UNIVERSIDADE FEDERAL DO CEARÁ

CAMPUS DE ITAPAJÉ

CURSO DE GRADUAÇÃO EM SEGURANÇA DA INFORMAÇÃO

THAÍS ADRIELLE FERREIRA RODRIGUES

VULNERABILIDADES EM PROTOCOLOS DE APLICAÇÃO PARA DISPOSITIVOS IOT

ITAPAJÉ

2025

THAÍS ADRIELLE FERREIRA RODRIGUES

VULNERABILIDADES EM PROTOCOLOS DE APLICAÇÃO PARA DISPOSITIVOS IOT

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Segurança da Informação do Campus de Itapajé da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de tecnólogo em Segurança da Informação.

Orientador: Prof. Dr. Israel Eduardo de Barros Filho.

ITAPAJÉ

2025

Dados Internacionais de Catalogação na Publicação
Universidade Federal do Ceará
Sistema de Bibliotecas
Gerada automaticamente pelo módulo Catalog, mediante os dados fornecidos pelo(a) autor(a)

R617v Rodrigues, Thaís Adrielle Ferreira.
Vulnerabilidades em protocolos de aplicação para dispositivos IoT / Thaís Adrielle Ferreira Rodrigues. – 2025.
80 f. : il. color.

Trabalho de Conclusão de Curso (graduação) – Universidade Federal do Ceará, Campus de Itapajé, Curso de Segurança da Informação, Fortaleza, 2025.
Orientação: Prof. Dr. Israel Eduardo de Barros Filho.

1. Internet das Coisas. 2. Segurança. 3. MQTT . 4. CoAP. 5. AMQP. I. Título.

CDD 005.8

THAÍS ADRIELLE FERREIRA RODRIGUES

VULNERABILIDADES EM PROTOCOLOS DE APLICAÇÃO PARA DISPOSITIVOS IOT

Trabalho de Conclusão de Curso apresentado ao Curso de Graduação em Segurança da Informação do Campus de Itapajé da Universidade Federal do Ceará, como requisito parcial à obtenção do grau de tecnólogo em Segurança da Informação.

Aprovada em: 26/02/2025.

BANCA EXAMINADORA

Prof. Dr. Israel Eduardo de Barros Filho (Orientador)
Universidade Federal do Ceará (UFC)

Prof. Dr. Aminadabe Barbosa de Sousa
Universidade Federal do Ceará (UFC)

Prof. Dr. Ítalo Linhares de Araújo
Universidade Federal do Ceará (UFC)

Dedico este trabalho aos meus pais, cujo amor incondicional e apoio incansável foram fundamentais para minha jornada. Sem vocês, esta conquista não seria possível. Obrigada por serem minha base, minha força e minha maior inspiração.

AGRADECIMENTOS

A Deus, pela força, sabedoria e oportunidades concedidas ao longo desta caminhada.

Aos meus pais, meus irmãos e minha sobrinha, pelo amor, apoio incondicional e por sempre estarem ao meu lado, tornando cada conquista ainda mais significativa.

À Larisse, por seu amor, paciência e por estar sempre ao meu lado, tornando cada desafio mais leve e cada conquista ainda mais especial.

Ao meu orientador Prof. Dr. Israel Eduardo de Barros Filho, por sua dedicação, ensinamentos e por guiar-me com tanta sabedoria durante este processo.

Ao Programa de Educação Tutorial (PET), na pessoa do Prof. Dr. João Henrique Côrrea, por proporcionar um ambiente de aprendizado e crescimento que contribuiu imensamente para minha formação.

Aos professores, pelo conhecimento compartilhado e pelo empenho em formar profissionais qualificados.

Aos técnicos administrativos, cujo trabalho muitas vezes discreto, mas essencial, garantindo o bom funcionamento da instituição e facilitando nossa jornada acadêmica.

Aos amigos, pelo apoio, pelas conversas, pelos momentos de descontração e pelo companheirismo ao longo dessa trajetória. Sem vocês, o caminho teria sido muito mais difícil.

E a todos que, de alguma forma, contribuíram para que este momento fosse possível, meu mais sinceros agradecimentos.

"A única maneira segura de armazenar dados é não tê-los." (Autor desconhecido.)

RESUMO

A Internet das Coisas (IoT) conecta dispositivos físicos, permitindo a coleta e troca de dados. Com o rápido crescimento do número desses dispositivos, surgem desafios de segurança devido a vulnerabilidades nos protocolos de comunicação. Com isso, este trabalho analisa as vulnerabilidades dos protocolos MQTT, CoAP e AMQP, destacando suas limitações em relação à criptografia, autenticação e resiliência a ataques cibernéticos. Foi realizada uma revisão da literatura com artigos publicados entre 2020 e 2024 para identificar ameaças comuns e estratégias de mitigação. As descobertas revelam que, embora o MQTT e o CoAP sejam leves e eficientes, carecem de mecanismos de segurança inerentes, enquanto o AMQP oferece melhor segurança mas é necessário utilizar mais recursos computacionais. Esta pesquisa ressalta a necessidade de estruturas de segurança aprimoradas na comunicação IoT.

Palavras-chave: Internet das Coisas; segurança; MQTT; CoAP; AMQP.

ABSTRACT

The Internet of Things (IoT) connects physical devices, allowing them to collect and exchange data. With the rapid growth in the number of these devices, security challenges arise due to vulnerabilities in communication protocols. This paper analyzes the vulnerabilities of the MQTT, CoAP and AMQP protocols, highlighting their limitations in terms of encryption, authentication and resilience to cyber attacks. A literature review was conducted with articles published between 2020 and 2024 to identify common threats and mitigation strategies. The findings reveal that while MQTT and CoAP are lightweight and efficient, they lack inherent security mechanisms, while AMQP offers better security but requires the use of more computational resources. This research highlights the need for improved security frameworks in IoT communication.

Keywords: Internet of Things; security; MQTT; CoAP; AMQP.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

AMQP	*Advanced Message Queuing Protocol*/Protocolo Avançado de Enfileiramento de Mensagens
CoAP	*Constrained Application Protocol*
DDoS	*Distributed Denial of Service*/Negação de serviço distribuída
DoS	*Denial of Service*/Ataque de Negação de Serviço
DTLS	*Datagram Transport Layer Security*
IoT	*Internet of Things*/Internet das Coisas
MITM	*Man in The Middle*/Homem do meio
MQTT	*Message Queuing Telemetry Transporte*/Protocolo de Mensagens de Telemetria de Fila
TCP/IP	*Transmission Control Protocol/Internet Protocol* / Protocolo de Controle de Transmissão/Protocolo da Internet

SUMÁRIO

1 INTRODUÇÃO

A Internet das Coisas (IoT) pode ser entendida como a rede ubíqua e global que ajuda e provê a funcionalidade de integrar o mundo físico. Isso se dá por meio da coleta, do processamento e da análise de dados gerados pelos sensores da IoT (ALBERTIN, 2017). Além disso, a Internet das Coisas pode ser descrita como uma rede que conecta dispositivos (ou coisas) exclusivamente identificáveis à internet, permitindo que eles coletem, enviem, armazenem e recebam dados (FILHO *et al.*, 2015).

Em todo o mundo, o número de dispositivos conectados à Internet das Coisas (IoT) está crescendo rapidamente. Em 2023, havia cerca de 15,9 bilhões de dispositivos de IoT, e esse número deve quase dobrar até 2030, chegando a mais de 32 bilhões. Em 2033, a China será o país com o maior número de dispositivos, com cerca de 8 bilhões. Hoje, a maioria desses dispositivos é usada em áreas de consumo, como *smartphones* e outros aparelhos eletrônicos, e espera-se que essa tendência continue pelos próximos anos (VAILSHERY, 2024).

Além disso, a IoT está se expandindo para setores como energia, transporte, varejo e até mesmo serviços públicos, com mais de 100 milhões de dispositivos conectados em áreas como eletricidade, gás e gerenciamento de resíduos. Esse crescimento mostra o quanto a IoT está se tornando essencial em diversas partes da nossa vida cotidiana (VAILSHERY, 2024).

De acordo com os dados, pode-se perceber que a Internet das Coisas (IoT) está revolucionando a forma como os dispositivos se conectam e compartilham informações, tornando-se uma parte essencial do nosso dia a dia. Desde casas inteligentes até sistemas industriais, a IoT permite que dispositivos diversos trabalhem em conjunto, trazendo mais praticidade e eficiência para pessoas e empresas.

No entanto, para que isso seja possível, é fundamental que exista uma comunicação eficiente e segura entre esses dispositivos. Os protocolos MQTT, CoAP e AMQP são amplamente utilizados na camada de aplicação da Internet das Coisas (IoT) devido às suas características específicas que atendem às necessidades de comunicação entre dispositivos. Esses protocolos desempenham um papel essencial para garantir o funcionamento adequado das aplicações conectadas.

Existem vários protocolos usados na comunicação entre dispositivos IoT, cada um com características próprias que atendem a necessidades diferentes. O MQTT (*Message Queuing Telemetry Transport*), por exemplo, é um protocolo bem leve e ideal para dispositivos com pouca capacidade de processamento e para redes com baixa largura de banda. Ele funciona no modelo

de publicação e assinatura e usa o protocolo TCP/IP, o que garante que as mensagens sejam entregues com confiabilidade.

O CoAP (*Constrained Application Protocol*), por sua vez, é focado em dispositivos com recursos limitados, assim como o MQTT. Ele usa o protocolo UDP, que é mais eficiente em termos de comunicação. O CoAP é parecido com o HTTP, pois também usa métodos como GET, POST, PUT e DELETE, o que facilita bastante sua integração com a web.

Já o AMQP (*Advanced Message Queuing Protocol*) é utilizado para organizar e gerenciar a troca de mensagens entre dispositivos. Ele funciona de maneira assíncrona, o que permite uma comunicação mais flexível e independente entre os dispositivos.

O objetivo principal deste trabalho é analisar as falhas de segurança em protocolos da camada de aplicação utilizados em dispositivos IoT, com foco em MQTT, CoAP e AMQP. Esses protocolos, amplamente adotados, apresentam vulnerabilidades que podem comprometer a privacidade e a integridade das informações, além de facilitar ataques, como negação de serviço (DDoS). Além de explorar os protocolos mais usados, como MQTT, CoAP e AMQP, este trabalho também tem o objetivo de identificar vulnerabilidades conhecidas e investigar como essas falhas podem ser exploradas e quais riscos representam, contribuindo para o aprimoramento da segurança em dispositivos IoT.

