotou que a etapa de substituição foi bem sucedida. Consequentemente, a etapa de renovação foi contemplada.

Além disto, nos testes relativos as etapas de suspensão e revogação foram realizadas as operações 4, 5 e 6. Na operação 4, de método DELETE foi enviada uma solicitação de revogação da credencial. A operação 5 é uma solicitação de dados feita para verificar a efetividade da operação de revogação, a mesma é listada no Código 7. Além disto a resposta da operação 6 valida que as credenciais foram devidamente revogadas. É importante observar que a API do ChirpStack não dispunha da possibi-

lidade de fazer a suspensão de credenciais. A Tabela 13 consolida os resultados dos testes associados aos códigos de execução, as credenciais envolvidas, a adequação dos requisitos e os critérios de análise.

Tabela 13 – Consolidação do Cenário 3.

	Código da Execução	**Credenciais envolvidas**	**Adequação aos requisitos**	**Critérios de Análise**	**Resultados**
Experimento 1	Código 5 Código 6	devEUI, appKey, nwkKey, genAppKey	Cumpridos parcialmente	Critério 1	A operação executa as etapas renovação e substituição. Não executa a etapa de rotatividade.
				Critério 2	A operação executa a fase de gerenciamento de credenciais parcialmente
Experimento 2	Código 7	devEUI, appKey, nwkKey, genAppKey	Cumpridos parcialmente	Critério 1	A operação executa a etapa de suspensão. Não executa a etapa de revogação.
				Critério 2	A operação executa a fase de gerenciamento de credenciais parcialmente.

Fonte: O autor.

Conforme observado na Tabela 13 o primeiro experimento, relativo as etapas de renovação, substituição e rotatividade teve uma cumpre parcialmente os requisitos de ciclo de vida. Portanto, o Critério 1, em relação a este experimento não foi cumprido. A análise dá conta que a operação executa as etapas renovação e substituição mas não executa a etapa de rotatividade.

Ainda no primeiro experimento, o Critério 2 dá conta que a operação executa a fase de gerenciamento de credenciais apenas parcialmente. Isto se deve ao fato de não implementar todas as etapas desta da fase.

A Tabela 13 ainda sumariza os resultados do Experimento 2. O mesmo, em relação ao primeiro critério de análise que diz respeito ao atendimento dos requisitos das etapas não se mostrou adequado as etapas que devem acontecer de acordo com os padrões de ciclo de vida de credenciais. Isto se deve pelo fato do ambiente realizar a etapa de revogação, mas não a etapa de suspensão. Por este mesmo motivo este experimento não atende ao Critério 2, pois não implementa corretamente a fase de gerenciamento de credenciais.

De forma geral, este critério indica que o ciclo de vida de credenciais é atendido parcialmente nas etapas analisadas no mesmo. Consequentemente a fase analisada, também é atendida apenas parcialmente.

4.5 CENÁRIO 4

O objetivo deste cenário é a verificação das etapas de autenticação e autorização e da fase de autenticação. Esta análise ciclo de vida acontece no ambiente de testes do trabalho descrito no Capítulo 3. Assim como, o ambiente do contexto é composto por uma rede LoraWan privada implementada pelo ChirpStack.

O dispositivo final simulado é configurado para fazer uma solicitação de junção. Esta solicitação é equivalente a uma solicitação de autenticação, conforme ex-

posto na Seção 2.2.2.1. Esta operação é encaminhada ao *Network Server* que reencaminha ao *Join Server* que processa a solicitação de autenticação e emite a mensagem de aceitação. Além disto, é necessário fazer o processo de ativação, que diz respeito a autorização do dispositivo.

Com o cenário preparado e com o dispositivo tendo realizado a solicitação de junção, foi realizada uma operação POST para o *Join Server* do ChirpStack. A solicitação enviou as informações contidas no Código 8 e teve como objetivo a realização do processo de ativação.

Código 8: Experimento 1 do Cenário 4.

```
1  # Solicitação POST http://localhost:8080/ /api/devices/ device_activation.dev_eui/activate
2
3  {
4       "deviceActivation": {
5            "aFCntDown": 0,
6            "appSKey": "09a1c687b5de522321 d99e7f4c95aa9c",
7            "devAddr": "c1ff6012",
8            "devEUI": "71657cf0d33e130a",
9            "fCntUp": 0,
10           "fNwkSIntKey": "A5d308B01a5C67aDE18cA1fFE5ddd41c",
11           "nFCntDown": 0,
12           "nwkSEncKey": "aE5cb50C2A2465ce62688Bca1278af73",
13           "sNwkSIntKey": "0E3b8a6722CeD1EE2faA984AA76dCC1e"
14           }
15 }
```

Conforme o Código 8 assevera, foram enviadas credenciais e outras informações para o processo de ativação. As credenciais de sessão envolvidas foram appSKey, fNwkSIntKey, nwkSEncKey e sNwkSIntKey. Além disso, a credencial identificador do dispositivo na rede atual, devAddr também consta na requisição. Para mais, o identificador único do dispositivo DevEUI também é incorporado na requisição.

Tendo realizado esta solicitação de ativação, foi necessária a verificação de sua efetividade. Para isso, o Código 9 realizou uma requisição GET para o *join server*. O esperado seria obter como resultado esperado, o status de ativação e as credenciais enviadas no Código 8.

