UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
DAINF - DEPARTAMENTO ACADÊMICO DE INFORMÁTICA
CURSO DE BACHARELADO EM SISTEMAS DE INFORMAÇÃO

MATHEUS KENJI GLASSEY KAGEYAMA
VICTOR AUGUSTO DOMINGOS CALIXTO DE MENEZES

COMPARTILHAMENTO DE RECURSOS EM REDES LORA

TRABALHO DE CONCLUSÃO DE CURSO

CURITIBA
2019

MATHEUS KENJI GLASSEY KAGEYAMA
VICTOR AUGUSTO DOMINGOS CALIXTO DE MENEZES

COMPARTILHAMENTO DE RECURSOS EM REDES LORA

Proposta de Trabalho de Conclusão de Curso apresentado ao Curso de Bacharelado em Sistemas de Informação da Universidade Tecnológica Federal do Paraná, como requisito parcial para a obtenção do título de Bacharel.

Orientadora: Prof.ª Dra. Anelise Munaretto Fonseca
DAINF - Departamento Acadêmico de Informática -UTFPR

Coorientador: Prof. Dr. Mauro Fonseca
DAINF - Departamento Acadêmico de Informática -UTFPR

CURITIBA
2019

Ministério da Educação
UNIVERSIDADE TECNOLÓGICA FEDERAL DO PARANÁ
Câmpus Curitiba
Diretoria de Graduação e Educação Profissional
Departamento Acadêmico de Informática
Coordenação do Curso de Bacharelado em Sistemas de Informação

TERMO DE APROVAÇÃO

COMPARTILHAMENTO DE RECURSOS EM REDES LORA

por

Matheus Kenji Glassey Kageyama

Victor Augusto Domingos Calixto De Menezes

Este Trabalho de Conclusão de Curso foi apresentado no dia 08 de Julho de 2019 como requisito parcial à obtenção do grau de Bacharel em Sistemas de Informação na Universidade Tecnológica Federal do Paraná - UTFPR - Câmpus Curitiba. O(a)(s) aluno(a)(s) foi(ram) arguido(a)(s) pelos membros da Banca de Avaliação abaixo assinados. Após deliberação a Banca de Avaliação considerou o trabalho _____.

_____ **Profa. Anelise Munaretto Fonseca** (Presidente - UTFPR/Curitiba)	_____ **Profa. Ana Cristina Barreiras Kochem Vendramin** (Avaliador(a) 1 - UTFPR/Curitiba)
_____ **Prof. Guilherme Luiz Moritz** (Avaliador 2(a) - UTFPR/Curitiba)	_____ **Profa. Leyza Baldo Dorini** (Professora Responsável pelo TCC – UTFPR/Curitiba)
_____ **Prof. Marcelo Mikosz Goncalves** (Coordenador do curso de Bacharelado em Sistemas de Informação – UTFPR/Curitiba)	

“A Folha de Aprovação assinada encontra-se na Coordenação do Curso.”

RESUMO

KAGEYAMA, Matheus Kenji Glassey; MENEZES, Victor Augusto Domingos Calixto de. Compartilhamento de recursos em redes LoRa. 2019. 35 f. Trabalho de Conclusão de Curso – Curso de Bacharelado em Sistemas de Informação, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

Neste trabalho será realizado um estudo sobre a arquitetura e protocolo de comunicação LoRaWAN (Long Range Wide Area Network) relativo à Internet das Coisas com o objetivo de propor uma arquitetura que viabilize o compartilhamento do servidor de rede e do *gateway* em uma rede LoRa (Long Range, protocolo de camada física a ser utilizado pelo LoRaWAN). Esta pesquisa demonstra uma rede descentralizada e autoconfigurável, com o propósito de reduzir a quantidade de dispositivos necessários para ingressar em uma rede. A fim de viabilizar esta proposta, pretende-se utilizar como estratégia o IPv6 como identificador do servidor de aplicação, ao invés do padrão atual o qual não provê um identificador capaz de servir para o roteamento de mensagens entre servidores. Para isso, se faz necessário o estudo de estratégias para alteração da criação e gerenciamento das sessões de aplicação e rede utilizadas pelos servidores e sensores, além de modificações na forma de ingresso dos dispositivos a uma rede.

Palavras-chave: LoRa. LoRaWAN. IPv6.

ABSTRACT

KAGEYAMA, Matheus Kenji Glassey; MENEZES, Victor Augusto Domingos Calixto de. Resource sharing on LoRa networks. 2019. 35 f. Trabalho de Conclusão de Curso – Curso de Bacharelado em Sistemas de Informação, Universidade Tecnológica Federal do Paraná. Curitiba, 2019.

This paper presents a study about LoRaWAN (Long Range Wide Area Network) protocol and architecture relatives to Internet of Things, and their applications, with the objective to offer an architecture that enables gateway and network server sharing into a LoRa (Long Range, physical layer protocol that is utilized by LoRaWAN) network through IPv6. This research demonstrates a decentralised and autoconfigurable network architecture, with the purpose of reducing the amount of necessary devices to join a network. In order to make this proposal feasible, it is intended to use IPv6 as an application server identifier, instead of current pattern, which doesn't provides a routable identifier for messages between servers. Also, it is required to study some strategies to change the session management and creation in network devices communication.

Keywords: LoRa. LoRaWAN. IPv6.

LISTA DE FIGURAS

LISTA DE ABREVIATURAS E SIGLAS

IoT	*Internet of Things*, do inglês, internet das coisas
IP	*Internet Protocol*
IPv6	*Internet Protocol version 6*
LoRa	*Long Range*
LoRaWAN	*Long Range Wide Area Network*
LPWAN	*Low Power Wide Area Network*
OTAA	*Over The Air Activation*, do inglês, ativação pelo ar
TTN	*The Things Network*

SUMÁRIO

1 INTRODUÇÃO

LoRa (AUGUSTIN et al., 2016), um acrônimo para *Long Range* (Longo Alcance), é um protocolo de camada física que se popularizou pelas promessas de baixo custo, longo alcance e baixo consumo (AKPAKWU et al., 2018). LoRaWAN é a arquitetura de rede de acesso do protocolo LoRa, que segundo a LoRa Alliance (2015) define os protocolos de comunicação da camada de enlace e sua estrutura (ALLIANCE, 2015).

O avanço nas pesquisas em Internet das Coisas (IoT) (Gartner, Inc., 2017) e o rápido desenvolvimento da tecnologia LoRa vem impulsionando o mercado na área de cidades inteligentes. O crescimento da tecnologia LoRa pode ser observado na quantidade de publicações encontradas com a palavra-chave "LoRa" no site IEEE Xplore, repositório de artigos e revistas do IEEE. Nos anos de 2015, 2016, 2017 e 2018 houve, respectivamente, 20, 47, 186 e 307 trabalhos publicados. Segundo o instituto de pesquisa Gartner, Inc. (2017), até o final de 2017, 8,4 bilhões de "Coisas" estariam conectadas, em um crescimento de 31% em relação a 2016. Ainda segundo Gartner, Inc. (2017), o ramo de IoT movimentou $2 trilhões de dólares no ano de 2017.

A tecnologia LoRaWAN possui grande potencial a ser explorado, porém cada usuário que deseja experimentar esta tecnologia necessita implementar a infraestrutura completa, incluindo todos os dispositivos, como sensor/nó final, *gateway* e servidor (AUGUSTIN et al., 2016). Algumas soluções foram propostas a fim de flexibilizar a utilização dessa tecnologia (BLENN; KUIPERS, 2017). Como é proposto na The Things Network (TTN), há a possibilidade do usuário possuir apenas o nó final, se cadastrar no site da TTN e utilizar a infraestrutura de outro usuário. Como cita Blenn e Kuipers (2017), a TTN é uma rede financiada por alguns voluntários que disponibilizam seus *gateways*. O grande problema da TTN é que esta é uma rede centralizada e hierárquica com pontos de falha e todo o fluxo de conexões utilizando a rede de uma organização. Além disso, há a necessidade de cadastrar cada dispositivo na rede TTN. Estes problemas serão discutidos na Seção 2.

O trabalho a ser desenvolvido neste projeto visa propor uma forma de compartilhamento de recursos de infraestrutura (*gateway* e servidor de rede) e infraestrutura de redes LoRa descentralizada, não necessitando que os pacotes sejam encaminhados para um servidor central e cadastrado, mas sim que seja possível uma autoconfiguração entre o servidor desta rede e um sensor que não pertença à determinada rede. A importância desta pesquisa se justifica pela flexibilidade e independência que ela pode trazer para a tecnologia LoRa. Para exemplificar, analisa-se a situação a seguir: uma bicicleta com um sensor LoRa de geoposicionamento que venha a ser furtada, poderá ter sua localização monitorada em tempo real, pois para cada *gateway* pelo qual esta bicicleta passar, independentemente do mesmo pertencer à rede do proprietário ou não, a posição será atualizada.

Como pode ser observado em Centenaro et al. (2016), um pacote sai do sensor e utiliza-se do *gateway* para chegar ao servidor onde se verifica que o pacote pertence à rede ou não. Assim, na hipótese do pacote não pertencer àquela rede, ele será descartado. Com isso, pode-se observar que os recursos do *gateway*, dispositivos esses que não possuem tanto poder de processamento quanto um servidor, já foram utilizados. Isso porque caso o pacote tenha sido descartado no servidor, ele teve que passar de qualquer forma por este nó intermediário. Por esta razão, este trabalho visa otimizar os recursos utilizados. Sabendo que o tamanho dos pacotes é determinado pela taxa de envio de dados da rede e das especificações regionais onde a rede se situa (THIELEMANS; BEZUNARTEA; STEENHAUT, 2017), os pacotes provindos de redes LoRa são pequenos e, de forma geral, os servidores têm alto poder de processamento. Assim sendo, o impacto causado nos servidores tende a ser ínfimo comparado ao impacto no *gateway*. Portanto, a principal ideia é maximizar a utilização dos recursos do *gateway*.