Diante disso, a escolha deste tema decorre da grande expansão do uso de dispositivos IoT em diversas áreas e as preocupações com segurança, pois esses dispositivos são vulneráveis a ataques devido a suas limitações técnicas. Este trabalho é, portanto, importante para identificar vulnerabilidades nesses protocolos e propor práticas de mitigação promovendo um uso mais seguro da tecnologia IoT.

Este trabalho adota uma metodologia baseada em revisão bibliográfica da literatura para analisar os protocolos MQTT, CoAP e AMQP na camada de aplicação da pilha TCP/IP em sistemas IoT, com foco em suas vulnerabilidades, ataques e possíveis mecanismos de segurança. A revisão foi realizada em bases científicas reconhecidas, incluindo IEEE *Xplore*, ACM *Digital Library*, *Scopus* e *Web of Science*, abrangendo publicações entre 2020 e 2024. O processo metodológico seguiu quatro etapas: coleta de dados, seleção de artigos com critérios de inclusão e exclusão, análise dos estudos e definição de questões centrais sobre a segurança e uso desses protocolos. Essa abordagem estruturada visa contribuir para o entendimento das ameaças e soluções de segurança em IoT.

O restante deste trabalho está organizado da seguinte forma: na Seção 2 serão

abordados conceitos relacionados à Segurança da Informação, Segurança em IoT, Internet das Coisas, Arquiteturas e Camadas IoT, além dos Protocolos de Aplicação, com foco em MQTT, CoAP e AMQP, e as vulnerabilidades e ataques comuns nesses protocolos. Na Seção 3 é apresentado a metodologia onde será apresentado os métodos e procedimentos adotados na pesquisa. Na Seção 4 os resultados, onde serão analisados os dados obtidos. E por fim, na Seção 5 será apresentado a conclusão e trabalhos futuros.

2 FUNDAMENTAÇÃO TEÓRICA

Nesta seção serão expostos alguns conceitos importantes relacionados a Segurança da Informação, Internet das Coisas, arquiteturas e camadas IoT, os protocolos de aplicação: MQTT, CoAP, AMQP, além de vulnerabilidades e ataques nos protocolos de aplicação em IoT.

2.1 Segurança da Informação

A Segurança da Informação é crucial na era digital, com o objetivo de resguardar as informações contra acessos não permitidos, corrupção e furto. A dependência cada vez maior de sistemas de informação requer que as organizações implementem ações eficientes para minimizar riscos e assegurar a confiabilidade, integridade e disponibilidade das informações. Este tópico discute os conceitos básicos, princípios, riscos e práticas recomendadas em relação à segurança da informação.

No contexto de Segurança da Informação, refere-se à adoção de ações para garantir a confidencialidade, integridade, disponibilidade e demais aspectos da segurança das informações dentro das necessidades do cliente.

Segurança da Informação pode ser entendida como a preservação da confidencialidade, integridade e disponibilidade da informação; adicionalmente, outras propriedades, tais como autenticidade, responsabilidade, não repúdio e confiabilidade, podem também estar envolvidas (ABNT, 2006).

Segurança da Informação como uma área do conhecimento dedicada à proteção de ativos da informação contra acessos não autorizados, alterações indevidas ou sua indisponibilidade (SÊMOLA, 2003).

Por sua vez, a Segurança de Informação caracteriza-se pela utilização apropriada de mecanismos de proteção sobre um ativo ou um grupo de ativos, com o objetivo de manter o valor que este tem para as instituições. A implementação dessas salvaguardas visa garantir a confidencialidade, integridade e disponibilidade (CID), abrangendo não apenas sistemas ou programas, mas também informações guardadas ou transmitidas em vários meios, além do eletrônico ou impresso (BASTOS; CAUBIT, 2009).

Os conceitos dos autores acima demonstram um aspecto comum entre eles: confidencialidade, integridade e disponibilidade. Sendo considerados como os pilares principais da Segurança da Informação, como é percebido na Figura 1.

Figura 1 – Pilares de Segurança da Informação

Fonte: elaborada pela autora

Dessa forma, a Segurança da informação refere-se a tomar ações para garantir a confidencialidade, integridade, disponibilidade e demais aspectos da segurança das informações, atendendo às necessidades do cliente (LYRA, 2008).

A Figura 1 apresenta os principais pilares da Segurança da Informação: confidencialidade, integridade e disponibilidade, que são fundamentais para a proteção e gestão das informações, que são definidos como:

a) confidencialidade: garantia de que o acesso à informação é restrito aos seus usuários legítimos (BEAL, 2005), ou seja, a informação estará acessível apenas para pessoas autorizadas;
b) integridade: toda informação deve ser mantida na mesma condição em que foi disponibilizada pelo seu proprietário, visando protegê-las contra alterações indevidas, intencionais ou acidentais (SÊMOLA, 2003), ou seja, assegura que os dados não sejam alterados indevidamente, mantendo sua precisão e consistência;
c) disponibilidade: garantia de que a informação e os ativos associados estejam disponíveis para os usuários legítimos de forma oportuna (BEAL, 2005), ou seja, garantia de que as informações estejam acessíveis sempre que necessárias.

Além dos princípios citados acima autores como Lyra (2008, p.4) cita alguns outros princípios:

a) autenticação: garantir que um usuário é de fato quem alega ser;
b) não repúdio: capacidade do sistema de provar que um usuário executou uma determinada ação;
c) legalidade: garantir que o sistema esteja aderente à legislação;
d) privacidade: capacidade de um sistema de manter anônimo um usuário, impossibilitando o relacionamento entre o usuário e suas ações;
e) auditoria: capacidade do sistema de auditar tudo o que foi realizado pelos usuários, detectando fraudes ou tentativas de ataque.

Por meio de várias práticas e atividades, como criar processos, definir políticas de segurança, seguir procedimentos, treinar os profissionais, e usar ferramentas para monitorar e controlar o que acontece, entre outras ações é garantido pela segurança da informação.

O objetivo da segurança é garantir a confidencialidade, integridade e disponibilidade dos elementos de valor da organização, dar proteção adequada aos ativos da informação que representassem criticidade para o negócio, preservando assim, os pilares da segurança da informação.

Os elementos da organização precisam ser protegidos contra ameaças. Essas ameaças são entendidas como a possibilidade de um evento, acidental ou proposital, causado por algum agente, que possa impactar o ambiente, o sistema ou os ativos de informação (BEAL, 2005).

As ameaças acidentais são falhas de *hardware*, desastres naturais, erros de programação; enquanto que ameaças propositais podem ser entendidas por roubos, invasões, fraudes, dentre outros. As ameaças propositais, podem ser passivas ou ativas. As ativas envolvem alteração de dados, já as passivas envolvem invasão e/ou monitoramento, mas sem alteração de informações (DIAS, 2000).

As ameaças ainda podem ser classificadas quanto a sua intencionalidade assumindo que as ameaças podem ser divididas e ameaças naturais que decorrentes de fenômenos da natureza. Involuntárias sendo ameaças inconsistentes, quase sempre causadas pelo desconhecimento. Ou ameaças voluntárias, propositais causadas por agentes humanos como *hackers*, invasores, espiões, ladrões, criadores e disseminadores de vírus de computador, incendiários (SÊMOLA, 2003).

A informação é um recurso essencial para a existência e o sucesso de uma empresa. Com a crescente necessidade de proteger seus ativos informacionais, diversas normas foram desenvolvidas para apoiar e orientar a área de Segurança da Informação. A conformidade é um aspecto essencial da Segurança da Informação. Entre as principais normas, leis e diretrizes,

destacam-se:

a) **ISO/IEC 27001**: especificações para um Sistema de Gestão de Segurança da Informação (SGSI);
b) **ISO/IEC 27002**: diretrizes para a implementação de controles de segurança;
c) **LGPD (Lei Geral de Proteção de Dados)**: regulamentação brasileira para proteção de dados pessoais;
d) **GDPR (*General Data Protection Regulation*)**: regulação europeia voltada à privacidade e proteção de dados;
e) **NIST*Cybersecurity Framework***: referência amplamente utilizada para gerenciamento de riscos cibernéticos.

As normas são fundamentais para manter a Segurança da Informação, pois fornecem diretrizes estruturadas e melhores práticas para proteger dados e mitigar riscos. A ISO/IEC 27001 e a ISO/IEC 27002 estabelecem uma abordagem sistemática para implementar e gerenciar controles de segurança, garantindo a confidencialidade, integridade e disponibilidade das informações.

A LGPD e a GDPR, por sua vez, são indispensáveis para assegurar o tratamento adequado e a proteção dos dados pessoais, promovendo a transparência e o respeito à privacidade. Já o NIST *Cybersecurity Framework* auxilia as organizações no gerenciamento de riscos cibernéticos, oferecendo uma estrutura adaptável e eficaz para prevenir, detectar e responder a incidentes de segurança. A adoção dessas normas é essencial para empresas que desejam proteger seus ativos informacionais, manter a confiança das partes interessadas e cumprir com as exigências regulatórias

Diante disso, a Segurança da Informação é um fator crítico para a continuidade e sucesso das organizações na era digital. A implementação de boas práticas, o uso de tecnologias emergentes e a conformidade com normas regulamentadoras são passos essenciais para mitigar riscos e proteger os ativos informacionais. As organizações devem permanecer vigilantes e adaptáveis diante do cenário de ameaças em constante evolução.

2.1.1 Segurança em IoT

Na era da Internet das Coisas, a expansão de dispositivos conectados trouxe avanços importantes, mas também aumentou os riscos de ataques cibernéticos. Qualquer equipamento conectado à Internet, como *smartphones* ou dispositivos domésticos inteligentes, pode ser alvo

de invasões e utilizado para fins prejudiciais.

Isso mostra que não são só computadores e celulares que podem colocar nossos dados pessoais em risco. Qualquer aparelho ou sistema que esteja *online*, como carros ou dispositivos domésticos, pode ser alvo de *hackers* e expor informações importantes. E o problema não se limita a dados financeiros; por exemplo, *hackers* podem até invadir veículos conectados e desativar funções de segurança, o que é bem preocupante (KASPERSKY, 2020).