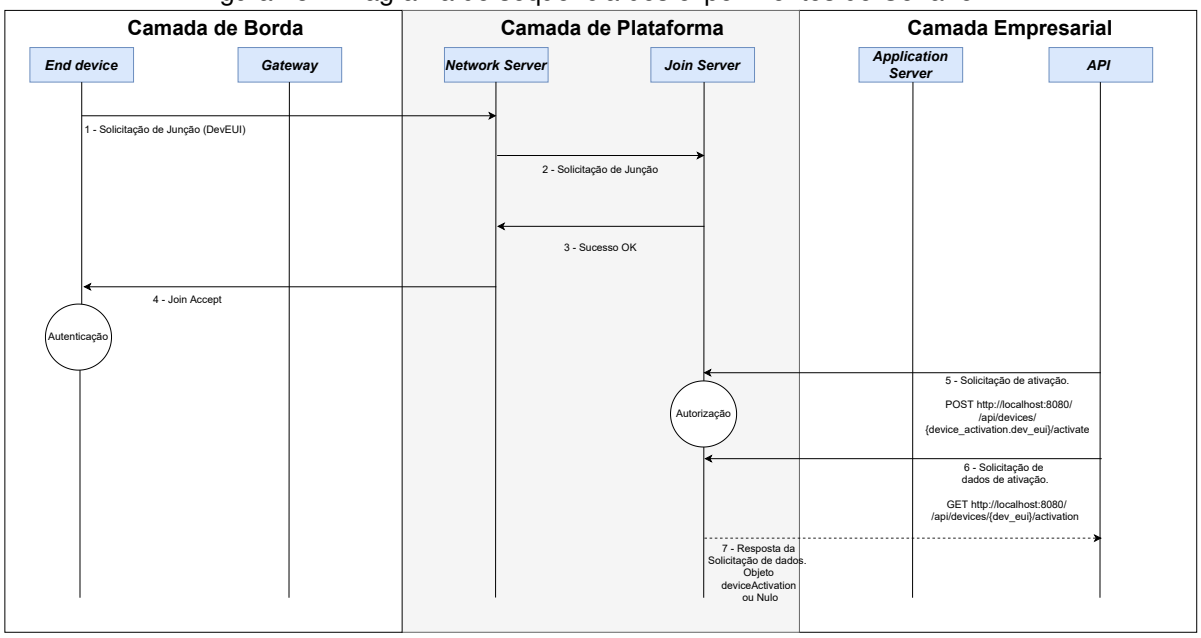
Código 9: Experimento 1 do Cenário 4.

```
1  # Solicitação GET http://localhost:8080/ /api/devices/dev_eui/activation
2
3  {
4       "deviceActivation": {
5            "aFCntDown": 0,
6            "appSKey": "09a1c687b5de522321 d99e7f4c95aa9c",
7            "devAddr": "c1ff6012",
8            "devEUI": "71657cf0d33e130a",
9            "fCntUp": 0,
10           "fNwkSIntKey": "A5d308B01a5C67aDE18cA1fFE5ddd41c",
11           "nFCntDown": 0,
12           "nwkSEncKey": "aE5cb50C2A2465ce62688Bca1278af73",
13           "sNwkSIntKey": "0E3b8a6722CeD1EE2faA984AA76dCC1e"
14           }
15 }
```

O Código 9 explicita que o teste realizado foi bem sucedido no sentido da ativação do dispositivo. Observa-se que foram retornadas as credenciais previamente cadastradas com as credenciais de sessão devidamente definidas.

Com o estas requisições retornando resultado positivo, então foi possível inferir que o ambiente de testes implementa corretamente as etapas de autenticação e autorização do ciclo de vida de credenciais conforme os padrões fundamentados no Capítulo 2. As requisições deste teste são elencadas na Figura 28.

Figura 28 – Diagrama de sequência dos experimentos do Cenário 4.



Fonte: O autor.

No diagrama de sequência da Figura 28 são listadas as etapas do teste no atual cenário. As primeiras quatro operações representam o procedimento de junção. As três operações posteriores, representam o processo de ativação.

No processo de ativação com a iniciativa do dispositivo simulado, segundo o teste, é realizada a autenticação. A autenticação se dá pelas operações solicitação de junção usando a DevEUI do dispositivo, esta solicitação é encaminhada ao *join server* que responde com uma mensagem de sucesso e encaminha uma mensagem de *join accept* pelo *network server*.

Em sequência, o processo de autorização é verificado com uma operação de solicitação de ativação. Esta solicitação, de número 5, tem o método POST e envia as credenciais e variáveis citadas no Código 8. Posteriormente, a requisição 6 solicita os dados do dispositivo, ademais a operação 7 recebe os dados do mesmo dispositivo. Este conjunto de operações com resultados positivos se desdobra na possibilidade de aferir que este cenário implementa as etapas de autenticação e autorização. Além disto, implementa a fase de autenticação do ciclo de vida de credenciais proposto pelos órgãos reguladores citados no Capítulo 2. O experimento, códigos e análises são relacionados na Tabela 14.

Tabela 14 – Consolidação do Cenário 4.

	Código da Execução	**Credenciais envolvidas**	**Adequação aos requisitos**	**Critérios de Análise**	**Resultados**
Experimento 1	Código 8 Código 9	appSKey, devAddr, devEUI, fNwkSIntKey, nFCntDown, nwkSEncKey e sNwkSIntKey	Cumpridos totalmente	Critério 1	A operação executa as etapas de autenticação e autorização.
				Critério 2	A operação executa a fase de autenticação.

Fonte: O autor.

A Tabela 14 relaciona os códigos, credenciais envolvidas, requisitos e critérios de análise com os resultados do experimento. Isso resulta na análise da adequação ao ciclo de vida. Em termos concretos o experimento realizado neste cenário teve como resultado a adequação ao ciclo de vida de credenciais nos termos dos Critérios 1 e 2. Isto indica que o ambiente implementa a etapa e fases do ciclo, respectivamente.

4.6 CENÁRIO 5

Este cenário tem como objetivo avaliar o ambiente de testes, uma rede privada LoraWan para IIoT em relação a etapa de auditoria do ciclo de vida das credenciais. Este cenário tem uma especificidade tendo em conta que a auditoria de acontecer em todas as fases do ciclo de vida. A auditoria é uma etapa transversal em relação as fases, conforme ilustrado na Figura 2.

Para verificar os eventos auditáveis, foi realizada uma operação de solicitação a API do ChirpStack como forma de obtenção de informações sobre os eventos de segurança realizados. A requisição é detalhada no Código 10.