1.1 Objetivos

Esta seção define os objetivos gerais e específicos deste trabalho com o intuito de explicitar o que será realizado nos próximos capítulos.

1.1.1 Objetivo geral

Analisar métodos atuais de roteamento de pacotes em redes LoRaWAN e propor um modelo de arquitetura descentralizada e autoconfigurável.

1.1.2 Objetivos específicos

Como objetivos específicos deste trabalho incluem-se investigar o protocolo utilizado para encaminhamento de pacotes em redes LoRa e propor alterações no formato de roteamento atual, elaborado por Augustin et al. (2016), através da utilização de um endereço IPv6 como identificador do nó destino da mensagem proveniente de um sensor associado a uma rede LoRa.

1.2 Organização do Documento

O Capítulo 1 apresentou o contexto, as motivações e os objetivos deste trabalho. O Capítulo 2 apresenta o que há na literatura relacionada com este trabalho e o estado da arte. No Capítulo 3 é abordada a metodologia deste trabalho. A arquitetura atual do LoRaWAN pode ser encontrada no Capítulo 4. A arquitetura proposta neste trabalho está descrita no Capítulo 5. Por fim, o Capítulo 6 retoma as contribuições e conclusões deste trabalho.

2 REVISÃO DE LITERATURA

Neste capítulo são apresentados o referencial teórico e o estado da arte.

2.1 Referencial teórico

Nesta seção são explanados os principais tópicos que circundam as redes LoRa, constituída por uma breve explicação de cada item e possíveis aplicações ou emprego em cenários reais. Além disso, são definidos conceitos importantes para o entendimento das demais partes integrantes deste trabalho.

2.1.1 IoT

A sigla IoT, referente à *Internet of Things*, pode ser entendida, segundo a definição de Gubbi et al. (2013), como sensores e atuadores interconectados que têm capacidade de, através de um *framework* (arquitetura) unificado, compartilhar informações entre diferentes plataformas. Já o grupo RFID (GUBBI et al., 2013) define IoT como uma rede global de objetos interconectados com endereçamento único e baseado em protocolos de comunicações.

Como Bardyn et al. (2016) afirmam, há uma gama de necessidades muito grande quando se trata de internet das coisas; Alguns cenários necessitam de poucos bps (bits por segundo), enquanto outros demandam Mbps (Megabits por segundo). O mesmo pode ocorrer com as distâncias, que variam de menos de um metro até quilômetros. Com isso, não é viável a utilização de apenas uma tecnologia. São necessárias várias para que seja possível cobrir o máximo de cenários. Akpakwu et al. (2018) apresentam diferentes cenários que se fazem presente de forma geral, como os ambientes inteligentes: casas, sistemas de transporte, cidades, serviços de saúde e indústria. Ao se tratar das tecnologias, é relevante segmentar em redes de longo alcance e de curto alcance (AKPAKWU et al., 2018). Para o primeiro caso, LoRa (AUGUSTIN et al., 2016), SigFox[1] e Ingenu-RPMA[2] são as mais comuns. Para as tecnologias de curto alcance, Bluetooth Low Energy (BLE) (GOMEZ; OLLER; PARADELLS, 2012), ZigBee (ZIGBEE, 2006) e WiFi são consideradas, segundo Akpakwu et al. (2018), as mais viáveis e populares.

Apesar das diferentes vertentes que uma rede IoT pode assumir, há requisitos que se fazem presentes na maioria dos cenários. Akpakwu et al. (2018) lista alguns fatores para analisar os protocolos utilizados atualmente para uma rede IoT, são eles: baixo custo dos dispositivos da rede, baixo custo de implantação, bateria de longa duração nos sensores, dispostos em lugares remotos e de difícil manutenção, ampla área de cobertura, segurança,

[1] https://www.sigfox.com/
[2] https://www.ingenu.com/technology/rpma/

privacidade e escalabilidade para que seja possível atingir uma grande quantidade de dispositivos conectados simultaneamente.

2.1.2 Low Power Wide Area Network (LPWAN) e LoRa

LPWAN é definido por aplicações de baixo gasto energético (*Low Power*) e com um longo alcance (*Wide Area*), tendo taxas de dados que variam normalmente entre 10bps até alguns kbps (BARDYN et al., 2016). Segundo Mikhaylov, Petaejaejaervi e Haenninen (2016), pode-se definir o desempenho desses sistemas através de três métricas: eficiência energética, escalabilidade e área de cobertura.

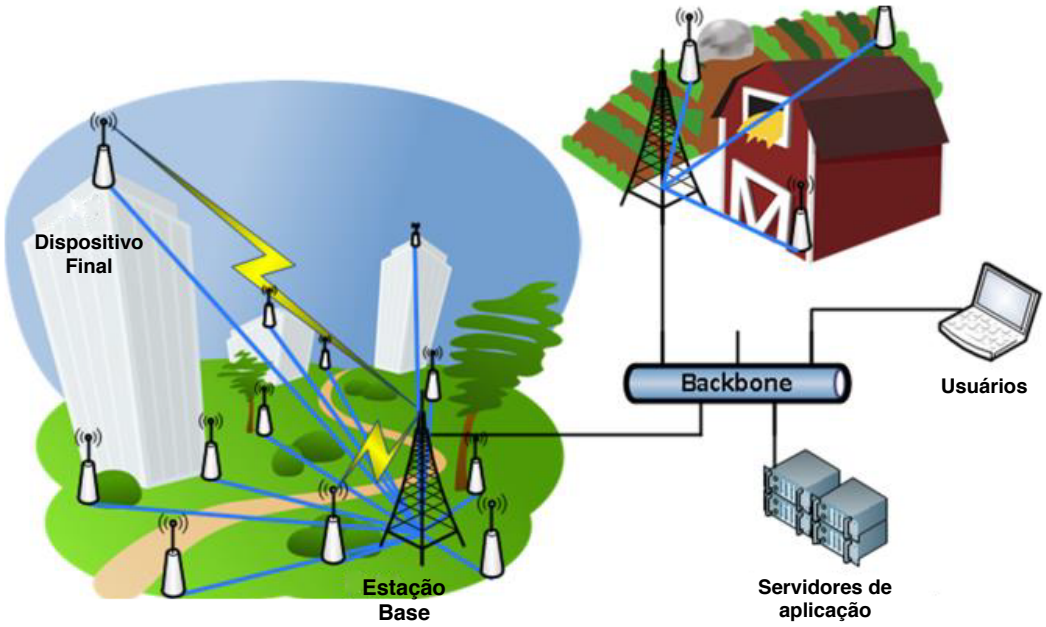
Essa definição faz com que LPWAN cubra um grande escopo de aplicações, porém de acordo com Bardyn et al. (2016) algumas características essenciais definem o projeto de uma rede LPWAN:

- Baixo consumo energético.
- Custo econômico reduzido através de dispositivos baratos, de fácil instalação na rede e pouca manutenção. Complexidade de *hardware* e *software* deve se manter simples.
- O objeto será ativado quando necessário, evitando assim gasto de energia.
- A infraestrutura de rede deve prever cobertura nacional e possibilidade de portabilidade entre países. A adição de infraestrutura ou um novo objeto deve ser simples e o protocolo deve ser padronizado, para possibilitar um maior número de objetos e desacoplamento.
- Transferência de dados entre objeto e usuário final precisa ser feita de forma segura.
- Na maioria das aplicações o objeto necessita ser facilmente localizado com baixo consumo de energia.
- Modulação deve ser robusta para suportar um possível enfraquecimento de sinal.
- Do ponto de vista da aplicação, os objetos coletarão dados para serem utilizados por uma grande variedade de serviços, diretamente ou através de um processo de fusão de informações e *machine learning* (aprendizado de máquina).

LPWANs são usualmente estruturadas como uma rede celular composta por dispositivos finais e estações base (BS), a exemplo da Figura 1. Os dispositivos finais se conectam às BS formando usualmente uma topologia estrela, que diferentemente da rede celular tradicional, possui um maior tráfego de *uplink* ao invés de *downlink* (MIKHAYLOV; PETAEJAEJAERVI; HAENNINEN, 2016). LoRaWAN é uma LPWAN, que se define basicamente pelo baixo gasto energético, mas proporcionando um longo alcance.

Comumente as redes LoRa utilizam-se de uma topologia estrela similar à tecnologia celular. Para o funcionamento da rede são necessários basicamente 3 tipos de dispositivos, sendo eles: o nó final, normalmente um sensor, o *gateway* e o servidor. O sensor tem a finalidade de coletar os dados, podendo ser temperatura, fumaça, umidade e inúmeros outros tipos, além da possibilidade de exercerem a função de atuadores. Com estes dados coletados, o nó final cria um pacote e o envia ao *gateway*. Este *gateway* é uma ponte entre

Figura 1 – Panorama típico de rede LPWAN.