Apesar da IoT ter ampliado a conectividade para diversos dispositivos, os desafios relacionados à segurança cibernética não são completamente inéditos. A presença de *hackers* é uma realidade desde que começamos a aproveitar as vantagens proporcionadas pela Internet (EMNIFY, 2023). A Internet das Coisas (IoT) trouxe conveniência e inovação, mas também abriu caminho para diversos riscos de segurança. Entre os principais riscos na segurança em IoT estão (EMNIFY, 2023):

a) autenticação fraca;
b) baixo poder de processamento;
c) aplicações legadas;
d) acesso compartilhado à rede;
e) padrões de segurança inconsistentes;
f) falta de criptografia;
g) atualizações de *firmware* ausentes;
h) brechas entre redes móveis e a nuvem;
i) vulnerabilidades físicas.

Começando sobre a autenticação inadequada em dispositivos IoT representa um risco significativo para a segurança, com senhas fracas frequentemente servindo como ponto de entrada para *hackers*. Muitos dispositivos IoT não possuem autenticação robusta, tornando-os vulneráveis a invasões que podem comprometer redes inteiras ou transformá-los em *botnets*. A solução inclui a implementação de autenticação multifator, uso de senhas padrão fortes e exigências de senhas seguras geradas pelos usuários (EMNIFY, 2023).

O baixo poder de processamento em dispositivos IoT é outro risco para a segurança, pois limita sua capacidade de executar recursos essenciais de segurança, como criptografia e *firewalls*, deixando-os suscetíveis a ataques. Embora isso ajude a reduzir custos e prolongar a vida útil da bateria, também pode dificultar atualizações OTA (*Over-the-Air*). Portanto, é fundamental que a rede suporte medidas de segurança integradas (EMNIFY, 2023).

Aplicativos antigos, que não foram projetados para conectividade em nuvem, apresentam vulnerabilidades significativas devido à falta de compatibilidade com padrões de segurança modernos. Atualizá-los para atender a novas exigências de segurança é um desafio complexo, mas necessário para proteger contra ataques cibernéticos contemporâneos (EMNIFY, 2023).

Dispositivos IoT frequentemente compartilham redes com outros dispositivos, criando potenciais pontos de vulnerabilidade. Ataques direcionados a um dispositivo podem comprometer toda a rede. A utilização de redes dedicadas, *firewalls* e VPNs pode ajudar a isolar falhas de segurança e limitar danos (EMNIFY, 2023).

A ausência de padrões de segurança universalmente aceitos na IoT resulta em diretrizes fragmentadas, dificultando a segurança e a interoperabilidade dos dispositivos. A padronização da segurança é essencial para proteger dispositivos e redes contra riscos associados à comunicação de máquina para máquina (EMNIFY, 2023).

Muitos dispositivos IoT não criptografam as transmissões de dados, expondo informações sensíveis a interceptações e ataques. A implementação de criptografia ponta a ponta é vital para garantir que dados transmitidos estejam protegidos contra acessos não autorizados (EMNIFY, 2023).

A incapacidade de atualizar *firmware* regularmente aumenta o risco de falhas de segurança não corrigidas. Embora as atualizações remotas sejam ideais, limitações de rede ou *hardware* frequentemente dificultam esse processo, tornando indispensável a capacidade de gerenciar essas atualizações de maneira eficiente (EMNIFY, 2023).

As interações entre dispositivos IoT e aplicativos baseados em nuvem frequentemente envolvem transmissões pela Internet pública, deixando-as suscetíveis a interceptação. A adoção de soluções de conectividade seguras pode mitigar essas vulnerabilidades e proteger as comunicações (EMNIFY, 2023).

A falta de visibilidade e controle sobre dispositivos IoT comprometidos impede uma resposta rápida a anomalias. É essencial implementar plataformas de gerenciamento robustas que permitam monitorar comportamentos suspeitos e desativar dispositivos quando necessário (EMNIFY, 2023).

O acesso físico a dispositivos IoT, especialmente aqueles em contato regular com pessoas, aumenta os riscos de manipulação ou roubo de dados. Medidas como componentes mais robustos e funcionalidades de segurança aprimoradas ajudam a mitigar esses riscos e proteger os dispositivos de acessos não autorizados (EMNIFY, 2023).

Os riscos relacionados aos dispositivos IoT são muitos e preocupam bastante. Problemas como senhas fracas, falta de proteção nos dados que circulam e até o acesso físico aos aparelhos podem facilitar ataques de *hackers*. Isso pode colocar informações importantes em perigo, interromper serviços ou até transformar os aparelhos em ferramentas para espalhar vírus e realizar ataques pela internet. Além disso, como cada fabricante tem suas próprias regras de segurança, fica difícil manter tudo protegido. Outro problema sério é que muitos desses dispositivos antigos não recebem atualizações, o que aumenta as chances de falhas de segurança (KASPERSKY, 2020).

A fim de garantir a segurança ao usar a Internet e dispositivos conectados, existem várias formas de proteção que podem ser adotadas. Uma delas é usar criptografia para proteger informações importantes. Também é essencial proteger o *gateway* da Internet e garantir que o sistema passe por uma verificação de segurança antes de iniciar. As atualizações do fornecedor, baseadas na nuvem, devem ser feitas regularmente para manter os dispositivos seguros. Outro ponto importante é usar programas que monitoram ameaças, para identificar vazamentos de dados. Além disso, uma conexão VPN pode ajudar a proteger a navegação e os dados contra possíveis ataques (KASPERSKY, 2020).

Para evitar problemas de segurança, além dos citados acima é importante tomar cuidados que deixem os dispositivos mais seguros, como usar senhas fortes, criptografar os dados e atualizar os aparelhos sempre que possível são passos importantes. Também é bom separar os dispositivos em redes diferentes e criar formas de proteger os aparelhos contra acessos físicos não autorizados. Assim, dá para usar a tecnologia de forma mais segura e evitar que ela vire um problema.

Proteger dispositivos IoT exige a adoção de práticas como autenticação multifator, criptografia de ponta a ponta, atualizações regulares de *firmware* e redes dedicadas para isolar vulnerabilidades. Exemplos como o ataque *Mirai Botnet*, que transformou milhares de dispositivos IoT em uma rede maliciosa, e a invasão de câmeras conectadas em 2020, que expôs dados de milhões de usuários, evidenciam a importância de medidas preventivas (RADWARE, 2025).

Em 2015, especialistas demonstraram como um *Jeep Grand Cherokee* poderia ser *hackeado*. Usando o sistema multimídia do carro, eles conseguiram controlar o motor, os freios, o volante e outras partes importantes do veículo, transformando-o em um "*carro de controle remoto*". Esse experimento destacou a importância de isolar dispositivos conectados em veículos, especialmente com a crescente popularidade dos carros autônomos, para garantir a segurança dos

usuários. Esses casos reforçam a necessidade de segurança robusta para evitar que dispositivos conectados sejam usados como vetores de ataques cibernéticos.

2.2 Internet das Coisas

O conceito de *Internet of Things*, ou Internet das Coisas, surgiu no ano de 1999, pelo cientista Kevin Ashton, onde ele utilizou essa expressão para descrever a rede que conecta objetos do mundo físico à internet (ASHTON, 2009). Entretanto, para Magrini (2018), existem divergências em relação ao conceito de Internet das Coisas (IoT), não havendo, portanto, um único conceito. Para o autor o termo IoT pode ser entendido como "um ambiente de objetos físicos interconectados com a internet por meio de sensores pequenos e embutidos" (MAGRINI, 2018).

Gubbi et al., descrevem a IoT como "um paradigma que transforma objetos do cotidiano em dispositivos inteligentes capazes de coletar dados e se comunicar por meio de uma infraestrutura de rede comum" (GUBBI *et al.*, 2013). Além disso, em Atzori et al., a IoT é "a visão de um mundo conectado, no qual "coisas" físicas tornam-se participantes ativas, compartilhando informações e oferecendo serviços por meio de interfaces padronizadas" (ATZORI *et al.*, 2010).

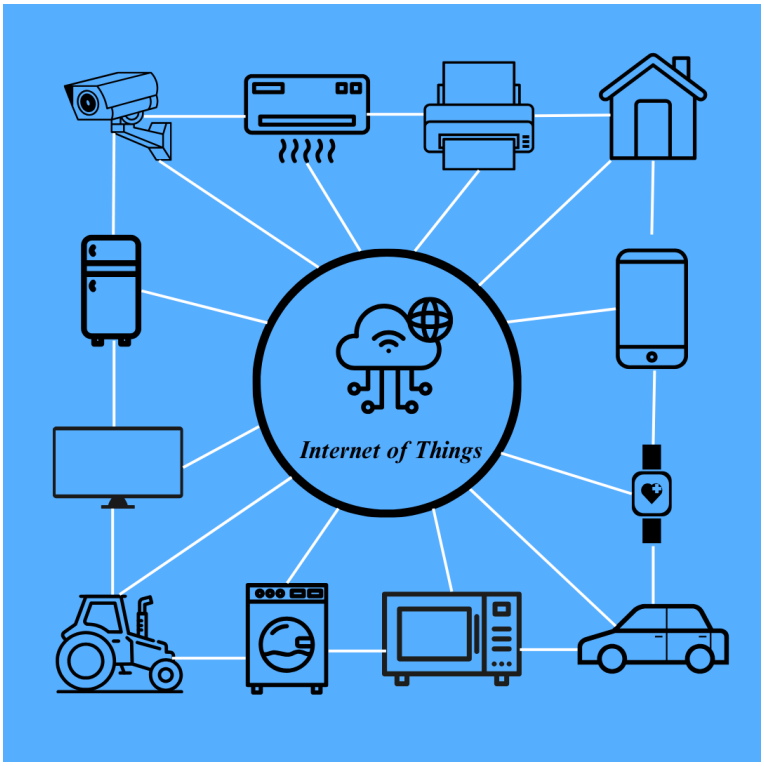
O Instituto Nacional de Padrões e Tecnologia dos EUA - NIST, define IoT como "uma coleção de tecnologias diversificadas que interagem com o mundo físico" (NIST, 2019), além disso, é considerado um "sistema dinâmico que inclui uma rede de sensores, controladores e dispositivos inteligentes interconectados para promover a troca de dados entre pessoas, sistemas e objetos no ambiente digital e físico" (NIST, 2016).

De maneira semelhante, a *International Telecommunication Union - Telecommunication Standardization Sector* (ITU-T) define a IoT como "uma rede global de infraestrutura que conecta objetos físicos e virtuais por meio de tecnologias de identificação e comunicação, com capacidades de captura e processamento de dados em tempo real" (ITU-T, 2012).