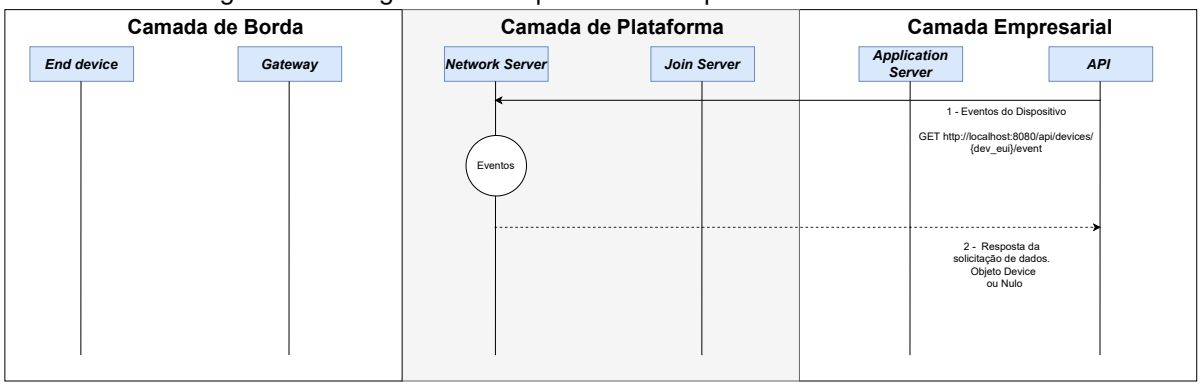
Código 10: Corpo da solicitação do Experimento 1 do Cenário 5

```
1  # Solicitação GET http://localhost:8080/ api/devices/dev_eui/events
2
3      {
4      {
5          "applicationID": "123",
6          "applicationName": "temperature-sensor",
7          "deviceName": "garden-sensor",
8          "devEUI": "AgICAgICAgI=",
9          "rxInfo": [
10             {
11             "gatewayID": "AwMDAwMDAwM=",
12             "time": "2019-11-08T13:59:25.048445Z",
13             "timeSinceGPSEpoch": null,
14             "rssi": -48,
15             "loRaSNR": 9,
16             "channel": 5,
17             "rfChain": 0,
18             "board": 0,
19             "antenna": 0,
20             "location": {
21             "latitude": 52.3740364,
22             "longitude": 4.9144401,
23             "altitude": 10.5
24             },
25                 "fineTimestampType": "NONE",
26                 "context": "9u/uvA==",
27                 "uplinkID": "jhMh8Gq6RAOChSKbi83RHQ=="
28                 }
29                 ],
30                     "txInfo": {
31                         "frequency": 868100000,
32                         "modulation": "LORA",
33                         "loRaModulationInfo": {
34                         "bandwidth": 125,
35                         "spreadingFactor": 11,
36                         "codeRate": "4/5",
37                         "polarizationInversion": false
38                         }
39                     },
40                     "acknowledged": true,
41                     "fCnt": 15,
42                     "tags": {
43                     "key": "value"
44                     }
45             }
46         }
```

No Código 10 foi obtido o resultado dos eventos do dispositivo final simulado. Dentre essas informações, os eventos de segurança são compreendidos apenas na linha 40 com a propriedade *acknowledged*. Esta propriedade tem valor booleano e está relacionado se a o dispositivo é reconhecido pelo sistema ou não. Embora esta informação tenha valor ao se tratar de informações do dispositivo, não oferece um histórico de eventos de segurança. Informações como data e hora de procedimentos de autenticação e autorização por exemplo, são salutares a um esquema de auditoria.

Portanto, assevera-se a partir da requisição realizada que o ambiente estudado não garante a realização da etapa de auditoria. Consequentemente as fases que esta compõe não são atendidas em termos de auditoria. No diagrama da Figura 29 as etapas e a requisição são elencadas.

Figura 29 – Diagrama de sequência dos experimentos do Cenário 5.



Fonte: O autor.

Na Figura 29 é realizada uma requisição identificada como número 1. Esta requisição usa o método GET do HTTP. A resposta da solicitação de dados é representado pela operação 2 do experimento e detalhada no Código 10. Esta resposta, embora retorne os dados solicitados, se mostrou insuficiente em termos de informações para auditoria. Segundo (SCHRECKER et al., 2016) a etapa de auditoria deve permitir que todas as operações de gerenciamento de credenciais possam ser rastreadas para fins de auditoria. O ambiente analisado não tem nenhuma possibilidade de rastreamento das operações do ciclo de vida. Portanto, os eventos disponibilizados pela API do ChirpStack não contemplam informações suficientes para disponibilizar uma auditoria nos moldes dos padrões do IIC. A análise do experimento atrelada as variáveis de credenciais, requisitos e critérios é analisada na Tabela 15.

Tabela 15 – Consolidação do Cenário 5.

	Código da Execução	**Credenciais envolvidas**	**Adequação aos requisitos**	**Critérios de Análise**	**Resultados**
Experimento 1	Código 10	devEUI	Não cumprido	Critério 3	A operação não executa etapa de auditoria em nenhuma das fases do ciclo de vida de credenciais.

Fonte: O autor.

Na Tabela 15 são roborados os dados relativos ao experimento deste cenário. A credencial usada neste experimento é a DevEUI para identificação do dispositivo do qual solicitou-se os eventos. Em relação aos critérios de análise, o Critério 1 relativo a implementação da etapa mostrou-se não contemplado no ambiente. O mesmo ocorreu com o Critério 2, no qual a insuficiência de informação fez com que as fases as quais a etapa de auditoria estava presente não ficarem em conformidade com o ciclo de vida de credenciais nos padrões estabelecidos.

O ciclo de experimentações tem sua conclusão neste cenário em que avalia-se o contexto da auditoria. Este e os demais cenários foram compilados para exposição dos resultados consolidados na Seção 4.7.

4.7 ANÁLISE E COMPARAÇÃO DOS RESULTADOS

O processo de concretização dos testes realizados no Capítulo 4 passou primeiramente pela inicialização e realização dos experimentos. Cada um destes experimentos faz uma análise de um grupo de etapas de uma fase do ciclo de vida de credenciais descrito na fundamentação do trabalho. Cada cenário de experimento inicia com uma definição de objetivos e uma definição do escopo relacionado. Assim, o escopo é relativo às etapas e fases analisadas. Assim como realizado individualmente em cada cenário, foi realizada uma análise de forma global dos cenários. A definição dos cenários e experimentos bem como os atributos que fazem parte da execução dos testes são descritos na Tabela 16.

Tabela 16 – Consolidação dos testes realizados.