Fonte: Adaptado de (MIKHAYLOV; PETAEJAEJAERVI; HAENNINEN, 2016)

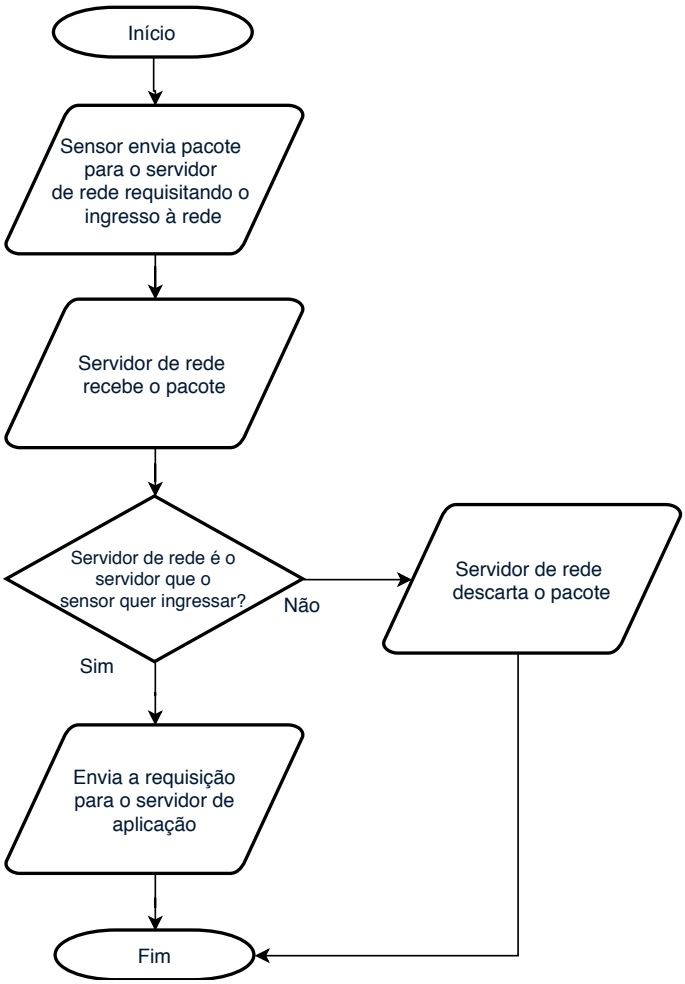
o servidor e o nó final, que basicamente recebe o pacote e o reencaminha ao servidor, independente se este for de um sensor conhecido pela rede ou não. Já o servidor, em posse dos dados, tem a função de gerenciar se os pacotes são duplicados e/ou de interesse para ele ou não. Atualmente, caso seja identificado que o pacote não tem o servidor como endereço de destino, este é descartado, senão o dado coletado é enviado ao servidor de aplicação (CENTENARO et al., 2016), a Figura 2 demonstra este fluxo de forma simplificada, vale ressaltar que a figura não mostra o fluxo completo, ela evidencia o momento que o servidor de rede descarta o pacote.

Os dispositivos finais são divididos em 3 classes: A, B e C. A classe A deve ser a classe padrão e mais utilizada dos dispositivos em uma rede LoRa, pois neste modo de operação os nós finais sempre iniciam a comunicação, que necessariamente é assíncrona. Este modo reduz muito o consumo de energia, porque o sensor inicia a comunicação em intervalos de tempo regulares, espera duas janelas de recepção para uma resposta restrita e após isso é desativado até que chegue o momento da próxima leitura.

A operação da classe A é ilustrada na Figura 3, onde sensores enviam suas mensagens, os *Gateways* recebem os sinais emitidos pelos sensores e enviam para os roteadores que realizam o roteamento para os *Broker Handlers*, responsáveis por redirecionar as mensagens as suas respectivas aplicações e realizar o gerenciamento de mensagens duplicadas. Por fim, a mensagem chega ao servidor de aplicação, o qual fará uso da informação contida no pacote enviado pelo sensor.

Para a classe B os dispositivos funcionam de forma similar à classe A, porém

Figura 2 – Fluxo de ingresso na rede.



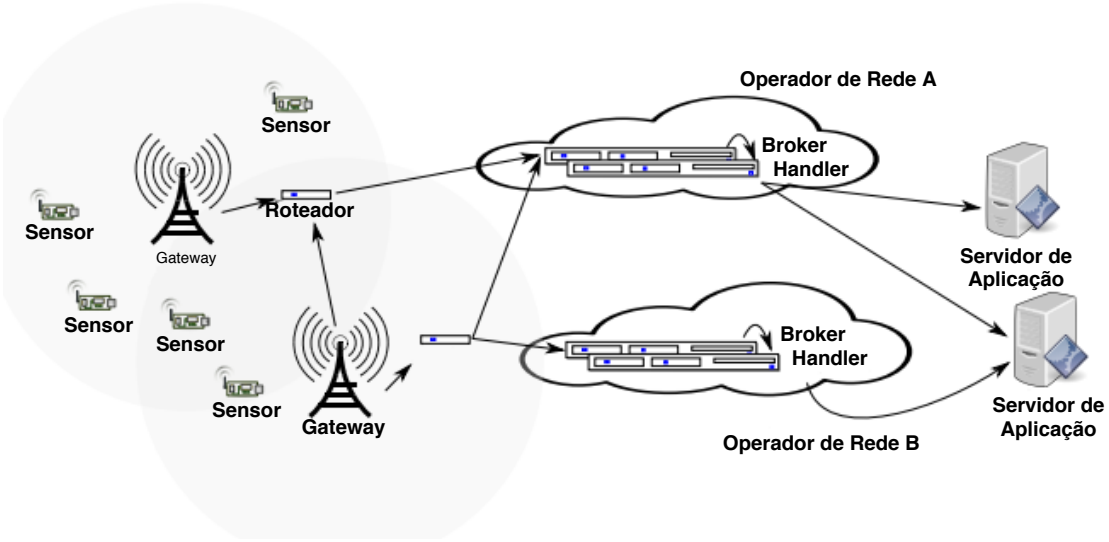
Fonte: Autoria própria.

a comunicação não é sempre iniciada pelo dispositivo final, dado que é possível criar uma tarefa para que o dispositivo fique ativo e possa ouvir em intervalos de tempo definidos (TOUSSAINT; RACHKIDY; GUITTON, 2016). Esta classe foi criada para tornar possível o recebimento de comandos através de uma comunicação remota. Vale ressaltar que o consumo energético desta classe é superior ao da classe A, dependendo de como é programado e da frequência desta tarefa que retira o sensor do modo desativado (ADELANTADO et al., 2017).

A classe C permanece ouvindo constantemente. É utilizada para dispositivos sem restrições de energia, que não necessitam de baterias e possuem uma fonte de energia constante. O dispositivo permanece ouvindo o tempo todo; nunca é desativado (CENTENARO et al., 2016; SOUZA; RABELLO, 2017).

A LoRa Alliance recomenda que, para a América, seja utilizada a faixa de frequência entre 902 a 928 MHz. Porém, pelo ato número 14448 (ANATEL, 2017), publicado em quatro de dezembro de 2017, a faixa entre 907,5 e 915 MHz possui utilização proibida. Assim, Souza e Rabello (2017) alertam para que a configuração seja corretamente feita para

Figura 3 – Rede LoRa.



Fonte: Adaptado de (BLENN; KUIPERS, 2017)

perfeito funcionamento da rede. Enquanto que na Europa, há a obrigação da utilização da faixa 863-870 MHz (SOUZA; RABELLO, 2017).

2.1.3 IPv6

O IPv6, definido na RFC 2460 em 1998, veio como sucessor do IPv4 com o intuito de aumentar a quantidade de IPs (*Internet Protocol*) disponíveis, simplificar o formato do cabeçalho, aumentar a possibilidade de opções e flexibilizações no cabeçalho para definir a qual fluxo cada pacote pertence (DEERING, 1998).

Um IPv6 pode ser dividido em duas partes, a parte de rede correspondendo aos 64 bits mais à esquerda e a parte de *host* correspondendo aos 64 bits mais à direita (CHOWN, 2008). Isto é, é possível identificar uma rede com 64 bits, enquanto os outros 64 bits são utilizados para identificar cada dispositivo dentro desta rede.

2.1.3.1 IPv6 para IoT

Como estudos da Cisco (2016) demonstram, a quantidade de usuários na internet ultrapassa os 4 bilhões e os dispositivos conectados alcançaram mais de 17 bilhões em quantidade; valores que justificam o esgotamento de endereços IPv4, que atingem a casa dos 2^{32}, pouco mais de 4 bilhões.

O IPv6 se mostra ideal para IoT, visto que há 2^{128} endereços disponíveis, significando mais de 340 undecilhões de endereços possíveis, não havendo necessidade da utilização de endereços falsos e *Network Address Translation* (NAT), como ocorre atualmente no IPv4 (ZIEGLER et al., 2013).

2.2 Estado da Arte

Com a rápida evolução do IoT diversas novas tecnologias foram desenvolvidas nos últimos anos como LoRa, SIGFOX[3], e Weightless[4], apesar de possibilitarem estruturas com baixo custo energético/monetário e fácil utilização, nenhuma delas possui suporte nativo ao IPv6 (WEBER et al., 2016).

Em contrapartida, alguns estudos já vem propondo a utilização do IPv6 em redes LoRa, como o caso do Weber et al. (2016), que desenvolveu e realizou experimentos com o IPv6 propondo o protocolo 6LoRaWAN, uma adaptação do protocolo 6LoWPAN que será visto na Seção 2.2. Porém, este artigo propôs uma conexão IPv6 ponto a ponto, assim, o sensor também possui um endereço IP, algo que não é necessário na arquitetura proposta por este trabalho, dado que há o enfoque no identificador do servidor de aplicação e não do sensor.

O protocolo IPv6 necessita de uma MTU (*Maximum Transmission Unit*, em português Unidade Máxima de Transmissão) de no mínimo 1280 bytes, o que impossibilitaria a implementação da rede. Através de técnicas de compressão definidas em Thubert e Hui (2011), os autores conseguiram alcançar um cabeçalho de apenas 60 bytes, o que apresenta um bom resultado dado que, o tamanho dos pacotes a serem transmitidos pela rede LoRa deve ser pequeno. Para alcançar o tamanho de pacote desejado os autores consideraram que algumas informações seriam padrões nas redes a utilizarem o IPv6 comprimido, como a versão, classe de tráfico e o rótulo de fluxo. Também reduziram algumas informações como o limite de saltos de 255 para 64.

Os resultados alcançados são bastante satisfatórios, pois foi possível desenvolver uma comunicação entre os sensores através de uma rede criada pelos pesquisadores, conforme Figura 4.

Uma outra pesquisa conduzida por Thubert e Hui (2011) utiliza-se do IPv6 comprimido para a criação da arquitetura 6LoWPAN, a qual possibilita que pacotes IPv6 sejam transportados em redes sem fio de baixo custo energético. Dessa forma tornam-se possíveis conexões entres sensores/sensores e sensores com servidores de forma descentralizada (MULLIGAN, 2007).