É relevante observar como cada autor ou organização foca em aspectos específicos, como interconexão, coleta de dados e interação entre ambientes físicos e digitais. Diante da diversidade de definições sobre a Internet das Coisas (IoT) apresentadas acima, torna-se evidente a complexidade e relevância desse tema para a área de Segurança da Informação. Cada perspectiva contribui com elementos que evidenciam os desafios de proteger um ambiente altamente interconectado e dinâmico.

A Figura 2 ilustra o conceito de Internet das Coisas (IoT). No centro da imagem, encontra-se um símbolo representando a conectividade central da IoT, que combina elementos como uma nuvem, sinais de rede e ícones globais. Essa representação simboliza a integração de dispositivos em uma infraestrutura compartilhada, frequentemente baseada na computação em nuvem. Essa infraestrutura possibilita a comunicação entre os dispositivos conectados e o processamento de dados de forma eficiente, destacando a interconexão inteligente que transforma os objetos cotidianos em elementos colaborativos e automatizados.

Figura 2 – Representação da Internet das Coisas



Fonte: Elaborada pela autora.

Em torno desse núcleo, Figura 2, estão representados diversos tipos de dispositivos do cotidiano, incluindo câmera de segurança, ar-condicionado, impressora, casa inteligente, smartphone, relógio inteligente, automóvel inteligente, micro-ondas, máquina de lavar, trator, smart tv e geladeira. Esses dispositivos estão conectados ao núcleo por meio de linhas que simbolizam os diferentes modelos de comunicação, sejam eles centralizados, onde os dispositivos enviam dados a um sistema principal, ou distribuídos, em que os dispositivos podem se comunicar diretamente entre si.

A Internet das Coisas (IoT) pode ser aplicada em diversos lugares, como casas, prédios, indústrias, transportes, hospitais e sistemas de gerenciamento de água, energia e ali-

mentos. Ela torna esses ambientes mais inteligentes ao utilizar a tecnologia de forma discreta para melhorar suas funções. Para criar um sistema baseado em IoT, é importante usar sensores e atuadores conectados em uma rede. Esses dispositivos coletam e enviam informações, que podem ser usadas para monitorar aspectos do ambiente, como a temperatura ou os horários em que há mais ou menos movimento no local (FRIESS, 2013).

Diante disso, a Internet das Coisas não apenas revoluciona a interação do ser humano com o ambiente ao seu redor, como também abre portas para a melhoria em áreas complexadas, como saúde, indústria, cidades e outras. Com a constante evolução das tecnologias e o aumento da conectividade, espera-se que a IoT continue a expandir suas fronteiras, trazendo inovações que irão moldar o futuro das interações cotidianas e operações empresariais.

2.3 Arquiteturas e camadas IoT

A arquitetura da Internet das Coisas (IoT) é fundamental para entender como os dispositivos conectados operam de maneira integrada, permitindo a coleta, o processamento e a troca de dados em tempo real. Ela é formada por camadas que organizam as funções e responsabilidades de cada componente no ecossistema IoT, incluindo sensores, dispositivos, plataformas de análise e aplicativos.

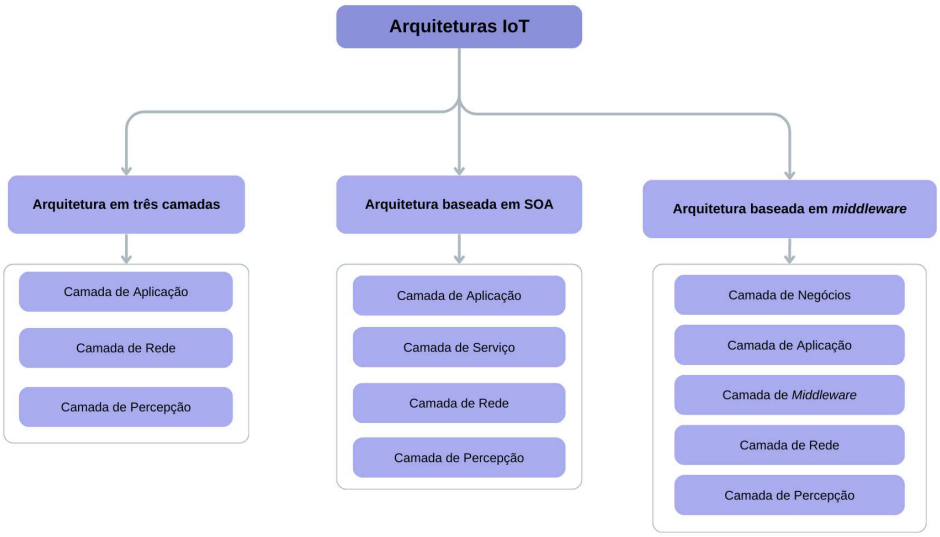
Compreender essas arquiteturas é essencial para otimizar o desempenho dos sistemas e para garantir a segurança e a escalabilidade das soluções, fatores fundamentais em um ambiente cada vez mais dependente de ligação e interoperabilidade.

A Figura 3 mostra as principais arquiteturas utilizadas em sistemas IoT. Sendo elas: arquitetura em três camadas, arquitetura baseado em SOA, arquitetura baseada em *middleware* e arquitetura em cinco camadas.

A arquitetura em três camadas, como o próprio nome sugere, é estruturada em três níveis básicos, sendo eles: camada de percepção, camada de rede e camada de aplicação. Cada camada desempenham funções específicas:

a) camada de percepção: também conhecida como a camada dos sensores, a chamada camada de percepção é como a base de tudo. Ela é responsável por interagir com o ambiente usando dispositivos inteligentes, sensores, câmeras, GPS e atuadores. Essa camada faz coisas importantes, como conectar os dispositivos ao sistema IoT, medir e processar dados em tempo real sobre o que está acontecendo, e enviar essas informações para outras partes do sistema. É basicamente a parte

Figura 3 – Arquiteturas IoT



Fonte: adaptado de (ISLAM *et al.*, 2023).

que sente e passa as informações adiante (FIELDING, 2000). O grande sucesso da IoT, conhecido como *big data*, começa nessa camada (ISLAM *et al.*, 2023).

b) camada de rede: também chamada de camada de comunicação, fica no meio da estrutura da IoT e é considerada o "coração"desse sistema (UPADHYAY; UPADHYAY, 2021). Essa camada tem o papel de criar caminhos para os dados, permitindo que eles sejam enviados em pacotes por meio da rede. Ela conecta tudo, passando mensagens entre servidores, a nuvem e os dispositivos (DIZDAREVIC *et al.*, 2019). Em resumo, a camada de rede faz a ponte entre a camada de percepção (que coleta os dados) e a camada de aplicação (que usa esses dados). Essa camada funciona com várias tecnologias, como Wi-Fi, *bluetooth* e *ZigBee*, e utiliza equipamentos como *gateways*, *hubs* e outros (DIZDAREVIC *et al.*, 2019) (AL., 2021). Ela é muito importante porque usa protocolos sem fio, o que facilita a instalação de dispositivos em lugares difíceis sem precisar de muita infraestrutura ou esforço humano. A camada de rede transmite e recebe dados de várias aplicações, usando diferentes padrões e protocolos de comunicação. Além disso, ela tem recursos que permitem integrar facilmente vários dispositivos, tornando o conceito de IoT mais acessível e conectado (MOHAMED, 2019).

c) camada de aplicação: a camada de aplicação fica no topo da rede IoT e entendido como a ligação entre os usuários e as aplicações. Ela é responsável por pegar os dados que vêm da camada de rede e usá-los em tarefas específicas. É nessa

camada que as informações coletadas ganham significado e são transformadas em ações ou resultados úteis para quem está usando o sistema (ZHANG *et al.*, 2021).

As três camadas trabalham de forma integrada para tornar o sistema IoT funcional e eficiente. A camada de percepção, responsável por coletar informações do ambiente, forma a base do ecossistema. A camada de rede conecta os dispositivos, transmite os dados coletados e garante a comunicação entre os diferentes elementos. Por fim, a camada de aplicação transforma esses dados em serviços úteis para os usuários, permitindo a implementação de soluções inteligentes em diversos contextos, como residências e veículos. Essa estrutura demonstra como cada camada desempenha um papel crucial para o sucesso da IoT, como exemplifica a Figura 4.

Figura 4 – Arquitetura de três camadas IoT

Fonte: adaptado de (KITRUM, 2023).

O paradigma SOA (Arquitetura Orientada a Serviços) funciona como uma troca entre dois tipos de *softwares*: o que fornece serviços e o que os utiliza. O *software* consumidor envia uma solicitação ao produtor, que responde com o serviço solicitado. Essa comunicação ocorre por meio de mensagens claras e compreensíveis para ambos os lados. Além disso, um *software* que produz serviços também pode consumir serviços de outro sistema, criando uma interação dinâmica (FERREIRA, 2015).

Arquitetura baseada em SOA, ou arquitetura orientada a serviços, é um modelo que conecta diferentes partes de *software* e hardware usando interfaces e protocolos padronizados. Ela adiciona uma camada de serviço entre a de rede e a de aplicação, que ajuda a organizar e ampliar os serviços. Essa camada inclui funções como gerenciar serviços, descobrir solicitações, combinar serviços e conectar tudo por meio de interfaces. No final, a camada de aplicação utiliza tudo isso para atender às necessidades do usuário de forma eficiente (ISLAM *et al.*, 2023).

O gerenciamento de serviços cuida dos processos para garantir que as solicitações sejam atendidas. A parte de descoberta de serviço encontra os pedidos feitos. A composição de serviços é responsável por comunicar e juntar os serviços para resolver os pedidos de forma rápida. Todos os serviços se conectam através da interface de serviço, e a camada de aplicação é a que realmente atende o pedido do usuário (ISLAM *et al.*, 2023).

A Arquitetura Orientada a Serviços oferece uma solução eficiente para integrar diferentes plataformas e atender às demandas do usuário de forma ágil. Essa organização estruturada e baseada em camadas é essencial para sistemas modernos, permitindo maior flexibilidade e escalabilidade

Outra arquitetura importante no contexto de IoT é a arquitetura baseada em *middleware*, conhecida também como arquitetura em cinco camadas. A arquitetura de cinco camadas, como mostra a Figura 5, surgiu para acompanhar os avanços da IoT e suas várias aplicações nos negócios. Ela adiciona duas camadas extras à arquitetura tradicional de três camadas: a camada de *middleware*, que fica entre a camada de rede e a camada de aplicação, e a camada de negócio, que está no topo da estrutura. Essas camadas extras ajudam a gerenciar melhor os dados e a transformar as informações em estratégias e decisões de negócios (GUPTA; QUAMARA, 2020).