Cenário de Testes	**Experimento**	**Código da Execução**	**Credenciais envolvidas**	**Adequação aos requisitos**	**Critérios de Análise**	**Resultados**
Cenário 1	Experimento 1	Código 1 Código 2	devEUI	Cumpridos totalmente	Critério 1	A operação executa as etapas de iniciação, verificação e registro.
					Critério 2	A operação executa a fase de registro.
Cenário 2	Experimento 1	Código 3 Código 4	devEUI, appKey, nwkKey, genAppKey	Cumpridos totalmente	Critério 1	A operação executa as etapas de geração, vinculação, emissão e armazenamento.
					Critério 2	A operação executa a fase de gerenciamento de credenciais.
Cenário 3	Experimento 1	Código 5 Código 6	devEUI, appKey, nwkKey, genAppKey	Cumpridos parcialmente	Critério 1	A operação executa as etapas renovação e substituição. Não executa a etapa de rotatividade.
					Critério 2	A operação executa a fase de gerenciamento de credenciais parcialmente.
	Experimento 2	Código 7	devEUI, appKey, nwkKey, genAppKey	Cumpridos parcialmente	Critério 1	A operação executa a etapa de suspensão. Não executa a etapa de revogação.
					Critério 2	A operação executa a fase de gerenciamento de credenciais parcialmente.
Cenário 4	Experimento 1	Código 8 Código 9	appSKey, devAddr, devEUI, fNwkSIntKey, nFCntDown, nwkSEncKey e sNwkSIntKey	Cumpridos totalmente	Critério 1	A operação executa as etapas de autenticação e autorização.
					Critério 2	A operação executa a fase de autenticação.
Cenário 5	Experimento 1	Código 10	devEUI	Não cumprido	Critério 3	A operação não executa etapa de auditoria em nenhuma das fases do ciclo de vida de credenciais.

Fonte: O autor.

Na Tabela 16 os testes são agrupados pelos cenários de testes. Cada cenário conta com experimentos, que por sua vez usam códigos para fazer as operações teste. Para cada experimento são listadas as credenciais do protocolo LoraWan envolvidas. Além disto, cada experimento é analisado pelo prisma dos critérios de análise levantados na Seção 3.1 e são apontados os pareceres desta análise.

No Cenário 1 objetivo é testar as etapas de iniciação, verificação e registro na fase de registro. Os testes são operacionalizados pelos Códigos 1 e 2. A credencial envolvida no teste é a devEUI. Este cenário teve os requisitos da análise cumpridos totalmente. Para tanto, este teste assevera que o cenário executa as etapas e a fase analisada. Isto contribui para os resultados de forma a atestar que ambiente de testes está em conformidade com os padrões.

O Cenário 2 teve como objetivo testar as etapas de geração, vinculação, emissão e armazenamento na etapa de gerenciamento de credenciais. Os testes foram

concretizados por meio das operações realizadas nos Códigos 3 e 4. Neste teste temos as credenciais devEUI, appKey, nwkKey e genAppKey envolvidas. Os requisitos foram cumpridos totalmente, o que soma aos resultados uma provada adequação do ambiente ao ciclo de vida. Adequação ás etapas de geração, vinculação, emissão e armazenamento bem como á fase de gerenciamento de credenciais.

O Cenário 3 tem dois experimentos. O primeiro usa dos Códigos 5 e 6 para operacionalização. As credenciais do LoraWan usadas são devEUI, appKey, nwkKey e genAppKey. Neste experimento os requisitos forma cumpridos apenas parcialmente. Isto se deve ao fato do servidor de aplicação não contar com a etapa de rotatividade. Assim, soma ao resultado geral um ponto de atenção do fato do protocolo LoraWan implementado pelo ChirpStack tem dissonâncias com o padrão de ciclo de vida. Ainda assim a renovação e substituição são compreendidas pelo cenário e por isso faz jus parcialmente os requisitos.

No mesmo Cenário 3 e Experimento 2, operacionalizado pelo Código 7 que faz uso das credenciais devEUI, appKey, nwkKey e genAppKey os requisitos são cumpridos apenas parcialmente. Isto se deve ao fato do ambiente não contar com a etapa de revogação do ciclo de vida de credenciais. Isto tem parte na análise geral com a constatação que a fase de gerenciamento de credenciais fica comprometida em termos de revogação.

Por sua vez, o Cenário 4 é concretizado com os Códigos 8 e 9 e analisa as credenciais appSKey, devAddr, devEUI, fNwkSIntKey, nFCntDown, nwkSEncKey e sNwkSIntKey. Na análise feita este cenário se adéqua aos critérios, de forma que são executadas as operações de autenticação e autorização. Por consequência a fase de autenticação se apresenta disponível de forma satisfatória. Isto agrega ao resultado que as operações de autenticação e autorização existem e estão disponíveis como parte do ciclo, mas isto é vital para o funcionamento de um esquema de autenticação portanto esse resultado era esperado.

O Cenário 5, que faz uso da credencial de identificação devEUI, operacionalizada pelo Código 10, não cumpre os requisitos analisados. O Critério 3 não é contemplado. Isto implica que não existe possibilidade de auditoria no ambiente dos testes. O que fere os padrões de ciclo de vida de credenciais. Na Tabela 17 apresenta-se um resumo dos resultados obtidos categorizados pelas fases e etapas do ciclo de vida.

Tabela 17 – Resumo dos resultados.