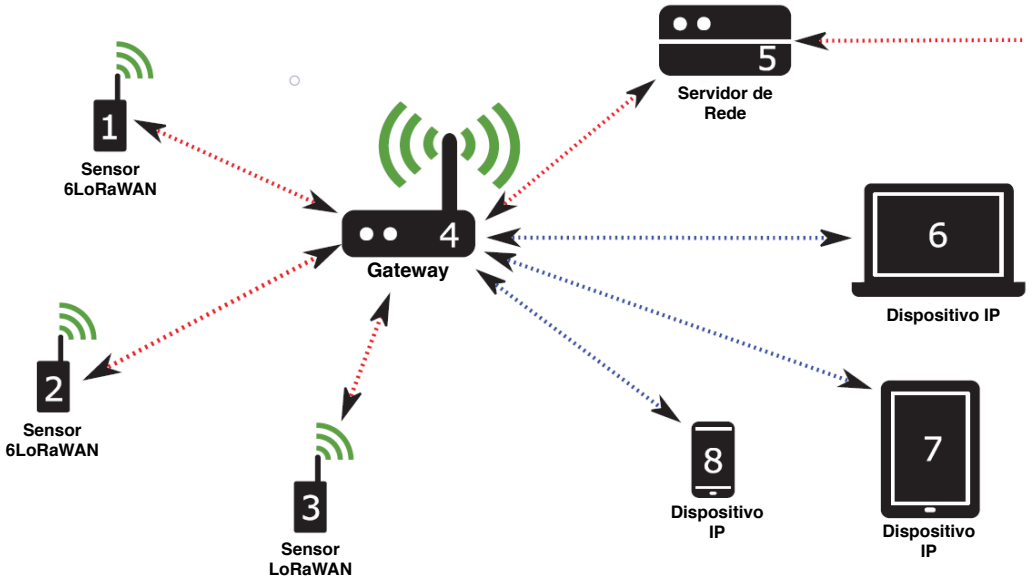
2.2.1 The Things Network

A The Things Network (TTN) é um projeto em escala global com caráter *crowd-funded*[5] o qual desenvolve uma rede LoRa que pode ser utilizada sem custos através de *gateways* disponibilizados por voluntários (BLENN; KUIPERS, 2017).

A topologia da rede empregada no TTN[6] segundo o seu site oficial, busca constituir

[3]https://www.sigfox.com/en
[4]http://www.weightless.org/
[5]Financiado pela multidão em tradução literal.
[6]https://www.thethingsnetwork.org

Figura 4 – Arquitetura da rede 6LoRaWAN.



Fonte: (WEBER et al., 2016)

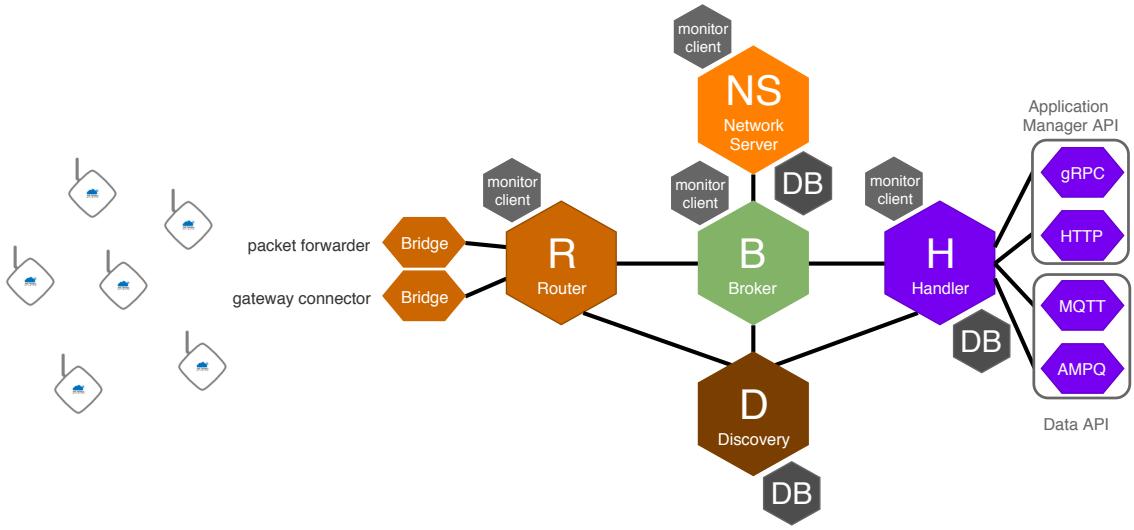
um sistema de *backend* responsável pelo roteamento de mensagens entre os dispositivos e as aplicações, para que as mensagens sejam entregues, embora o *gateway* com o qual o nó sensor se conecte em determinado instante, não pertença ao mesmo dono.

Redes que não utilizam o protocolo IP como LoRa necessitam que seja implementado um método de roteamento antes das mensagens chegarem aos servidores de aplicações. Assim, o objetivo principal é que seja criada uma rede descentralizada onde *Brokers* (Intermediário - Corretor) presentes na rede sejam responsáveis por mapear uma mensagem de um dispositivo para sua devida aplicação.

Segundo a Figura 5 observam-se alguns componentes importantes para o funcionamento da rede, os quais se diferenciam de uma rede LoRa comum (NETWORK, 2018):

- *Broker*: realiza o redirecionamento das mensagens provenientes dos dispositivos às suas respectivas aplicações, assim como o gerenciamento de mensagens duplicadas e a escolha de melhor rota para a resposta.
- *Network server* (servidor de rede): responsável pelo gerenciamento dos endereços dos dispositivos e o roteamento no nível LoRaWAN, como por exemplo a adição de comandos MAC às mensagens para transmissão a taxas mais altas, baseado no número de *gateways* e suas respectivas forças de sinais.
- *Handler*: decodifica e converte as mensagens de forma que as mesmas sejam compreendidas pelas aplicações. Também codifica as respostas.
- *Discovery*: determina por onde o tráfego de mensagens deve ser encaminhado. Atualmente é implementado de forma centralizada e gerido pelo The Things Network

Figura 5 – Arquitetura de uma rede TTN.



Fonte: Adaptado de (NETWORK, 2018)

Foundation, possibilitando o controle de quais componentes podem "publicar" para quais servidores de aplicações.

- *DataBase* (DB, em português base de dados): indica onde há armazenamento de dados.

2.3 Conclusão do Capítulo

Neste capítulo foi discutido como pesquisas no ramo de IoT têm buscado novas formas de trafegar mensagens entre sensores e servidores, propondo maneiras que permitam o compartilhamento de equipamentos como a The Things Network, ou utilizem tecnologias consolidadas como o IPv6 em uma rede LoRaWAN, vide Weber et al. (2016). O Capítulo 3 apresenta a metodologia para alcançar o modelo descentralizado proposto neste trabalho.

3 METODOLOGIA

Como visto na Seção 2.2, existem alguns trabalhos que propõem a implementação de alterações nas redes LoRa convencionais para que seja possível a utilização de IPv6 para realizar o roteamento ou possuir uma infraestrutura descentralizada tornando possível a reutilização de *gateways*, como é o caso do trabalho de (WEBER et al., 2016).

Neste trabalho entretanto, é proposta uma arquitetura que utilizará o IPv6 nas redes LoRa, apenas como um identificador do servidor de aplicação, sendo o servidor de rede responsável por realizar o roteamento até os servidores de aplicações, com a utilização do protocolo IP, permitindo assim que seja possível reutilizar os *gateways* sem a necessidade de uma estrutura centralizadora para controle e roteamento como apresentado no *The Things Network*. Dessa forma busca-se minimizar o impacto e a quantidade de alterações necessárias.

3.1 Modelo atual

Parte proeminente deste trabalho é compreender o funcionamento detalhado das redes LoRa, suas trocas de mensagens e roteamento dos pacotes partindo do dispositivo final ao servidor de aplicação e vice-versa. Assim sendo, a descrição de todos os passos encontra-se no Capítulo 4 para que seja possível apontar e identificar as alterações necessárias e propostas no Capítulo 5, com o intuito de adaptar o modelo existente de roteamento ao descrito nesse trabalho.

Optou-se pela separação dos capítulos que tratam da descrição do modelo atual e a apresentação das alterações propostas, pela facilidade na leitura e compreensão das modificações.

3.2 Definição da arquitetura

Será analisada a implementação de uma arquitetura para trocas de mensagens entre o sensor e o servidor de aplicação LoRa. A arquitetura se baseará em aproveitar os primeiros 64 bits do IPv6 do servidor como um identificador único. Dessa forma, os sensores presentes em uma rede LoRa, ao possuírem este identificador do servidor de aplicação, poderão enviar a sua informação independentemente do *gateway* e servidor de rede que receberá o pacote. Lembrando que há a necessidade do dispositivo final iniciar a conexão, uma vez que só será possível realizar o tráfego reverso (servidor de aplicação para o sensor) após as trocas iniciais de mensagens através do identificador IPv6 do servidor armazenado no dispositivo. O servidor de aplicação não sabe em que rede o sensor está conectado até o momento em que ocorrem as primeiras trocas de mensagens.

A arquitetura deverá considerar algumas características, como o fato da rede LoRa disponibilizar um tráfego de dados limitados por pacote, pois a mesma visa o desempenho energético dos seus componentes. Dessa forma, estruturas como sessões e/ou tabelas de roteamento deverão ser utilizadas. Vale ressaltar que o servidor ficará responsável por toda concentração do processamento, tabelas e decisões.

Para desenhar o modelo será descrita a implementação de forma minuciosa, além da utilização de diagramas e fluxogramas para explicitar o funcionamento completo da rede com as respectivas modificações definidas nesse trabalho.