A camada de *middleware* funciona como um intermediário que organiza e filtra os dados coletados pelos dispositivos. Ela também ajuda a explorar esses dados e controla o acesso a diferentes dispositivos para várias aplicações. Nos últimos anos, essa camada ficou mais importante porque facilita a criação de novos serviços e o uso de métodos antigos em projetos novos (PALMACCIO *et al.*, 2021).

Já a camada de negócio, organiza os serviços e atividades da IoT. Ela pega os dados da camada de aplicação e transforma em gráficos, modelos de negócio e outras formas de visualização. Além disso, essa camada é responsável por planejar, implementar e acompanhar tudo o que acontece na IoT, garantindo que os resultados das outras camadas estejam alinhados com o objetivo final e garantindo a privacidade dos usuários (BUJARI *et al.*, 2018).

Figura 5 – Arquitetura de *middleware* IoT

Fonte: adaptado de (KITRUM, 2023).

2.4 Protocolos de aplicação para IoT

Os protocolos de IoT são como um conjunto de regras que ajudam os dispositivos da Internet das Coisas a "*conversar*"entre si. Eles garantem que aparelhos de diferentes marcas ou plataformas possam se comunicar de forma fácil e sem problemas. Esses protocolos são muito importantes porque tornam possível a troca de informações entre os dispositivos, o que é essencial para que projetos de IoT funcionem bem (DIGITAL, 2025). Sem essas regras, seria difícil ou até impossível os dispositivos trabalharem juntos corretamente.

A comunicação eficiente entre dispositivos em Internet das Coisas (IoT) é essencial, especialmente em ambientes com recursos limitados e conectividade instável. Para atender

a essas necessidades, diversos protocolos de aplicação foram desenvolvidos, cada um com características específicas para cenários distintos. Entre os principais, destacam-se o MQTT, o CoAP e o AMQP, sendo estes os protocolos escolhidos para serem analisados nesta pesquisa. A seguir, esses protocolos serão analisados mais detalhadamente.

2.4.1 MQTT

Message Queue Telemetry Transport (MQTT) é um protocolo de transporte de mensagens baseado no modelo publicação/assinatura entre servidor e cliente, voltado para dispositivos restritos e redes inseguras, com baixa largura de banda e alta latência. Foi desenvolvido em 1999 por *Andy Stanford-Clark* (IBM) e *Arlen Nipper* (Arcom), para ser usado no setor de petróleo e gás para permitir perda mínima de bateria e uso mínimo de largura de banda ao conectar-se a oleodutos via satélite (HIVEMQ, 2024).

O protocolo MQTT surgiu inicialmente como *Message Queuing Telemetry Transport*, nome associado ao produto IBM MQ *Series*, que deu suporte à sua concepção e implementação inicial. Em 2010, a IBM lançou o MQTT 3.1 como um protocolo aberto e gratuito, disponível para implementação por qualquer pessoa. Posteriormente, em 2013, o protocolo foi submetido ao órgão de padronização *Organization for the Advancement of Structured Information Standards* (OASIS) para manutenção. Em 2019, o OASIS introduziu uma versão atualizada, o MQTT 5 (HIVEMQ, 2024).

O protocolo MQTT usa um formato de mensagem binária para comunicação entre o cliente e o *broker*. Esse formato usado pelo protocolo é projetado para reduzir o tamanho das mensagens e aumentar a eficiência da comunicação. Ao usar um formato binário, o protocolo pode minimizar a quantidade de dados que precisa ser transmitida e reduzir o poder de processamento necessário para interpretar mensagens, ideal para ambientes com recursos limitados (HIVEMQ, 2024).

Tabela 1 – Cabeçalho de uma mensagem MQTT

bit	7	6	5	4	3	2	1	0
***Byte* 1**	Tipo da Mensagem				Flag DUP	Nível QoS		*RETAIN*
***Byte* 2**	Largura restante							

Fonte: adaptado de (IBM, 2010).

A Tabela 1 ilustra o formato do cabeçalho do MQTT, onde cada mensagem possui um cabeçalho fixo de dois *bytes*. O primeiro *byte* contém o campo que identifica o tipo da

mensagem, além de marcadores adicionais, como DUP, nível de QoS e *RETAIN*. O tipo da mensagem ocupa os bits 7 a 4 do primeiro byte do cabeçalho fixo A lista de tipos definida na versão 3.1 do MQTT é descrita na Tabela 2. Os quatro *bits* restantes desse *byte* são divididos em quatro campos, que atuam como marcadores para indicar preferências configuradas antes do envio da mensagem (MARTINS; ZEM, 2015). Esses campos são:

a) *Duplicate Delivery (DUP)*: acrônimo relativo à entrega duplicada, este marcador ocupa o bit 4 e é ativado quando o cliente ou o servidor tentam reenviar mensagens do tipo *PUBLISH*, *PUBREL*, *SUBSCRIBE* ou *UNSUBSCRIBE* que tenham Quality of Service (QoS) maior que 0 e *requeiram acknowlegment* (ACK);
b) *Quality of Service (QoS)*: este marcador ocupa os dois penúltimos bits e indica o nível de garantia da entrega de uma mensagem *PUBLISH*. Os diferentes níveis de QoS são apresentados a seguir e nas Figuras 7, 8 e 9;
c) *RETAIN*: quando um cliente envia uma mensagem *PUBLISH* ao servidor com este marcador ativado, ela deve ser retida no servidor mesmo depois de ser entregue aos assinantes. No evento de uma nova subscrição a um tópico, a última mensagem retida para este tópico deve ser enviada para o novo assinante caso este marcador esteja ativado.

Tabela 2 – Tipos de mensagens no MQTT

Nome	**Enumeração**	**Descrição**
Reservado	0	Reservado
CONNECT	1	Requisição de conexão do cliente ao servidor
CONNACK	2	ACK de conexão
PUBLISH	3	Publicação de mensagem
PUBACK	4	ACK de publicação
PUBREC	5	Publicação recebida (garantia de entrega parte I)
PUBREL	6	Publicação liberada (garantia de entrega parte II)
PUBCOMP	7	Publicação completa (garantia de entrega parte III)
SUBSCRIBE	8	Requisição de subscrição do cliente
SUBACK	9	ACK de subscrição
UNSUBSCRIBE	10	Requisição de cancelamento de subscrição do cliente
UNSUBACK	11	ACK de cancelamento de subscrição
PINGREQ	12	Requisição PING
PINGRESP	13	Resposta PING
DISCONNECT	14	Cliente desconectando
Reservado	15	Reservado

Fonte: (IBM, 2010)

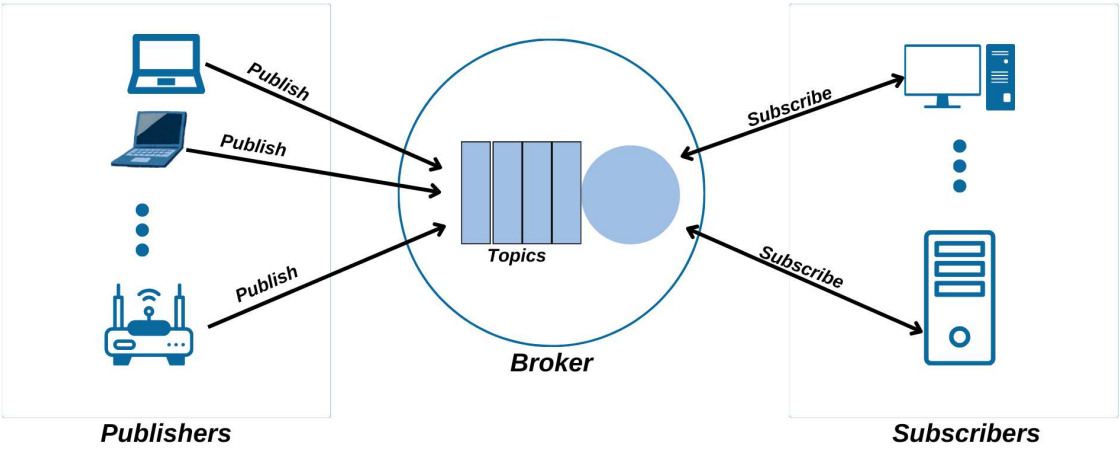
A largura restante do cabeçalho fixo (*byte* 2) é usada para representar a quantidade de bytes remanescentes na mensagem. Incluindo dados do cabeçalho variável e do *payload*. Cabeçalho variável é um componente presente em alguns tipos de mensagem MQTT e está

localizado entre o cabeçalho fixo e o *payload*. Usado principalmente nas mensagens *CONNECT*, este cabeçalho possui dois campos para nome e versão do protocolo, respectivamente e mais uma série de marcadores que definirão algumas diretivas para a conexão entre cliente e servidor (MARTINS; ZEM, 2015).

A arquitetura do protocolo MQTT é baseada em publicação/assinatura e utiliza o protocolo TCP/IP para fornecer conectividade, o MQTT usa um modelo baseado em *broker*, os clientes se conectam a esse *broker* e as mensagens são publicadas em tópicos. Os assinantes podem então assinar tópicos específicos e receber as mensagens publicadas. Os tópicos são *strings* nas quais as mensagens são publicadas e assinadas e as assinaturas são usadas para especificar sobre quais tópicos um cliente está interessado em receber mensagens (HIVEMQ, 2024).