Fase do ciclo de vida	Etapa ciclo de vida	Critério de Análise	Cenário	Resultado obtido
Registro	Iniciação	Critério 1 Critério 2	Cenário 1	Adequado ao ciclo de vida
Registro	verificação	Critério 1 Critério 2	Cenário 1	Adequado ao ciclo de vida
Registro	Registro	Critério 1 Critério 2	Cenário 1	Adequado ao ciclo de vida
Gerenciamento de credenciais	Geração	Critério 1 Critério 2	Cenário 2	Adequado ao ciclo de vida
Gerenciamento de credenciais	Emissão	Critério 1 Critério 2	Cenário 2	Adequado ao ciclo de vida
Gerenciamento de credenciais	Armazenamento	Critério 1 Critério 2	Cenário 2	Adequado ao ciclo de vida
Gerenciamento de credenciais	Renovação, substituição e renovação.	Critério 1 Critério 2	Cenário 3	Parcialmente adequado ao ciclo de vida
Gerenciamento de credenciais	Suspensão e revogação.	Critério 1 Critério 2	Cenário 3	Parcialmente adequado ao ciclo de vida
Autenticação de entidades	Autenticação	Critério 1 Critério 2	Cenário 4	Adequado ao ciclo de vida
Autenticação de entidades	Autorização	Critério 1 Critério 2	Cenário 4	Adequado ao ciclo de vida
Autenticação de entidades	Auditoria	Critério 3	Cenário 5	Inadequado ao ciclo de vida

Na Tabela 17 pode-se observar as fases, etapas e critérios de análise categorizados pelos cenários de forma que possibilita uma análise dos resultados. Realizando uma análise geral do ciclo de vida de credenciais em redes privadas para IIoT implementadas com ChirpStack chegou-se a conclusão que:

- O ambiente cumpre os requisitos ligados as etapas de iniciação, verificação e registro da fase de registro.
- O ambiente cumpre os requisitos ligados as etapas de geração, vinculação, emissão e armazenamento da fase de gerenciamento de credenciais.
- O ambiente cumpre os requisitos associados as etapas de autenticação e autorização na fase de autenticação.
- O ambiente cumpre parcialmente os requisitos relativos a fase de gerenciamento de credenciais. Pois implementa corretamente as etapas de renovação e substituição mas não executa a etapa de rotatividade.

- O ambiente cumpre parcialmente os requisitos relativos a fase de gerenciamento de credenciais. Pois implementa corretamente a etapa de suspensão mas não executa a etapa de revogação.
- O ambiente não cumpre, nem parcialmente, a etapa de auditoria. Etapa que está presente em todas as fases.

Desta maneira o presente trabalho atesta que as redes LoraWan privadas implementadas em ChirpStack não se encontram totalmente adequadas aos padrões estabelecidos para a IIoT. Deste modo, este trabalho contribui para disponibilizar uma análise deste protocolo e do servidor ChirpStack de modo que os desenvolvedores possam estar cientes dos riscos e das inadequações deste arranjo em se tratando do ciclo de vida de credenciais em observância aos padrões.

4.8 CONSIDERAÇÕES PARCIAIS

O ponto de culminância que significam os cenários de testes, tem início na fundamentação teórica e na definição do objetivo desta pesquisa no Capítulo 2. Já no Capítulo 3 fora desenhada a proposta que se tornou o meio para delinear um ambiente geral de experimentação. Este por sua vez possibilitou um plano de testes que culminou nos cenários de experimentação.

No Capítulo 4 foram inicializados os experimentos para a execução dos mesmo no ambiente de testes. Nas Seções 4.2, 4.3, 4.4, 4.5 e 4.6 foram desenvolvidos os experimentos de maneira completa, com a explicitação dos códigos, métodos e diagramas usados.

Na Seção 4.7 foi feito um resumo dos pontos analisados em relação ao ciclo de vida de credenciais de forma a concretizar a contribuição deste trabalho. Por fim, o Capítulo 4 disponibiliza um resumo das análises que foram possíveis pelos testes. Esta análise se soma a outros pontos do trabalho como uma contribuição que era esperada no objetivo da pesquisa.

5 CONSIDERAÇÕES & PRÓXIMOS PASSOS

A IIoT é um conceito relativamente bem estabelecido. Com diversas padronizações que levam a existir uma arquitetura bem definida em conformidade com os órgãos de padronização como ISO, NIST e IIC. Diante disto, existem recomendações em segurança e ciclo de vida de identidades que serviram para balizar a definição dos objetivos. Além disto o objetivo foi determinado de acordo com um cenário computacional aderente aos requisitos da IIoT. Para tanto o protocolo LoraWan passou a fazer parte do cenário no qual a análise do ciclo de vida foi desenhada. Por fim, ao processo de definição do cenário foram incorporados os servidores ChirpStack. Em consequência firmou-se como objetivo deste trabalho uma análise do ciclo de vida de credenciais em redes LoraWan privadas para IIoT.

Em termos gerais, pode-se afirmar que o presente trabalho atingiu seu objetivo. Haja vista que faz uma fundamentação teórica para caracterizar a definição de ciclo de vida de credenciais em IIoT. A caracterização está de acordo com as principais instituições reguladoras do mercado internacional no ramo de Internet Industrial.

Em sequência o trabalho faz uma pesquisa sobre os requisitos de IIoT e delimita um ambiente a ser trabalho, neste caso LoraWan com provimento de servidores ChirpStack. Assim, é realizada uma pesquisa bibliográfica a partir da definição do problema. Tendo percorrido este caminho foi desenvolvida a proposta de análise, o ambiente de testes para concretização dos experimentos e da análise.

O caminho necessário para o atingimento do objetivo de analisar o ciclo de vida das credenciais de LoraWan provido pelo ChirpStack. Com o acercamento dos procedimentos de junção e ativação no LoraWan foram relacionadas as etapas e fases do ciclo de vida de credencias com as credenciais do protocolo. Esta interseccionalidade é uma característica importante da contribuição deste trabalho. Porém, a importância mais concreta deste trabalho foi poder evitar incidentes de segurança da informação em industrias por vulnerabilidades no ciclo de vida de credenciais.

Os principais fatores que contribuíram para o atingimento do objetivo foi a disponibilidade de documentação sobre o ciclo de vida de credenciais por entidades reguladoras de IIoT. A disponibilidade um ambiente de código aberto para implementação do protocolo LoraWan, para simulação e operação de dispositivos. Especialmente a API do ChirpStack tornou a manipulação de dispositivos e servidores bastante simplificada. Esta simplicidade que permitiu a execução das operações dos testes de maneira simples, prática e eficiente.