Questões referentes à segurança entre o dispositivo final e o servidor da aplicação não são contempladas nas especificações da rede LoRa e por isso não serão abordadas. A disponibilização e cobrança pelo uso dos recursos envolvidos no processo, como banda de internet ou utilização de servidores intermediários para roteamento dos pacotes também não serão apresentadas nesse projeto.

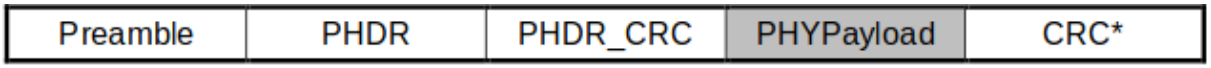
4 ARQUITETURA ATUAL

Neste capítulo serão descritos os modelos técnicos atuais de comunicação em uma rede LoRaWAN.

4.1 Formato das mensagens

Segundo a documentação da LoRa Alliance (2015), redes LoRaWAN utilizam formatos diferentes para mensagens de *uplink* e *downlink*. A Figura 6 representa um modelo de mensagem de *uplink*, contendo um preâmbulo, o cabeçalho da camada física e e o CRC (*Cyclic redundancy codes*) que é utilizado apenas em mensagens de *uplink* (Koopman; Chakravarty, 2004). O *payload* está descrito na seção 4.1.1.

Figura 6 – Formato da mensagem de camada física LoRaWAN para *uplink*.



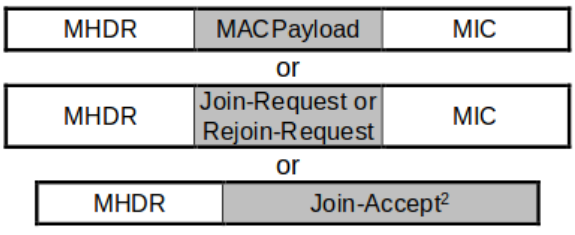
Fonte: (ALLIANCE, 2015)

Dentre os campos descritos na Figura 6, o **PHYPayload** está detalhado na Seção 4.1.1, o qual sofreu alterações para suportar o novo formato de roteamento proposto por esse trabalho.

4.1.1 Camada MAC (PHY *payload*)

Internamente ao PHY Payload há a possibilidade de três variações de mensagens (vide Figura 7), sendo a primeira utilizada para comunicação recorrente para troca de dados e as outras duas para ativação do dispositivo via OTAA, como explicado na Seção 4.3

Figura 7 – Formato da mensagem de camada MAC LoRaWAN.



Fonte: (ALLIANCE, 2015)

Figura 8 – Mensagem camada MAC com tamanho dos campos.

Size (bytes)	1	7..*M*	4
PHYPayload	MHDR	MACPayload	MIC

Fonte: (ALLIANCE, 2015)

Ainda em relação à mensagem na camada MAC, vale relembrar que o baixo custo energético em uma rede LoRa é algo importante, sendo essencial atentar-se para os pequenos tamanhos dos campos das mensagens, como ilustra a Figura 8.

O *Header* da mensagem da camada MAC, ilustrada pela Figura 9, possui o campo MType (*Message Type*) o qual possibilita a existência de 6 tipos de mensagens, sendo: *join request, join accept, unconfirmed data up/down* e *confirmed data up/down*. Além disso, há outros dois campos restantes, o RFU (*Reserved for Future Usage*) e *Major* que define a versão principal do protocolo. Futuramente é possível utilizar esse campo para identificar o novo padrão de roteamento de mensagens definido neste trabalho, a serem analisadas pelos servidores de rede com o intuito de redirecioná-las entre servidores IP.

Figura 9 – *Header* da mensagem (MHDR).

Bit#	7..5	4..2	1..0
MHDR bits	MType	RFU	Major

Fonte: (ALLIANCE, 2015)

4.2 Informações armazenadas no processo de ativação

Algumas informações precisam ser armazenadas no dispositivo para garantir a integridade e confidencialidade das mensagens, podendo ser armazenadas antes ou durante o processo de ativação do dispositivo, sendo:

- *End-device address* (DevAddr): identifica o dispositivo na rede com 32 bits, sendo os primeiros 7 para reconhecimento da rede e os outros 25 bits para o dispositivo final, podendo ser atribuído manualmente pelo gerenciador da rede.
- *Application identifier* (AppEUI): representa um identificador global no formato IEEE EUI64 identificando unicamente a entidade no processo de *JoinReq*. Esse identificador é armazenado previamente ao processo de ativação.
- *Network session key* (NwkSKey): chave de sessão específica para cada dispositivo, utilizada para calcular o MIC (*Message Integrity Code*), tanto pelo servidor de rede quanto pelo dispositivo final. Também serve para criptografia do campo de *payload* das mensagens na camada MAC.

- *Application session key* (AppSKey): tanto o servidor de aplicação quanto o dispositivo final usam a AppSKey para criptografia do campo de *payload* de dados da mensagem específica da aplicação. A chave não é utilizada para verificações de integridade.

A definição de cada um desses itens é importante para compreender o processo de ativação explicado na Seção 4.3.

4.3 Ativação do Dispositivo Final

Para ingressar em uma rede LoRaWAN, um nó necessita de uma ativação no servidor, havendo dois tipos, de acordo com a documentação da LoRa Alliance (2015), sendo através da Over-The-Air Activation (OTAA - ativação pelo ar) ou Activation By Personalization (ABP - ativação por personalização). A The Things Network (2016) recomenda a utilização da OTAA para se conectar à rede deles, devido a sua segurança. Desta forma, o foco da arquitetura proposta será a OTAA. Para viabilizar o trabalho proposto, é necessário alterar o funcionamento da ativação pelo ar.

4.3.1 Over-the-Air Activation (OTAA)

A OTAA possui 4 passos para que um nó seja incluído em uma rede (ALLIANCE, 2015):

- JoinReq: este passo é iniciado pelo nó final em direção ao servidor para ingressar em uma rede. Atualmente, são enviadas as seguintes informações para o servidor:
  - JoinEUI: Este EUI (*Extended Unique Identifier* - Identificador Único Estendido) é um identificador único do servidor contendo 64 bits. Pela arquitetura atual este identificador tem que ser inserido manualmente no sensor para que o servidor possa identificá-lo como parte integrante da rede no momento do processo de *Join*. O campo JoinEUI será alterado posteriormente para a arquitetura funcionar da forma prevista.
  - DevEUI: como indicado pelo nome, este item também é um Identificador Único Estendido com 64 bits, porém neste caso ele identifica o sensor de forma única.
  - DevNonce: este item contém 16 bits. Iniciando em 0 ele identifica cada chamada de *JoinRequest*. Assim, como descrito em Alliance (2015), caso o mesmo JoinEUI seja utilizado com o mesmo valor de DevNonce, o servidor descartará esta tentativa de inserção na rede. Desta forma, o campo DevNonce é incremental e deve ser armazenado no dispositivo.
- JoinAccept: caso o servidor consiga validar todas as informações recebidas no JoinReq, este retornará um pacote ao nó final chamado JoinAccept, informando ao dispositivo que ele tem permissão para pertencer à rede. Esta mensagem contém os seguintes campos:
  - JoinNonce: Este campo funciona de forma semelhante ao campo DevNonce,

porém este identificador agora representa o identificador do servidor. Ele contém 24 bits e não pode se repetir. Este campo é importante devido à criptografia das mensagens, dado que as chaves de sessão serão derivadas — juntamente com outras informações — do JoinNonce.
  - Home_netID: identificador único atribuído pela LoRa Alliance (2015) contendo 24 bits que identifica a rede à qual o dispositivo final pertence.
  - DevAddr: identificador do nó para a rede atribuído dinamicamente, assim que o nó é inserido na rede. Esse identificador será a forma pela qual o dispositivo será reconhecido pelo servidor, necessitando de 32 bits para a identificação de um nó na rede.
  - Há também os parâmetros DLSettings (contém configurações de *downlink*, 1 byte) e RxDelay (atraso entre receptor e transmissor, 1 byte) e de forma opcional o parâmetro CFList (lista de parâmetros da rede, 16 bytes).
- RekeyInd: este comando é utilizado para confirmar a atualização da chave de segurança. Além disto, o *RekeyInd* é utilizado para negociar a versão *minor* do protocolo LoRaWAN entre o servidor e o nó final. Isto ocorre apenas em versões superiores a 1.1 do LoRaWAN. Contém 1 byte, sendo os 4 últimos bits reservados para versão *minor* e os 4 primeiros reservados para uso futuro (*Reserved for Future Use* (RFU)).
- RekeyConf: este pacote contém 1 byte com a versão *minor* suportada pelo servidor seguindo o padrão de divisão igual ao pacote do *RekeyInd*, 4 bits para uso futuro e 4 para versão *minor*.