A arquitetura do protocolo MQTT é mostrada na Figura 6, ilustrando os seus principais componentes e como eles interagem, onde:

Figura 6 – Arquitetura do protocolo MQTT



Fonte: adaptado de (AL-FUQAHA *et al.*, 2015).

a) *broker* (servidor central): o *broker* é o elemento central na arquitetura MQTT. Ele gerencia a comunicação entre publicadores e assinantes, recebendo mensagens dos publicadores e as encaminhando para os assinantes que se inscreveram nos tópicos correspondentes (HIVEMQ, 2024);
b) *publisher* (publicador): o publicador é o cliente que envia mensagens ao *broker*. Ele publica dados em um tópico específico. Exemplo disso é um sensor de temperatura envia leituras para o tópico casa/sala/temperatura (HIVEMQ, 2024);
c) *subscriber* (assinante): o assinante é o cliente que recebe mensagens de tópicos

aos quais está inscrito. Ele só receberá mensagens publicadas nos tópicos que subscreveu. Exemplo na prática: um aplicativo de monitoramento de temperatura está inscrito no tópico casa/sala/temperatura (HIVEMQ, 2024);

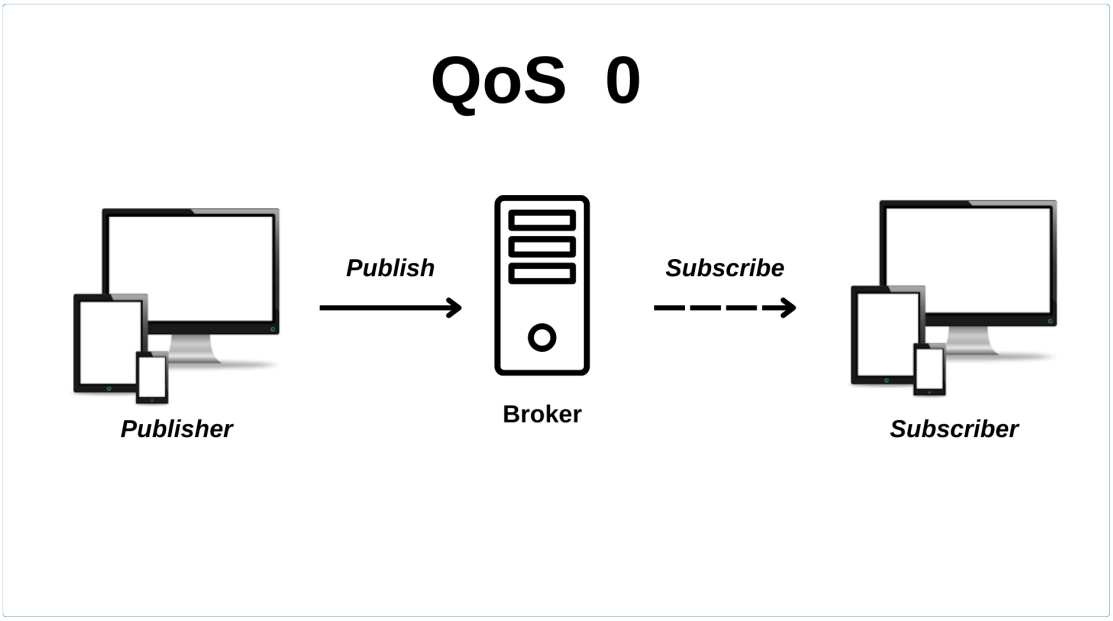
d) tópico: representado como um caminho hierárquico (ex.: casa/cozinha/umidade). Os tópicos conectam publicadores e assinantes de forma indireta, ou seja, o publicador não precisa saber quem está inscrito no tópico (HIVEMQ, 2024);

e) mensagens: são as unidades de dados que transitam na arquitetura MQTT. Uma mensagem contém: Tópico - Indica "endereço" do dado e *Payload* - O conteúdo enviado (ex.: 25°C ou 60% de umidade) (HIVEMQ, 2024).

Cabe ressaltar, que um cliente pode assinar vários tópicos de uma vez, e um tópico pode ter vários assinantes. Além de tópicos e assinaturas, o MQTT também suporta coringas, que podem ser usados para assinar vários tópicos que correspondem a um determinado padrão. Os dois tipos de curingas são o coringa de nível único (+), que corresponde a um único nível em um tópico, e o coringa multinível (#), que corresponde a todos os níveis após o nível especificado em um tópico (HIVEMQ, 2024).

Além disso, o MQTT disponibiliza três níveis de Qualidade de Serviço (QoS): QoS 0, QoS 1 e QoS 2, que se referem ao nível de garantia na entrega de mensagens entre o cliente (*plublish/subscribe*) e o *broker*. Onde:

Figura 7 – MQTT QoS 0



Fonte: adaptado de (CEDALO, 2023).

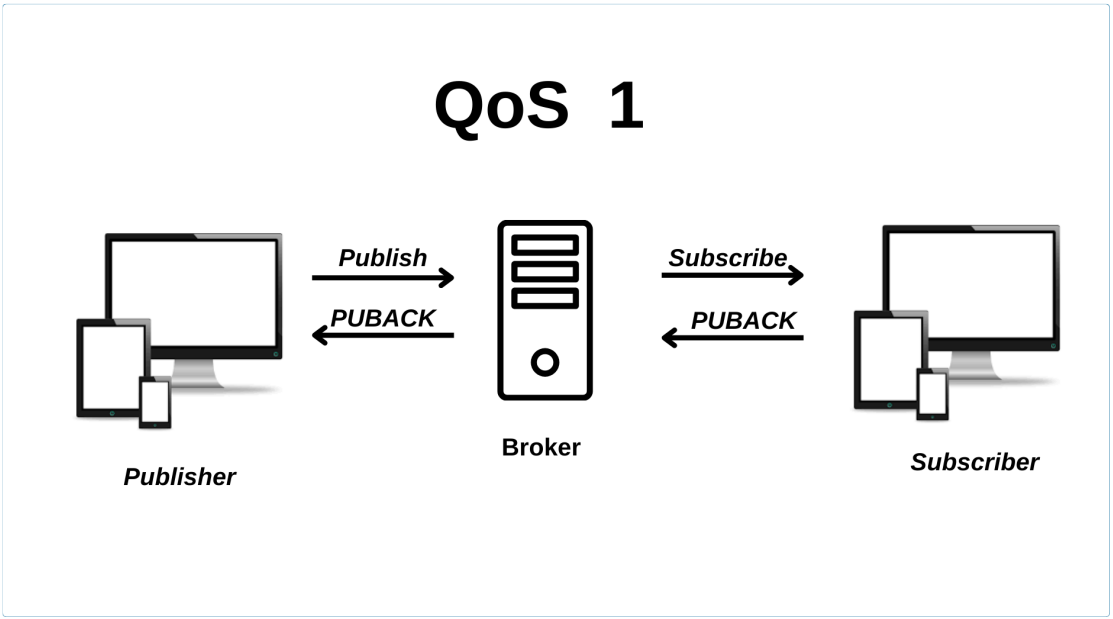
a) QoS 0: este nível fornece entrega "no máximo uma vez", onde as mensagens são

enviadas sem confirmação e podem ser perdidas. Este é o nível mais baixo de QoS e é normalmente usado em situações onde a perda de mensagens é aceitável ou onde a mensagem não é crítica (HIVEMQ, 2024). Um exemplo de MQTT QoS 0 é o sensor na entrada de um supermercado que conta o número de clientes que entram e saem da loja. Neste cenário, o publicador envia a contagem atual e não precisa de nenhuma confirmação (LIGHT, 2022). Como mostra a Figura 7 o publicador envia a mensagem ao *broker* uma vez e não espera por nenhuma confirmação;

b) QoS 1: este nível fornece entrega "pelo menos uma vez", onde as mensagens são confirmadas e reenviadas se necessário. Com QoS 1, o publicador envia a mensagem ao *broker* e aguarda a confirmação antes de prosseguir. Se o *broker* não responder dentro de um tempo definido, o publicador reenvia a mensagem, como mostra a Figura 8, que começa quando o *publish* (remetente de dados) envia uma mensagem ao *broker* (receptor de dados), que então a encaminha aos assinantes (receptor de dados). Uma vez que o *broker* recebe a mensagem, ele envia uma confirmação (pacote *PUBACK*) de volta ao remetente. Se a confirmação for perdida, o publicador (remetente de dados) retransmite a mensagem até receber a confirmação do *Broker* MQTT (receptor de dados) (CEDALO, 2023). Este nível de QoS é normalmente usado em situações onde a perda de mensagens é inaceitável, mas a duplicação de mensagens é tolerável (HIVEMQ, 2024) O MQTT QoS 1 é comumente utilizado em cenários como o de veículos automotivos. Por exemplo, quando o cinto de segurança do motorista não está afivelado, o carro envia alertas de forma repetida até que o problema seja resolvido. Nesse contexto, o publicador precisa receber a confirmação do assinante para considerar a transmissão concluída com sucesso (LIGHT, 2022);

c) QoS 2: este nível fornece entrega "exatamente uma vez", onde as mensagens são confirmadas e reenviadas até que sejam recebidas exatamente uma vez pelo assinante. QoS 2 é o nível mais alto de QoS e é normalmente usado em situações onde a perda ou duplicação de mensagens é completamente inaceitável. Com QoS 2, o publicador e o corretor se envolvem em um processo de confirmação de duas etapas, onde o corretor armazena a mensagem até que ela tenha sido recebida e reconhecida pelo assinante (HIVEMQ, 2024). A Figura 9 mostra

Figura 8 – MQTT QoS 1



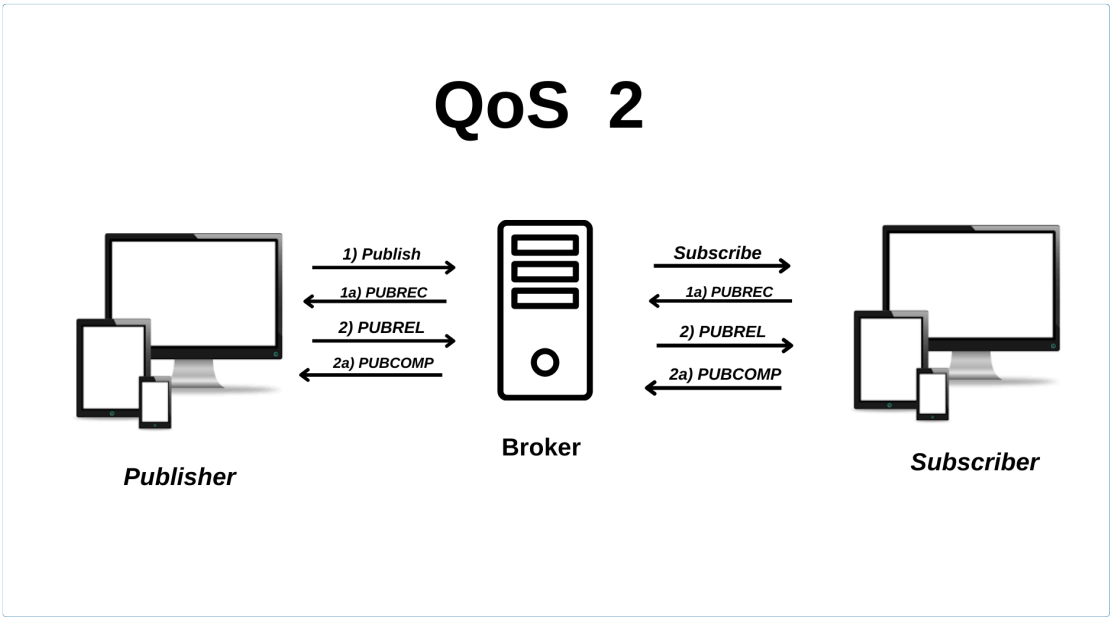
Fonte: adaptado de (CEDALO, 2023).

melhor como funciona o QoS 2, em que o remetente de dados (um publicador ou corretor MQTT) envia o pacote *PUBLISH* e aguarda uma confirmação. O receptor de dados (um corretor MQTT ou assinante) responde ao remetente com um pacote de confirmação *PUBREC* para confirmar que a mensagem foi recebida. Depois disso, o remetente envia um pacote *PUBREL* de publicação e liberação para o receptor para confirmar o recebimento do pacote *PUBREC*. Assim que o receptor recebe o pacote *PUBREL*, ele responde com outra confirmação – o pacote *PUBCOMP* de publicação completa – e então começa a processar o pacote de dados original (CEDALO, 2023). Esse nível de QoS pode ser utilizado em uma linha de produção automatizada, onde a robótica é empregada em cada estação de montagem. No entanto, o processo não pode avançar para a próxima estação até que a estação anterior seja concluída, o que torna o MQTT QoS 2 mais adequado (LIGHT, 2022).