Em decorrência deste trabalho foram possíveis tecer considerações sobre o

escopo analisado no projeto. Foi possível afirmar que as redes privadas LoraWan providas por ChirpStack não implementam totalmente um ciclo de vida seguro e recomendado. Existem disparidades entre o ciclo de vida recomendado e o implementado principalmente em relação a auditoria. Ademais, etapas do ciclo de vida recomendado como rotatividade e revogação de credenciais não são implementadas neste ambiente. Assim, permite ao trabalho servir como base para implantação de tecnologias como LoraWan de forma consciente dos riscos ou que a implementação possa prover os pontos no qual esse arranjo computacional deixa a desejar em termos de ciclo de vida de credenciais.

Em que pese os objetivos tenham sido atingidos, algumas dificuldades foram encontradas na execução do projeto. Foi necessária toda uma assimilação de conceitos de tecnologia de internet das coisas para a indústria, vertical econômica até então pouco explorada pelo autor. Além disto, os conceitos relativos a redes LoraWan necessitaram de muita prática para serem compreendidos a ponto de fazer uma análise de seus mecanismos de segurança. Estas questões foram resolvidas com um trabalho relativamente profundo de pesquisa na fundamentação destes conceitos.

Outra questão que exigiu afinco do autor foi o desenho final da proposta e do ambiente de testes. Estas etapas, além do objetivo definido claramente, exigiu especificações técnicas muito apuradas. Estes detalhes passaram a ser conhecidos de acordo com o crescimento da familiaridade com a tecnologia do ambiente de testes. Além disso, os experimentos foram uma parte delicada na elaboração do trabalho, pois foram elaboradas diversas versões para garantir a clareza e reprodutibilidade dos testes. Isto foi resolvido e aprimorado usando recursos não textuais como diagramas de sequencia, códigos e tabelas.

Durante a execução do trabalho houveram mudanças no escopo definido na fase de projeto. Estas questões se deram principalmente pala necessária exploração das tecnologias envolvidas no trabalho. Além disto, foram substituídos alguns testes *in-loco* para testes simulados, tendo em vista a limitação de cronograma hábil com os parceiros do mercado industrial para tais testes e questões relacionadas a pandemia.

Ademais, este trabalho aponta para trabalhos futuros. Um levantamento bibliográfico mais compreensivo e atualizado em termos de IIoT poderia ser realizado, pois é uma contribuição no conceito de IIoT. Outro trabalho futuro que advém naturalmente deste trabalho seria o melhoramento do ChirpStack para o ciclo de vida de credenciais que esteja totalmente em conformidade com o ciclo de vida. Implementações com intuito de rastrear as operações de ciclo de vida se fazem especialmente importantes. Um outro trabalho que pode ser feito é resolver problemas de segurança como o armazenamento de credenciais as claras no banco de dados usado pelo ChirpStack. Tais tópicos podem ser objeto de trabalho futuro.

Durante o desenvolvimento deste trabalho as seguintes publicações foram realizadas:

- SILVA, S. H.; MIERS, C. Uma proposta de análise do ciclo de vida de credenciais em redes Lorawan privadas para IIoT. In: Anais da XIX Escola Regional de Redes de Computadores. Porto Alegre, RS, Brasil: SBC, 2021. p. 25–30. Disponível em: <https://sol.sbc.org.br/index.php/errc/article/view/18537>.
- SILVA, S. H.; MIERS, C. Análise de mecanismos de autenticação de dispositivos em redes móveis 5G para Internet Industrial (IIoT) categorizada por mMTC, eMBB e URLLC. In: Anais da XIX Escola Regional de Redes de Computadores. Porto Alegre, RS, Brasil: SBC, 2021. p. 25–30. Disponível em: <https://sol.sbc.org.br/index.php/errc/article/view/18548>.
- SILVA, S. H.; MIERS, C. C. Análise de requisitos de identificação e autorização para dispositivos e *gateways* de borda em IIoT. In: Anais da XVIII Escola Regional de Redes de Computadores. Porto Alegre, RS, Brasil: SBC, 2020. p. 60–65. Disponível em: <https://sol.sbc.org.br/index.php/errc/article/view/15190>.

Além da publicações listadas, houveram outras publicações submetidas que não foram aceitas e estão sendo avaliadas para nova submissão disso. Por fim, está sendo elaborado um artigo com os resultados completos da dissertação para submissão em evento específico da área.

REFERÊNCIAS

ABERLE, L. **A Comprehensive Guide to Enterprise IoT Project Success. IoT Agenda**. 2015.

ALLIANCE, L. Lorawan™ 1.1 specification. 2017.

ANATEL. Ato nº 14448, de 04 de dezembro de 2017. **Diário Oficial [da] República Federativa do Brasil**, Brasília, DF, 2017. Disponível em: <https://informacoes.anatel.gov.br/legislacao/atos-de-certificacao-de-produtos/2017/1139-ato-14448>.

BADER, S. R.; MALESHKOVA, M.; LOHMANN, S. Structuring reference architectures for the industrial internet of things. **Future Internet**, Multidisciplinary Digital Publishing Institute, v. 11, n. 7, p. 151, 2019.

BATAGLINI, W. V. et al. Desenvolvimento de um instrumento para diagnóstico de maturidade das tecnologias da indústria 4.0 no setor têxtil e de confecção. 2021.

BERLANDA, R. G. Guia de segurança da informação para a conectividade de dispositivos iot. 2021.

BOR, M.; VIDLER, J. E.; ROEDIG, U. Lora for the internet of things. Junction Publishing, 2016.

BOYE, C. A.; KEARNEY, P.; JOSEPHS, M. Cyber-risks in the industrial internet of things (iiot): Towards a method for continuous assessment. p. 502–519, 2018.

BREIVOLD, H. P.; SANDSTRÖM, K. Internet of things for industrial automation–challenges and technical solutions. In: IEEE. **2015 IEEE International Conference on Data Science and Data Intensive Systems**. [S.l.], 2015. p. 532–539.

BROWN, G. et al. Ultra-reliable low-latency 5g for industrial automation. **Technol. Rep. Qualcomm**, v. 2, p. 52065394, 2018.

CARVALHO, D.; MIERS, C. Uma proposta de estudo comparativo de nb-iot vs lorawan para aplicação em redes iiot privadas para automação e monitoramento de processos. In: **Anais da XIX Escola Regional de Redes de Computadores**. Porto Alegre, RS, Brasil: SBC, 2021. p. 19–24. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/errc/article/view/18536>.