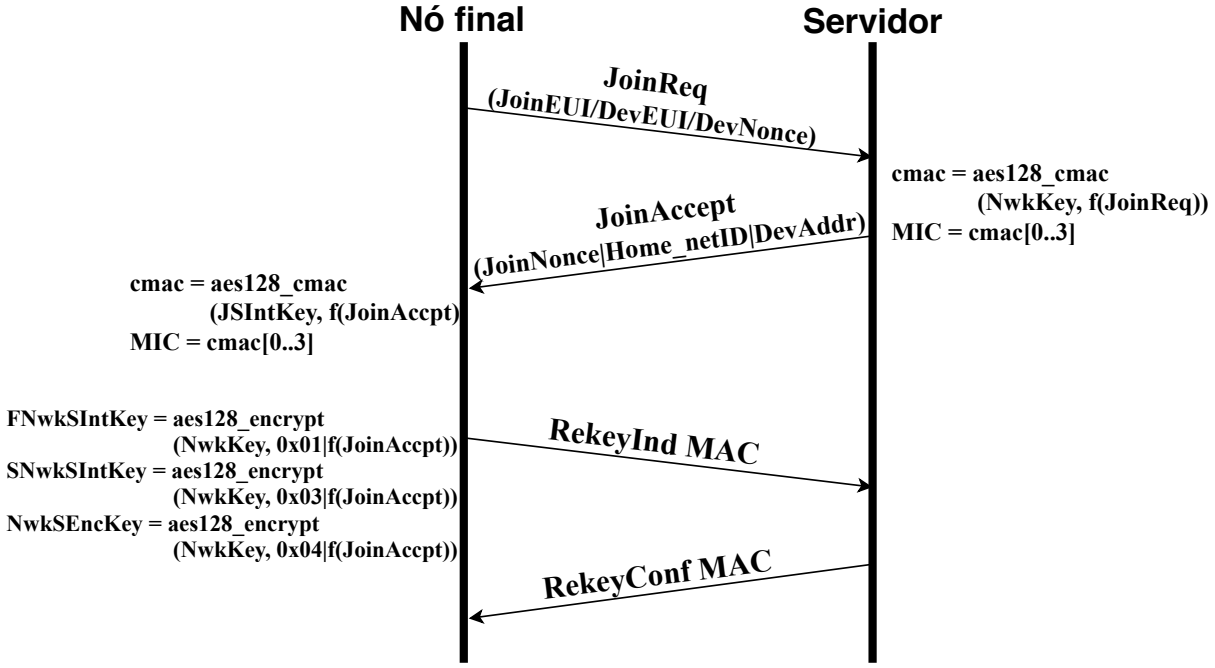
A Figura 10 mostra o fluxo de mensagens para a ativação em dispositivos OTA.

4.4 Contexto de Sessão

Existem dois tipos de sessões em uma rede LoRaWAN sendo a sessão de rede, entre o servidor de rede e o sensor, e a sessão de aplicação, entre o servidor de aplicação e o sensor. A primeira consiste nos seguintes itens:

- F/SNwkSIntKey: derivada da **NwkKey**, a qual é adicionada no dispositivo final durante a fabricação, serve como identificador da sessão aberta entre o servidor de rede e o dispositivo final.
- NwkSEncKey: derivada da **NwkKey**, a qual é adicionada no dispositivo final durante a fabricação, utilizada para realizar a criptografia das mensagens entre o servidor de rede e o dispositivo final.
- FCntUp: contador de quadros durante a comunicação entre o sensor e o servidor de rede, soma-se mais um ao campo anexando a cada envio de mensagem ao servidor, evita-se assim ataques de *replay*.
- FCntDwn (LW 1.0) ou NFCntDwn (LW 1.1): contador de quadros durante a comunicação entre o sensor e o servidor de rede, soma-se mais um ao campo anexando a

Figura 10 – Diagrama Join OTAA.



Fonte: Adaptado de (HAXHIBEQIRI et al., 2018)

cada envio de mensagem ao dispositivo final, evita-se assim ataques de *replay*.

- DevAddr: endereço do dispositivo durante a existência da sessão.

A segunda sessão consiste em:

- AppSKey: derivada da **AppKey**, adicionada no dispositivo durante a fabricação, serve como identificador da sessão aberta entre o servidor de aplicação e o sensor.
- FCntUp: contador de quadros durante a comunicação entre o sensor e o servidor de aplicação, soma-se mais um ao campo anexando a cada envio de mensagem ao servidor, evita-se assim ataques de *replay*.
- FCntDown (LW 1.0) ou AFCntDwn (LW 1.1): contador de quadros durante a comunicação entre o sensor e o servidor de aplicação, soma-se mais um ao campo anexando a cada envio de mensagem ao dispositivo final, evita-se assim ataques de *replay*.

4.5 Conclusão do Capítulo

Neste capítulo foi realizada uma descrição do funcionamento do protocolo atual, buscando prover um entendimento do modelo vigente incluindo os agentes presentes nos processos, os formatos das suas trocas de mensagens e as informações geradas e armazenadas durante o processo. Assim, auxiliando na compreensão das necessidades de modificações e os pontos que serão alterados e discutidos no Capítulo 5.

5 ARQUITETURA PROPOSTA

Neste capítulo são descritas as alterações propostas por este trabalho, com o intuito de possibilitar um roteamento de mensagens originadas do sensor e com destino ao servidor de aplicação utilizando o protocolo IPv6 entre o servidor de rede e o servidor de aplicação. Opta-se por esse formato decorrente do estudo realizado da arquitetura atual no Capítulo 4 e da análise de trabalhos correlatos discutidos no Capítulo 2, leva-se em conta que o modelo proposto busca ter uma implementação simples e que gere o menor número de alterações possíveis na arquitetura atual.

Vale ressaltar que assim como o funcionamento atual da rede LoRa, na arquitetura proposta é necessário que o dispositivo final faça o primeiro contato com os servidores para que seja iniciada uma conexão. Os servidores de aplicação não conseguirão alcançar o dispositivo final sem que este tenha enviado mensagens anteriormente. Dessa forma, a proposta de alteração contempla os dispositivos que ingressam em uma rede LoRa através do modelo de ativação via *Over-the-Air Activation* (OTAA), que terá as modificações descritas na seção a seguir.

5.1 Alterações na *Over-the-Air Activation* (OTAA)

Para possibilitar uma rede descentralizada e com comunicação através de um IPv6 é preciso compreender que apesar do IPv6 conter 128 bits, ou seja 16 bytes, ao adquirir um IPv6, na realidade a aquisição é de um intervalo de 64 bits. Ou seja, com apenas os 64 bits iniciais é possível garantir que todos os IPs de máscara 64 bits serão pertencentes ao mesmo proprietário do IPv6.

O IPv6 pode ser dividido em duas partes: rede e *host*, sendo 64 bits para rede e 64 bits *host*. Com isso, é possível alterar o campo JoinEUI (64 bits), enviando a parte de rede do IPv6, ou seja, os 64 primeiros bits, como pode ser visto na Figura 11. Em seguida, o servidor intermediário (aquele ao qual o nó está enviando o JoinReq) completará o IPv6 com ::10:5a, como pode ser visto na Equação 1, fazendo alusão à palavra lora em *leetspeak*, um estilo de escrito de forma a substituir letras por números (DEITRICK et al., 2012). Por exemplo, se o sensor deseja enviar um pacote para o IPv6 2001:db8::, o servidor intermediário completará este endereço com ::10:5a, vide Equação 2. Assim, o endereço do IPv6 de destino será 2001:db8::10:5a, como mostrado na Equação 3.

$$IPv6\_rede \parallel :: 10 : 5a \tag{1}$$

$$2001 : db8 :: \parallel :: 10 : 5a \tag{2}$$

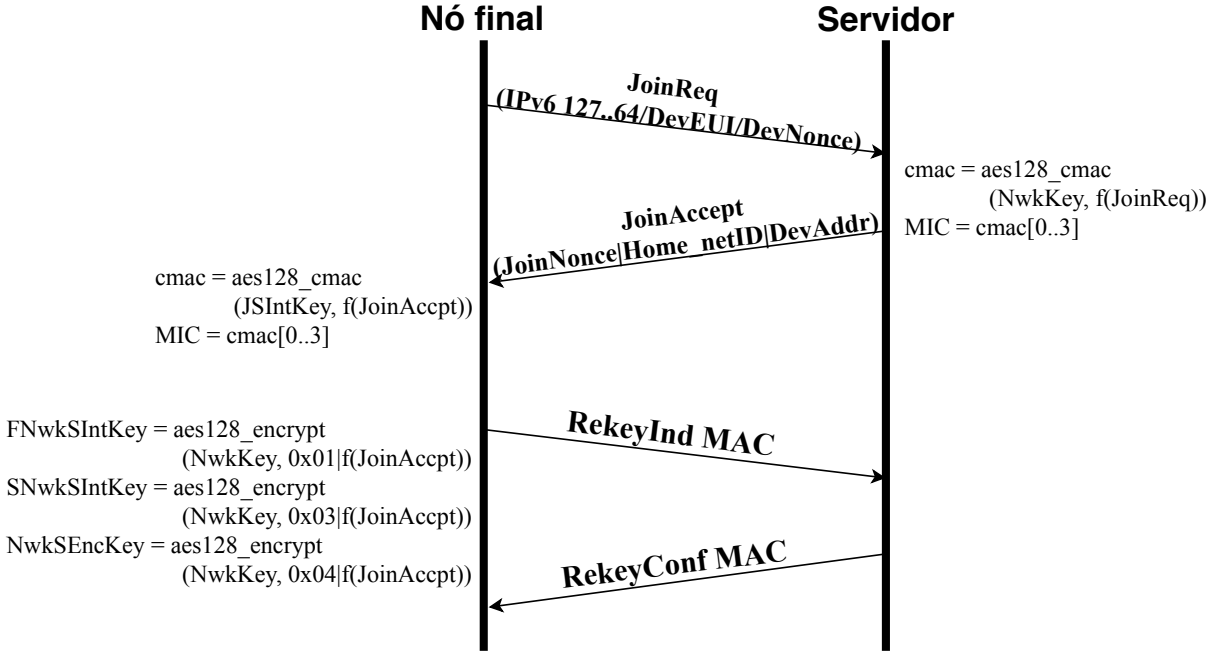
$$2001 : db8 :: 10 : 5a \tag{3}$$

Após isto, em posse do IPv6, o servidor intermediário enviará o pacote para a internet endereçando o IPv6 recebido, e o servidor final receberá os dados com a parte de rede junto do endereço ::10:5a. Há duas opções:

- O servidor final possuirá o IPv6 diretamente. Porém isto viabilizaria apenas um servidor LoRa por máscara 64 de IPv6.
- O servidor (*reverse proxy*) deve estar preparado para traduzir este IPv6 e encaminhar para um servidor LoRa ou um *cluster* de servidores LoRa.

Assim, com a parte de *host* padronizada, a arquitetura proposta se torna mais simples e factível para implementações.

Figura 11 – Diagrama Join OTAA com 64 bits do IPv6.