Outro ponto importante sobre o MQTT é em relação a persistência. No MQTT, a persistência de mensagens é um recurso importante porque evita que as mensagens sejam perdidas caso a rede ou o servidor falhem. Esse sistema funciona armazenando as mensagens no servidor até que elas cheguem ao destinatário. Existem três tipos de persistência de mensagens no MQTT (HIVEMQ, 2024):

a) não persistente: esse é o padrão do MQTT. Nesse modo, as mensagens não são

Figura 9 – MQTT QoS 2



Fonte: adaptado de (CEDALO, 2023).

guardadas no servidor, então, se o servidor ou a rede falharem, elas são perdidas. Isso é ideal para casos em que as mensagens não são tão importantes e podem ser enviadas de novo sem problemas (HIVEMQ, 2024);

b) persistente em fila: as mensagens ficam armazenadas no servidor até serem entregues ao assinante. Se o assinante estiver *offline*, as mensagens ficam na fila até ele voltar a se conectar. Esse modo é bom para situações onde é importante que todas as mensagens cheguem, mesmo que o assinante não esteja conectado o tempo todo (HIVEMQ, 2024);

c) persistente com confirmação: esse é o nível mais seguro de persistência. As mensagens ficam no servidor até o assinante confirmar que recebeu. Se ele não confirmar, a mensagem é enviada novamente até que ele faça isso. Esse modo é usado quando é essencial garantir que o assinante receba e processe todas as mensagens (HIVEMQ, 2024).

Vale ressaltar que a persistência de mensagens exige mais recursos, como espaço de armazenamento e processamento. Por isso, é importante escolher o nível de persistência certo de acordo com as necessidades do sistema.

Ademais, é fundamental discutir sobre os mecanismos de segurança do protocolo MQTT, visto que a maioria deles não é configurada ou fornecida por padrão, como criptografia de dados ou autenticação de entidades (DINCULEANÃ; CHENG, 2019). Com isso, o MQTT

normalmente é utilizado em conjunto com o TLS/SSL (*Transport Layer Security/Secure Sockets Layer*) para garantir a segurança na comunicação. A autenticação e autorização, por sua vez, podem ser realizadas por meio de métodos personalizados e gerenciadas usando artifícios como nome de usuário e senha (SILVA, 2024).

2.4.2 CoAP

O *Constrained Application Protocol* - CoAP, foi desenvolvido pelo grupo CoRE (*Constrained RESTful Environments*) da *Internet Engineering Task Force* (IETF) no início de 2010, com sua especificação oficial publicada no RFC 7252 em 2014 (SHELBY *et al.*, 2014). O CoAP é um protocolo de transferência projetado para nós e infraestruturas de rede com recursos limitados, como dispositivos com pouca memória RAM e ambientes sujeitos a altas taxas de perda de pacotes (SOUZA, 2017).

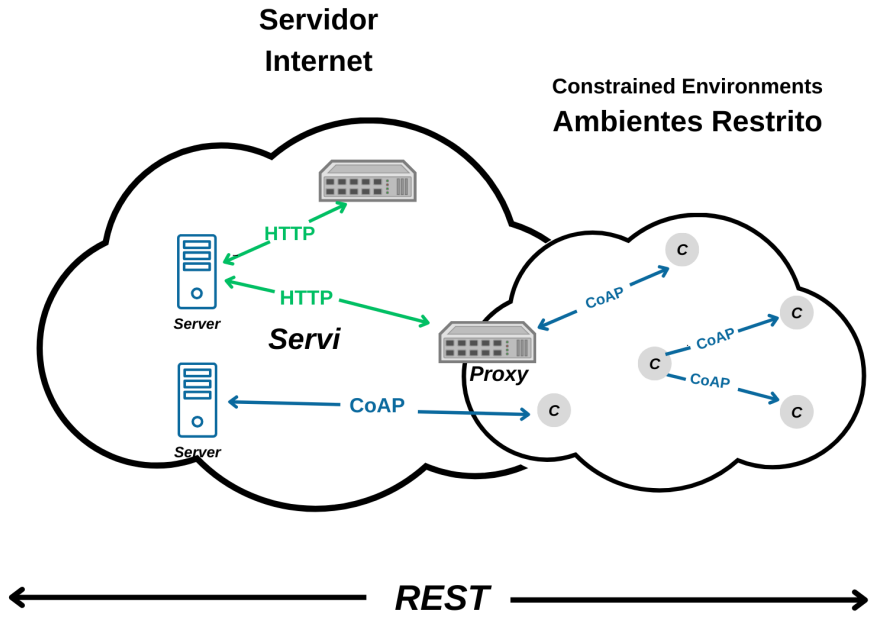
O CoAP segue o modelo *REST*, semelhante ao HTTP, mas com otimizações para redes de baixa capacidade. Ele é utilizado em aplicações como automação residencial, monitoramento ambiental e redes de sensores industriais. Além disso, o CoAP funciona do mesmo modo que o HTTP, usando um modelo cliente/servidor. Mas, quando máquinas se comunicam entre si, um mesmo sistema pode agir como cliente e servidor ao mesmo tempo. No CoAP, um cliente envia uma solicitação pedindo alguma ação em um recurso específico, que é identificado por um URI, usando um código de método. O servidor recebe esse pedido e responde com um código de resposta, podendo incluir informações sobre o recurso solicitado (SHELBY *et al.*, 2014).

Embora compartilhe características com o HTTP, como o uso do modelo cliente-servidor e os métodos *GET, POST, PUT* e *DELETE*, o CoAP se diferencia por operar sobre UDP em vez de TCP, tornando a comunicação mais leve e eficiente para dispositivos IoT. Além disso, ele utiliza mensagens compactas para reduzir o consumo de largura de banda e suporta comunicação *multicast*, um recurso inexistente no HTTP tradicional.

Diante disso, a Figura 10 ilustra a arquitetura do CoAP, seus componentes são (SHELBY *et al.*, 2014):

a) cliente CoAP (C): dispositivos que enviam requisições CoAP para os servidores CoAP. São normalmente sensores ou atuadores;
b) servidor CoAP: dispositivo ou aplicação que recebe e processa requisições CoAP, retornando respostas aos clientes CoAP;
c) *proxy*: componente intermediário que traduz mensagens entre CoAP e HTTP,

Figura 10 – Arquitetura do CoAP



Fonte: adaptado de (SHELBY, 2014).

permitindo a interoperabilidade entre redes CoAP e serviços baseados em HTTP;

d) servidor HTTP: sistema que processa requisições HTTP, geralmente na Internet, possibilitando a comunicação com sistemas baseados na web;

e) ambientes restritos (*Constrained Environments*): representam redes e dispositivos com limitações de energia, processamento e conectividade, comuns em IoT;

f) rede de comunicação: representada pelas setas no diagrama, indicando a troca de mensagens via CoAP e HTTP entre os componentes;

g) modelo *REST*: indicado na parte inferior da imagem, reforçando que o CoAP segue a arquitetura *RESTful*, semelhante ao HTTP.

A Figura 10 mostra a arquitetura do CoAP de uma forma geral. Um dos objetivos do CoRE é adaptar a arquitetura *Representational State Transfer* (REST), que foi proposta por *Fielding* em 2000, para ambientes com recursos limitados, como nós e redes. O CoAP foi criado seguindo essa arquitetura, então ele pode ser visto como um protocolo *RESTful* (SHELBY, 2014).

Além disso, o CoAP define quatro tipos de mensagens: *Confirmable, Non-confirmable, Acknowledgement, Reset* (SHELBY *et al.*, 2014).

a) *confirmable* (CON): são mensagens que requerem confirmação no destino. Assim, na ausência de perda de pacotes, cada mensagem desse tipo gera uma resposta do tipo *acknowledgment* ou *reset*;

b) *non-confirmable* (NON): não requerem confirmação de recebimento, o que é útil em aplicações que recebem leituras constantes de um sensor e a perda de algumas mensagens não não causa impacto significativo;

c) *Acknowledgement*: são mensagens que confirmam o recebimento de uma mensagem *Confirmable*. No entanto, é importante destacar que, por si só, uma mensagem ACK não indica o sucesso ou falha da requisição encapsulada na mensagem *Confirmable*;

d) *reset*: indica que uma mensagem específica (CON ou NON) foi recebida, mas falta algum contexto para seu processamento correto. Essa situação geralmente ocorre quando o receptor é reinicializado e perde o estado necessário para interpretar a mensagem.

Além disso, as mensagens do CoAP são codificadas em um formato binário com um cabeçalho de 4 bytes de tamanho fixo, seguido por um valor *Token* de comprimento variável, que pode ter entre 0 e 8 bytes de comprimento. Após o *token* pode não haver nenhuma ou uma série de opções do CoAP no formato *Type-Length-Value* (TLV), que são opcionalmente seguidas por um *payload*, ocupando o resto do datagrama. A Tabela 3 mostra o formato da mensagem CoAP (SHELBY *et al.*, 2014).