CHEN, X.; WANG, J.; WANG, L. A fast session key generation scheme for lorawan. In: IEEE. **2019 Australian & New Zealand Control Conference (ANZCC)**. [S.l.], 2019. p. 63–66.

CHIRPSTACK. **ChirpStack Architecture**. 2022. Disponível em: <https://www.chirpstack.io/project/architecture/>.

DANISH, S. M. et al. Securing the lorawan join procedure using blockchains. **Cluster Computing**, Springer, v. 23, n. 3, p. 2123–2138, 2020.

DEMPSEY, K.; PILLITTERI, V.; REGENSCHEID, A. **Managing the Security of Information Exchanges**. [S.l.], 2021.

DöNMEZ, T. C.; NIGUSSIE, E. Security of lorawan v1.1 in backward compatibility scenarios. **Procedia Computer Science**, v. 134, p. 51–58, 2018. ISSN 1877-0509. The 15th International Conference on Mobile Systems and Pervasive Computing (MobiSPC 2018) / The 13th International Conference on Future Networks and Communications (FNC-2018) / Affiliated Workshops. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1877050918311062>.

FEHRI, C. E. et al. Lora technology mac layer operations and research issues. **Procedia computer science**, Elsevier, v. 130, p. 1096–1101, 2018.

FELLI, L.; GIULIANO, R. Access control in woodland through blockchain and lorawan. In: **2021 AEIT International Conference on Electrical and Electronic Technologies for Automotive (AEIT AUTOMOTIVE)**. [S.l.: s.n.], 2021. p. 1–5.

FLOYER, D. Defining and sizing the industrial internet. **Wikibon, Marlborough, MA, USA**, 2013.

FOROUZAN, B. A.; COOMBS, C. A.; FEGAN, S. C. **Introduction to data communications and Networking**. [S.l.]: McGraw-Hill, 2001.

FRANÇA, T. C. de et al. Web das coisas: conectando dispositivos físicos ao mundo digital. **Livro Texto de Minicursos-SBRC**, v. 2011, p. 48, 2011.

GALVÃO, T. F.; PANSANI, T. d. S. A.; HARRAD, D. Principais itens para relatar revisões sistemáticas e meta-análises: A recomendação prisma. **Epidemiologia e serviços de saúde**, SciELO Public Health, v. 24, p. 335–342, 2015.

HAIDINE, A. et al. The role of communication technologies in building future smart cities. In: ____. [S.l.: s.n.], 2016. ISBN 978-953-51-2807-6.

HASSANZADEH, A.; MODI, S.; MULCHANDANI, S. Towards effective security control assignment in the industrial internet of things. In: IEEE. **2015 IEEE 2nd World Forum on Internet of Things (WF-IoT)**. [S.l.], 2015. p. 795–800.

HEJAZI, H. et al. Survey of platforms for massive iot. p. 1–8, 2018.

INGHAM, M.; MARCHANG, J.; BHOWMIK, D. Iot security vulnerabilities and predictive signal jamming attack analysis in lorawan. **IET Information Security**, IET, v. 14, n. 4, p. 368–379, 2020.

JAIDKA, H.; SHARMA, N.; SINGH, R. Evolution of iot to iiot: Applications & challenges. In: **Proceedings of the International Conference on Innovative Computing & Communications (ICICC)**. [S.l.: s.n.], 2020.

JR, G. P. et al. Towards firmware analysis of industrial internet of things (iiot) - applying symbolic analysis to iiot firmware vetting. In: . [S.l.: s.n.], 2017. p. 470–477.

KARNOUSKOS, S. Stuxnet worm impact on industrial cyber-physical system security. In: IEEE. **IECON 2011-37th Annual Conference of the IEEE Industrial Electronics Society**. [S.l.], 2011. p. 4490–4494.

KIM, D.-Y.; KIM, S. Lorawan technology for internet of things. **Journal of platform technology**, v. 3, n. 1, p. 3–8, 2015.

KOBARA, K. Cyber physical security for industrial control systems and iot. **IEICE TRANSACTIONS on Information and Systems**, The Institute of Electronics, Information and Communication Engineers, v. 99, n. 4, p. 787–795, 2016.

KORSTEN, M. et al. Industrial internet of things: unleashing the potential of connected products and services. **White Paper, in Collaboration with Accenture**, 2015.

KULIK, V.; KIRICHEK, R. The heterogeneous gateways in the industrial internet of things. In: IEEE. **2018 10th International congress on ultra modern telecommunications and control systems and workshops (ICUMT)**. [S.l.], 2018. p. 1–5.

LEBER, J. General electric's san ramon software center takes shape mit technology review. 2012. Disponível em: <https://www.technologyreview.com/2012/11/28/114725/general-electric-pitches-an-industrial-internet/>.

LEGAT, M. et al. Sistema de controle de acesso em iot. Florianópolis, SC., 2018.

LIN, S.-W. et al. Industrial internet consortium (iic) - the industrial internet of things volume g1: Reference architecture. 2017. Disponível em: <https://www.iiconsortium.org/IIC_PUB_G1_V1.80_2017-01-31.pdf>.

LOUKIL, S. et al. Analysis of lorawan 1.0 and 1.1 protocols security mechanisms. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 22, n. 10, p. 3717, 2022.

LUND, F. **Study of LoRaWAN Device and Gateway Setups: with ChirpStack implementation**. 2022.

MCPHERSON, R.; IRVINE, J. Secure decentralised deployment of lorawan sensors. **IEEE Sensors Journal**, IEEE, v. 21, n. 1, p. 725–732, 2020.

NAKAMURA, E. T.; RIBEIRO, S. L. A privacy, security, safety, resilience and reliability focused risk assessment methodology for iiot systems steps to build and use secure iiot systems. In: IEEE. **2018 Global Internet of Things Summit (GIoTS)**. [S.l.], 2018. p. 1–6.

NAOUI, S.; ELHDHILI, M. E.; SAIDANE, L. A. Enhancing the security of the iot lorawan architecture. In: IEEE. **2016 International Conference on Performance Evaluation and Modeling in Wired and Wireless Networks (PEMWN)**. [S.l.], 2016. p. 1–7.