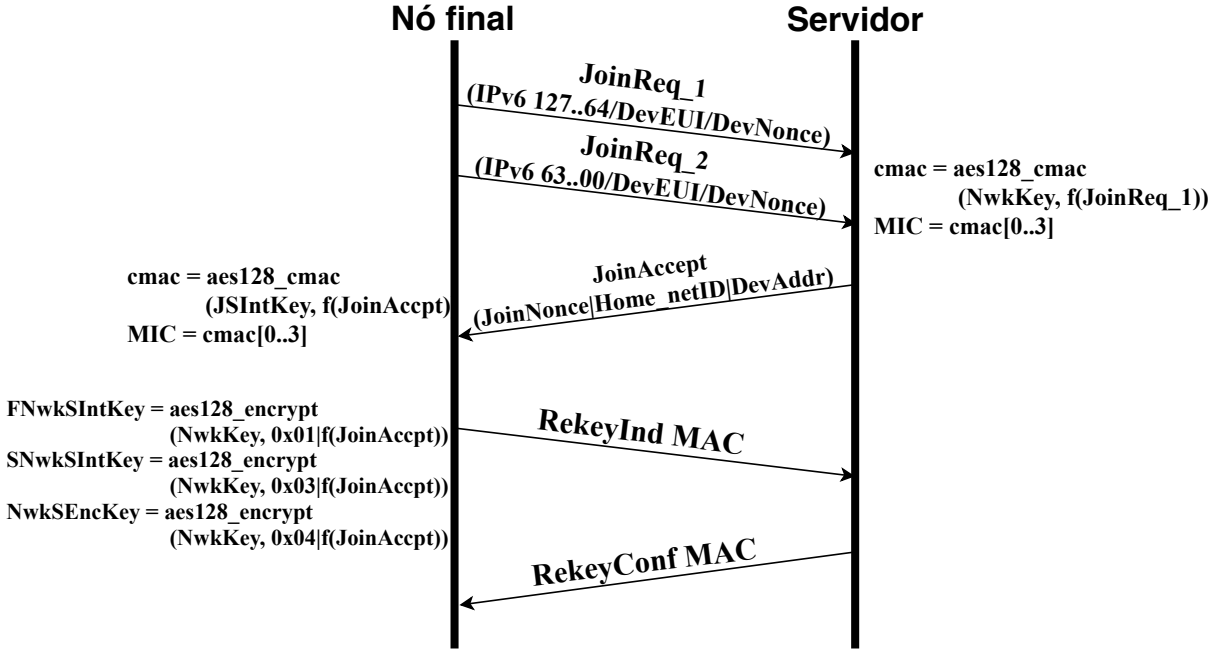


Fonte: Adaptado de (HAXHIBEQIRI et al., 2018)

Outra possibilidade mais complexa, porém viável, é manter a janela aberta para o JoinReq, para que seja possível que o sensor envie para o servidor dois pacotes de ingresso na rede, JoinReq1 e JoinReq2. O primeiro contendo a primeira parte do IPv6 e o segundo contendo a outra parte, como pode ser visto na Figura 12. Com isto, o servidor intermediário que encaminhará a mensagem para o servidor final não precisará completar o endereço com ::10:5a para enviar o pacote, mas sim terá o endereço de rede e de *host*.

As duas possibilidades se mostram viáveis. Porém a primeira necessita que o servidor final tenha implementado uma tradução de IPs para saber que o destino real

Figura 12 – Diagrama Join OTAA com 128 bits do IPv6.



Fonte: Adaptado de (HAXHIBEQIRI et al., 2018)

daquele pacote é o servidor LoRa. Na segunda, é necessário manter a janela aberta para receber dois pacotes ao invés de um.

A criptografia ocorrerá em dois momentos, com duas sessões abertas, tanto com o servidor que o nó conseguirá alcançar através do *gateway* quanto com o servidor ao qual ele realmente pertence. Com isso, pretende-se que o servidor intermediário não tenha acesso ao dados que estejam trafegando entre o dispositivo final e o servidor final.

Além da alteração supracitada, se faz necessário alterar outro item: o MHDR (*MAC Header*, em português, cabeçalho MAC). Como visto na Seção 4.1, o cabeçalho MAC possui três itens:

- MType: este campo corresponde ao tipo de mensagem possuindo 3 bits.
- RFU: o campo RFU possui 3 bits e é um campo reservado para que futuras implementações possuam flexibilidade e disponibilidade de bits para poder funcionar.
- Major: a versão *major* do protocolo, possui 2 bits.

A alteração se dará no último bit do campo RFU, que será utilizado da seguinte maneira, caso o bit de número 2 (último bit do campo RFU) do MHDR seja:

- 0: o servidor interpretará como um pacote de um nó final que está tentando ingressar na rede na forma atual, sem a utilização do IPv6, isto é, será descartado caso o servidor não o identifique como parte integrante da rede.
- 1: o servidor interpretará como um pacote que deseja utilizar o IPv6 para enviar os dados. Será verificado se o endereço IPv6 do servidor é igual ao endereço que o

sensor deseja enviar. Caso positivo, e esteja tudo correto no pacote, a conexão é aceita. Caso o IPv6 seja diferente do IPv6 do servidor, este pacote será encapsulado em um pacote HTTPS para ser enviado para a internet. Assim é possível atingir o servidor cujo sensor deseja enviar os dados.

Esta alteração torna a arquitetura compatível com as versões anteriores à versão 1.1 do protocolo LoRaWAN. Segundo a LoRa Alliance (2017), os bits do RFU são inseridos como 0, com isto, os servidores de versões anteriores não necessitarão de alteração para redes que não utilizam o IPv6. Isto é, continuará funcionando da forma atual.

5.2 Alterações no Contexto de Sessão

Neste trabalho é alterada a sessão entre a rede e o servidor descrita na Seção 4.4, para que além das informações atuais, também seja armazenado o último dígito do RFU, indicando assim se o servidor de rede deve utilizar o roteamento padrão do LoRaWAN ou se deve realizar um roteamento via IPv6 para um servidor terceiro. Os 64 bits do IPv6 são passados na requisição no campo de JoinEUI, onde dessa forma o servidor de rede redirecionará a mensagem de resposta proveniente do servidor de aplicação para o dispositivo final correto, com o auxílio da tabela de sessão entre o servidor de rede e o dispositivo final.

Dessa forma o novo padrão de *JoinRequest* para roteamento seguirá o fluxo definido na Figura 13. Os campos que serão adicionados estão demarcados em negrito.

Na primeira comunicação da Figura 13, o servidor de rede e o servidor de aplicação dispõe das informações trafegadas no *JoinRequest* para estabelecer a sessão que será utilizada nas trocas de mensagens futuras.

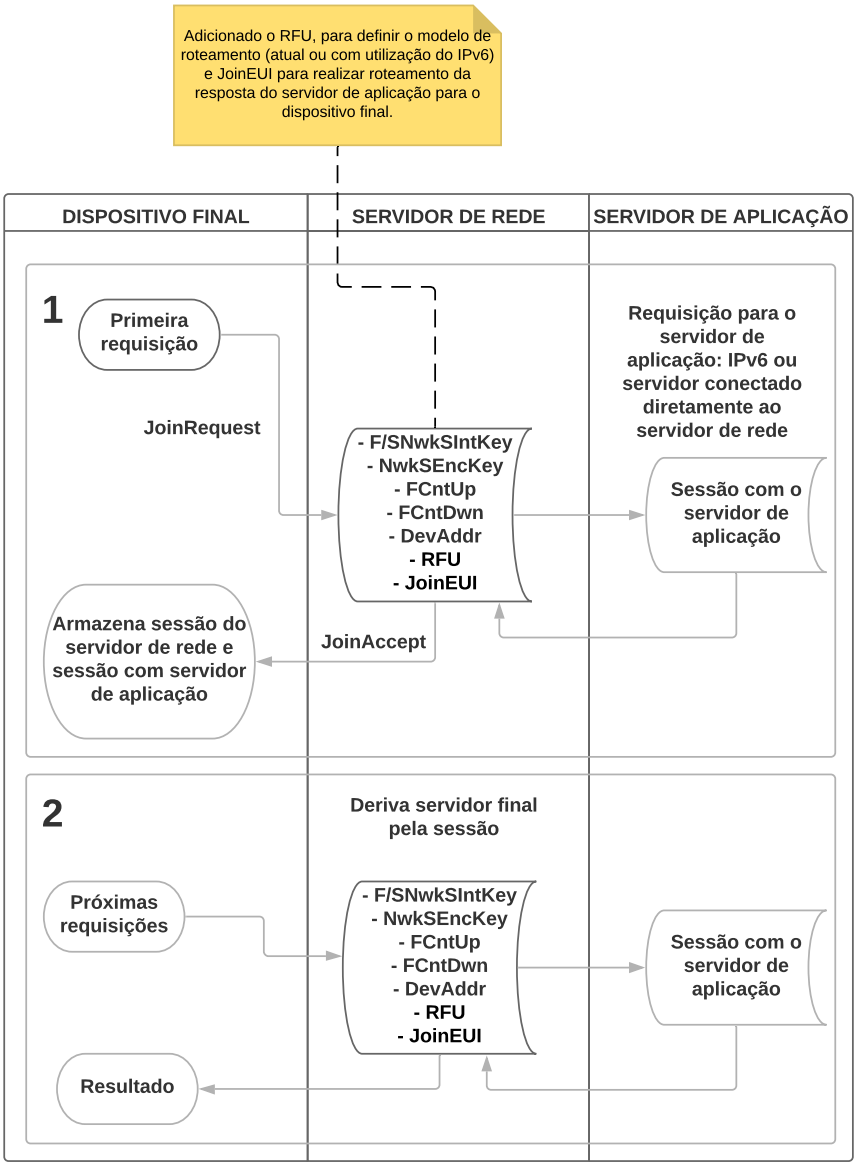
No fluxo 2 da Figura 13, ilustrando as requisições conseguintes, o roteamento para o servidor de aplicação de destino é realizado através dos dados armazenados na sessão do servidor de rede, o qual possui como chave para a requisição o **DevAddr** contido no *Frame Header* da requisição, como exemplifica a Figura 14.