NAOUI, S.; ELHDHILI, M. E.; SAIDANE, L. A. Trusted third party based key management for enhancing lorawan security. In: **2017 IEEE/ACS 14th International Conference on Computer Systems and Applications (AICCSA)**. [S.l.: s.n.], 2017. p. 1306–1313.

NIST. Framework for improving critical infrastructure cybersecurity. v. 1, n. 11, 2014.

PAL, A.; PURUSHOTHAMAN, B. **IoT technical challenges and solutions**. [S.l.]: Artech House, 2016.

PANCHAL, A. C.; KHADSE, V. M.; MAHALLE, P. N. Security issues in iiot: A comprehensive survey of attacks on iiot and its countermeasures. In: IEEE. **2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN)**. [S.l.], 2018. p. 124–130.

PANCHAL, A. C.; KHADSE, V. M.; MAHALLE, P. N. Security issues in iiot: A comprehensive survey of attacks on iiot and its countermeasures. In: IEEE. **2018 IEEE Global Conference on Wireless Computing and Networking (GCWCN)**. [S.l.], 2018. p. 124–130.

PEREIRA, A.; SIMONETTO, E. de O. Indústria 4.0: conceitos e perspectivas para o brasil. **Revista da Universidade Vale do Rio Verde**, v. 16, n. 1, 2018.

PIMENTEL, M.; FUCKS, H. **Sistemas Colaborativos**. Rio de Janeiro: [s.n.], 2013.

PÖTSCH, A.; HAMMER, F. Towards end-to-end latency of lorawan: Experimental analysis and iiot applicability. p. 1–4, 2019.

RALAMBOTIANA, M. **Key management with a trusted third party using LoRaWAN protocol: A study case for E2E security**. 2018.

RIBEIRO, A. J. d. J. **Problemas de Segurança na Internet das Coisas**. Tese (Doutorado), 2021.

RIBEIRO, V. et al. Enhancing key management in lorawan with permissioned blockchain. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 20, n. 11, p. 3068, 2020.

SAMONAS, S.; COSS, D. The cia strikes back: Redefining confidentiality, integrity and availability in security. **Journal of Information System Security**, v. 10, n. 3, 2014.

SANCHEZ-GOMEZ, J. et al. Secure authentication and credential establishment in narrowband iot and 5g. **Sensors**, v. 20, p. 882, 02 2020.

SANCHEZ-IBORRA, R. et al. Enhancing lorawan security through a lightweight and authenticated key management approach. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 18, n. 6, p. 1833, 2018.

SANTOS, L. et al. An architectural style for internet of things systems. In: **Proceedings of the 35th Annual ACM Symposium on Applied Computing**. [S.l.: s.n.], 2020. p. 1488–1497.

SCHRECKER, S. et al. Industrial internet consortium (iic) - the industrial internet of things volume g4: Security framework. 2016. Disponível em: <https://www.iiconsortium.org/pdf/IIC_PUB_G4_V1.00_PB-3.pdf>.

SEMTECH. An1200. 22 lora modulation basics. 2015.

SEMTECH. Lora® and lorawan®: A technical overview. 2019.

SILVA, G. et al. Estudo de viabilidade do uso de raspberry pi na névoa. SBC, Porto Alegre, RS, Brasil, p. 204–215, 2019. ISSN 0000-0000. Disponível em: <https://sol.sbc.org.br/index.php/wscad/article/view/8669>.

SISINNI, E. et al. Industrial internet of things: Challenges, opportunities, and directions. **IEEE transactions on industrial informatics**, IEEE, v. 14, n. 11, p. 4724–4734, 2018.

STELLIOS, I.; KOTZANIKOLAOU, P.; PSARAKIS, M. Advanced persistent threats and zero-day exploits in industrial internet of things. In: **Security and Privacy Trends in the Industrial Internet of Things**. [S.l.]: Springer, 2019. p. 47–68.

SULLIVAN, J. E.; KAMENSKY, D. How cyber-attacks in ukraine show the vulnerability of the us power grid. **The Electricity Journal**, Elsevier, v. 30, n. 3, p. 30–35, 2017.

TOZETTO, D. F.; SILVA, R. V.; FREIRE, R. Z. Sistema de comunicação lora: Uma solução para transmissão de dados no agronegócio envolvendo longas distâncias e baixo custo. In: **Simpósio Brasileiro de Automação Inteligente-SBAI**. [S.l.: s.n.], 2021. v. 1, n. 1.

TSAI, K.-L. et al. Secure session key generation method for lorawan servers. **IEEE Access**, IEEE, v. 8, p. 54631–54640, 2020.

TSAI, K.-L. et al. Low-power aes data encryption architecture for a lorawan. **IEEE Access**, IEEE, v. 7, p. 146348–146357, 2019.

WAGLE, S.; PECERO, J. E. Efforts towards iot technical standardization. In: SPRINGER. **International Conference on Ad-Hoc Networks and Wireless**. [S.l.], 2019. p. 524–539.

WANG, X. C. e Margaret Lech e L. A complete key management scheme for lorawan v1.1 †. **Sensors (Basel, Switzerland)**, v. 21, 2021.

WEISER, M. The computer for the 21st century.

WHITE, G. B.; SJELIN, N. The nist cybersecurity framework. In: **Research Anthology on Business Aspects of Cybersecurity**. [S.l.]: IGI Global, 2022. p. 39–55.

XING, J. et al. An improved secure key management scheme for lora system. In: **2019 IEEE 19th International Conference on Communication Technology (ICCT)**. [S.l.: s.n.], 2019. p. 296–301.

YU, X.; GUO, H. A survey on iiot security. In: **2019 IEEE VTS Asia Pacific Wireless Communications Symposium (APWCS)**. [S.l.: s.n.], 2019. p. 1–5.

ZENNARO, M. Introduction to the internet of things. **telecommunication and ICT4D Lab, The Abdus Salam International Centre for Theoretical Physics Trieste, Italy**, p. 1–48, 2017.

ZIMMERMANN, H. Osi reference model-the iso model of architecture for open systems interconnection. **IEEE Transactions on communications**, IEEE, v. 28, n. 4, p. 425–432, 1980.