5.3 Conclusões

Com as alterações propostas nesse trabalho pretende-se possibilitar uma nova forma de rotear os pacotes em uma rede LoRa, com a utilização do IPv6. Desse modo, os servidores de rede adquirem uma importância a mais, podendo agora definir novas rotas para servidores de aplicações que não estejam necessariamente ligados a eles fisicamente ou por conexões fora do padrão IP. Isto também possibilita uma rede mundial, descentralizada e autoconfigurável, dado que o servidor de aplicação pode estar em um país e o servidor de rede em outro.

As modificações sugeridas também foram pensadas com o objetivo de causar o menor impacto possível na arquitetura atual, mantendo os requisitos de uma rede IoT,

Figura 13 – Novo fluxo de *Join Request*.



Fonte: Autoria própria.

Figura 14 – *Frame Header* (FHDR).

Fonte: (ALLIANCE, 2015)

que preza por requisitos como baixo custo energético e banda limitada para tráfego de dados. Dessa forma, as conexões envolvendo dispositivos finais continuam a respeitar os limites de tamanho dos pacotes e não foram adicionadas novas requisições, para não

aumentar o *overhead* de sinalização com protocolos de *handshakes* adicionais ou interações semelhantes.

6 CONSIDERAÇÕES FINAIS

Para o desenvolvimento deste texto, foi realizada uma extensa revisão bibliográfica em redes IoT com ênfase na tecnologia LoRa, para que fosse possível conhecer o funcionamento das estruturas atuais e assim entender as necessidades e propor possíveis alterações.

O modelo apresentado neste trabalho tem por objetivo proporcionar a utilização de uma rede LoRaWAN de tal forma que seja possível compartilhar os dispositivos intermediários de uma conexão entre um sensor e um servidor de aplicação, assim sendo, *gateways* e servidores de rede poderiam ser reaproveitados para conexões de proprietários de dispositivos finais distintos, proporcionando um ganho de custo energético e de equipamentos e suas respectivas instalações.

Do ponto de vista teórico, conclui-se que com pequenas modificações — como a utilização de um identificador, baseado em IPv6, para o servidor de aplicação — é possível adicionar as informações e ferramentas de controle necessárias para que o novo modelo proposto funcione. Utilização de tecnologias consolidadas como o IPv6 permitem que o compartilhamento de recursos e o controle de tráfego sejam feitos de maneira padronizada globalmente. Dessa forma, reaproveita-se de uma estrutura vigente, diminuindo a quantidade de modificações necessárias em protocolos e equipamentos. A compatibilidade com versões anteriores também é extremamente importante, com isso, todas as alterações foram pensadas para que o fluxo atual continue a funcionar sem modificações ou adversidades.

Para trabalhos futuros, recomenda-se a implementação da arquitetura discutida em simuladores e ambientes reais de teste, contendo todos os integrantes de uma rede LoRa, como sensores, *gateways*, roteadores e servidores de rede e aplicação, com o intuito de validar este trabalho.

Referências

ADELANTADO, F. et al. Understanding the limits of LoRaWAN. **IEEE Communications Magazine**, IEEE, v. 55, n. 9, p. 34–40, 2017. Citado na página 14.

AKPAKWU, G. A. et al. A Survey on 5G Networks for the Internet of Things: Communication Technologies and Challenges. **IEEE Access**, v. 6, p. 3619–3647, 2018. Citado 2 vezes nas páginas 9 e 11.

ALLIANCE, L. LoRaWAN™ 1.1 Specification. **LoRa Alliance**, 2015. Citado 6 vezes nas páginas 9, 21, 22, 23, 24 e 30.

ALLIANCE, L. LoRaWAN™ Backend Interfaces 1.0 Specification. **LoRa Alliance**, 2017. Citado na página 29.

ANATEL. **Ato nº 14448**. 2017. <https://sei.anatel.gov.br/sei/publicacoes/controlador_publicacoes.php?acao=publicacao_visualizar&id_documento=2549681&id_orgao_publicacao=0>. Acessado em 23/06/2018. Citado na página 14.

AUGUSTIN, A. et al. A study of LoRa: Long range & low power networks for the internet of things. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 16, n. 9, p. 1466, 2016. Citado 3 vezes nas páginas 9, 10 e 11.

BARDYN, J. P. et al. IoT: The era of LPWAN is starting now. In: **ESSCIRC Conference 2016: 42nd European Solid-State Circuits Conference**. [S.l.: s.n.], 2016. p. 25–30. Citado 2 vezes nas páginas 11 e 12.

BLENN, N.; KUIPERS, F. LoRaWAN in the wild: Measurements from the things network. **arXiv preprint arXiv:1706.03086**, 2017. Citado 3 vezes nas páginas 9, 15 e 16.

CENTENARO, M. et al. Long-range communications in unlicensed bands: The rising stars in the IoT and smart city scenarios. **IEEE Wireless Communications**, IEEE, v. 23, n. 5, p. 60–67, 2016. Citado 3 vezes nas páginas 10, 13 e 14.

CHOWN, T. Ipv6 implications for network scanning. 2008. Citado na página 15.

CISCO. **Visual Networking Index Complete Forecast Highlights**. 2016. <https://www.cisco.com/c/dam/m/en_us/solutions/service-provider/vni-forecast-highlights/pdf/Global_Device_Growth_Traffic_Profiles.pdf>. Acessado em 10/05/2018. Citado na página 15.

DEERING, S. E. Internet protocol, version 6 (IPv6) specification. 1998. Citado na página 15.

DEITRICK, W. et al. Gender identification on twitter using the modified balanced winnow. **Communications and network**, Scientific Research Publishing, v. 4, n. 3, p. 189–195, 2012. Citado na página 26.

Gartner, Inc. **Gartner Says 8.4 Billion Connected "Things"Will Be in Use in 2017, Up 31 Percent From 2016**. 2017. <https://www.gartner.com/newsroom/id/3598917>. Acessado em 17/05/2018. Citado na página 9.

GOMEZ, C.; OLLER, J.; PARADELLS, J. Overview and Evaluation of Bluetooth Low Energy: An Emerging Low-Power Wireless Technology. **Sensors**, Molecular Diversity Preservation International, v. 12, n. 9, p. 11734–11753, 2012. Citado na página 11.

GUBBI, J. et al. Internet of Things (IoT): A vision, architectural elements, and future directions. **Future generation computer systems**, Elsevier, v. 29, n. 7, p. 1645–1660, 2013. Citado na página 11.

HAXHIBEQIRI, J. et al. A Survey of LoRaWAN for IoT: From Technology to Application. **Sensors**, Multidisciplinary Digital Publishing Institute, v. 18, n. 11, p. 3995, 2018. Citado 3 vezes nas páginas 25, 27 e 28.

Koopman, P.; Chakravarty, T. Cyclic redundancy code (CRC) polynomial selection for embedded networks. In: **International Conference on Dependable Systems and Networks, 2004**. [S.l.: s.n.], 2004. p. 145–154. Citado na página 21.

MIKHAYLOV, K.; PETAEJAEJAERVI, J.; HAENNINEN, T. Analysis of capacity and scalability of the LoRa low power wide area network technology. In: VDE. **European Wireless 2016; 22th European Wireless Conference; Proceedings of**. [S.l.], 2016. p. 1–6. Citado 2 vezes nas páginas 12 e 13.

MULLIGAN, G. The 6lowpan architecture. In: ACM. **Proceedings of the 4th workshop on Embedded networked sensors**. [S.l.], 2007. p. 78–82. Citado na página 16.

NETWORK, T. T. **Network Architecture**. 2018. <https://www.thethingsnetwork.org/docs/network/architecture.html>. Acessado em 09/07/2019. Citado 2 vezes nas páginas 17 e 18.

SOUZA, F. V. M. de; RABELLO, R. dos S. Desenvolvimento de um protótipo com utilização de LoRaWAN como Solução de Comunicação de Baixo Custo. 2017. Citado 2 vezes nas páginas 14 e 15.

The Things Network. **LoRaWAN Address Space**. 2016. <https://www.thethingsnetwork.org/docs/lorawan/address-space.html>. Acessado em 10/05/2019. Citado na página 23.

THIELEMANS, S.; BEZUNARTEA, M.; STEENHAUT, K. Establishing transparent IPv6 communication on LoRa based low power wide area networks (LPWANS). In: IEEE. **Wireless Telecommunications Symposium (WTS), 2017**. [S.l.], 2017. p. 1–6. Citado na página 10.

THUBERT, P.; HUI, J. W. Compression format for IPv6 datagrams over IEEE 802.15. 4-based networks. 2011. Citado na página 16.

TOUSSAINT, J.; RACHKIDY, N. E.; GUITTON, A. Performance analysis of the on-the-air activation in LoRaWAN. In: IEEE. **Information Technology, Electronics and Mobile Communication Conference (IEMCON), 2016 IEEE 7th Annual**. [S.l.], 2016. p. 1–7. Citado na página 14.

WEBER, P. et al. IPv6 over LoRaWAN^{TM}. In: **2016 3rd International Symposium on Wireless Systems within the Conferences on Intelligent Data Acquisition and Advanced Computing Systems (IDAACS-SWS)**. [S.l.: s.n.], 2016. p. 75–79. Citado 4 vezes nas páginas 16, 17, 18 e 19.

ZIEGLER, S. et al. IoT6 – Moving to an IPv6-Based Future IoT. p. 161–172, 05 2013. Citado na página 15.

ZIGBEE, A. Zigbee specification. **ZigBee document 053474r13**, 2006. Disponível em: <https://ci.nii.ac.jp/naid/10026813079/en/>. Citado na página 11.