Tabela 3 – Formato da mensagem CoAP

Ver	**T**	**TKL**	**Código**	**ID da mensagem**
Token (se houver, bytes TKL)				
Opções (se houver)				
1 1 1 1 1 1 1 1	*Carga útil (se houver)*			

Fonte: Adaptado de (SHELBY *et al.*, 2014)

Os campos do cabeçalho são definidos da seguinte forma (SHELBY *et al.*, 2014):

a) versão (ver): inteiro sem sinal de dois bits que indica o número correspondente à versão do CoAP;

b) tipo (T): inteiro sem sinal de dois bits que indica se o tipo da mensagem, que pode variar entre *Confirmable* (0), *Non-confirmable* (1), *Acknowlegment* (2) ou *Reset* (3);

c) largura do *token* (TKL): inteiro sem sinal de quatro bits que especifica o tamanho do *token*, variando de 0 a 8 bytes;

d) código: inteiro sem sinal de 8 bits que indica o tipo da mensagem. Pode representar um método de requisição (conforme tabela 4), ou um código de resposta

(conforme tabela 5);

e) ID da mensagem: inteiro sem sinal de 16 bits usado para identificar duplicações de mensagens bem como para a comparação de mensagens do tipo ACK, por exemplo;

f) O *token* serve para associar requisições às suas respectivas respostas. Ele é gerado pelo cliente, enviado junto com a requisição e deve ser reproduzido pelo servidor na resposta correspondente.

Tabela 4 – Códigos dos métodos CoAP

Código	Nome
00.1	*GET*
00.2	*POST*
00.3	*PUT*
00.4	*DELETE*

Fonte: Adaptado de (SHELBY *et al.*, 2014).

Tabela 5 – Códigos de resposta do CoAP

Código	**Descrição**
2.01	*Created*
2.02	*Deleted*
2.03	*Valid*
2.04	*Changed*
2.05	*Content*
4.00	*Bad Request*
4.01	*Unauthorized*
4.02	*Bad Option*
4.03	*Forbidden*
4.04	*Not Found*
4.05	*Method Not Allowed*
4.06	*Not Acceptable*
4.12	*Precondition Failed*
4.13	*Request Entity Too Large*
4.15	*Unsupported Content-Format*
5.00	*Internal Server Error*
5.01	*Not Implemented*
5.02	*Bad Gateway*
5.03	*Service Unavailable*
5.04	*Gateway Timeout*
5.05	*Proxying Not Supported*

Fonte: Adaptado de (SHELBY *et al.*, 2014).

Ademais, após o cabeçalho, o token e as opções, se houver, vem o *payload*, que é um campo opcional. Quando a mensagem possui *payload* ele deve ser indicado por um marcador de um byte (*Payload Marker*) que indica o fim das opções e o início da carga. A ausência do marcador significa um *payload* de largura 0 bytes, enquanto a presença de um marcador

seguido por um *payload* de 0 bytes deve ser processado como um erro no formato da mensagem (SHELBY *et al.*, 2014).

Diante do exposto, é importante destacar que o CoAP não implementa autenticação ou autorização por padrão, com isso, para estabelecer segurança o protocolo usa o *Datagram Transport Layer Security* (DTLS), que adiciona criptografia e autenticação, garantindo a integridade e a confidencialidade das comunicações (SHELBY *et al.*, 2014).

2.4.3 AMQP

Advanced Message Queuing Protocol (AMQP) é um protocolo de mensagens que permite uma comunicação confiável e assíncrona entre sistemas diferentes. Diferente do MQTT, o AMQP é mais "pesado", pois tem mais sobrecarga no cabeçalho e é mais complexo. Ele oferece recursos avançados, como filas de mensagens, roteamento das mensagens com base em critérios específicos e confirmações de entrega. O *Advanced Message Queuing Protocol*/Protocolo Avançado de Enfileiramento de Mensagens (AMQP) é muito usado em situações onde é necessário garantir a confiabilidade e também onde é importante ter boa escalabilidade (ROISENBERG, 2024).

O AMQP por ser um protocolo binário e assíncrono, tem suporte a múltiplos canais. Ele é dividido em três camadas: camada de modelo, camada de sessão e camada de transporte. A camada de modelo é onde estão definidos os comandos e objetos que as aplicações podem usar. Nessa camada, alguns requisitos são importantes, tais como: garantir que as diferentes implementações possam funcionar juntas, controlar a qualidade do serviço, fazer com que os comandos do protocolo se conectem facilmente com as bibliotecas das aplicações e deixar claro como isso acontece, para que cada comando execute uma única tarefa. Os principais componentes dessa camada são (CARMO, 2012):

a) servidor: também conhecido como *broker*, é o processo que aceita conexões de clientes e faz as funções de filas de mensagens e roteamento;
b) filas (*queue*): são áreas dentro do servidor onde as mensagens ficam armazenadas, tanto na memória quanto no disco, até serem enviadas para as aplicações que as consomem. As filas funcionam de forma independente e podem ser públicas ou privadas, duráveis ou temporárias, e até ter uma existência temporária dependendo da aplicação. Existem diferentes tipos de fila, como a "armazena-e-encaminha", que distribui as mensagens para os consumidores em forma de

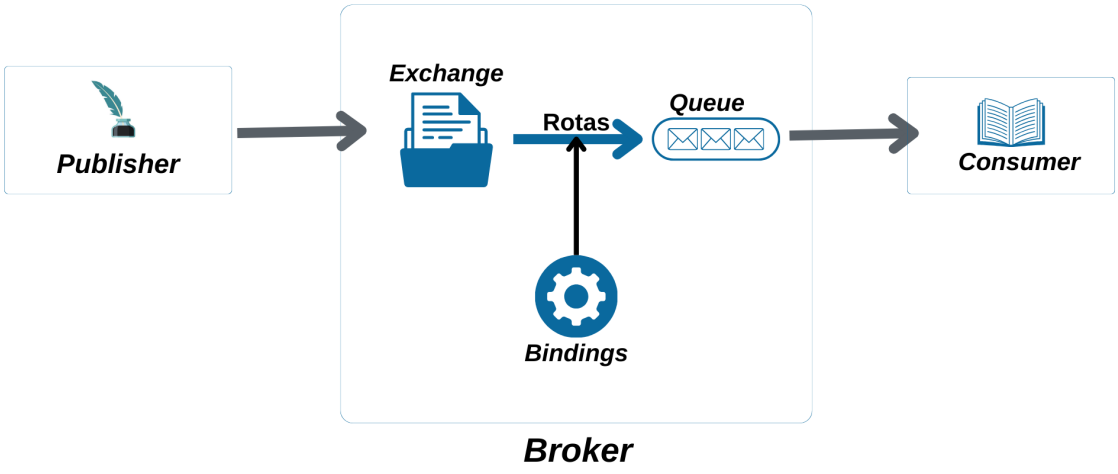
round-robin, ou uma fila que armazena as mensagens para um único consumidor;

c) *exchanges*: são processos dentro do servidor que recebem as mensagens das aplicações e as roteiam para as filas, com base em critérios específicos. Elas analisam as mensagens, usando informações no cabeçalho para decidir para qual fila a mensagem deve ser enviada. No caso deste trabalho, estamos usando *exchanges* diretas, onde a chave de roteamento precisa ser exatamente igual a uma entrada na tabela de *bindings* para que a mensagem seja enviada para a fila correta;

d) *bindings*: são as relações entre *exchanges* e filas. Elas definem como as mensagens serão roteadas dentro do sistema;

e) virtual *hosts*: são agrupamentos de *exchanges*, filas e objetos relacionados. Cada virtual host tem um ambiente de autenticação e segurança próprio, permitindo que as aplicações escolham qual virtual host usar após se autenticar no servidor.

Figura 11 – Arquitetura do AMQP: fluxo de mensagens



Fonte: adaptado de (SERRANO, 2018).

A Figura 11 representa a arquitetura da mensagem do AMQP, na qual um *Publisher* (publicador) envia mensagens para um *Exchange*, que é responsável por encaminhá-las para as filas corretas (*Queues*) de acordo com regras de roteamento (rotas). A ligação entre o *Exchange* e as filas é definida pelos *Bindings*, permitindo um direcionamento eficiente das mensagens. As filas armazenam as mensagens até que um *Consumer* (consumidor) esteja disponível para processá-las.

A segunda camada conhecida como camada de sessão atua como um intermediário entre as camadas de modelo e transporte, garantindo que os comandos sejam enviados corretamente para o *broker*. Ela tem a responsabilidade de garantir que as interações entre o cliente e o *broker* sejam confiáveis (CARMO, 2012).

Sessões são interações com nome entre um cliente e um servidor AMQP, também chamados de "pares". Todos os comandos, como envio de mensagens e criação de filas ou *exchanges*, precisam acontecer dentro de uma sessão. O ciclo de vida de alguns objetos da camada de modelo, como filas, *exchanges* e *bindings*, pode ser limitado à duração de uma sessão (CARMO, 2012).

A camada de sessão oferece alguns serviços importantes para a camada de modelo, tais como (CARMO, 2012):

a) identificação sequencial dos comandos: cada comando enviado pelos pares é identificado de forma única dentro da sessão, o que ajuda a garantir que o comando será executado exatamente uma vez. É utilizado um esquema de numeração sequencial para identificar os comandos. Esse identificador também ajuda a correlacionar o comando com seu resultado, especialmente quando ele é devolvido de forma assíncrona;
b) confirmação de que os comandos serão executados: esse serviço garante que o par solicitante possa descartar, com segurança, o estado relacionado a um comando, sabendo que ele será executado. A camada de sessão gerencia o envio e recebimento das confirmações, permitindo que o estado seja mantido durante a sessão. Isso é útil para o sistema se recuperar caso haja falhas temporárias em algum par. As confirmações podem ser enviadas em lotes ou até adiadas se o par solicitante não precisar de uma confirmação imediata;
c) notificação de comandos completados: diferente das confirmações, essa notificação informa o par solicitante de que o comando foi totalmente executado. Isso ajuda a manter a sincronização e a ordem de execução entre diferentes sessões. Se o par solicitante não precisar de uma confirmação imediata, as confirmações podem ser acumuladas e enviadas em lotes, o que diminui o tráfego na rede;
d) reenvio e recuperação em caso de falhas na rede: quando há falhas na rede, a sessão deve ser capaz de reenviar comandos cujos recebimentos pelo outro par não foram confirmados. A camada de sessão fornece os mecanismos necessários

para identificar esses comandos duvidosos e reenviá-los sem causar duplicidade.

Esses serviços ajudam a garantir que as comunicações dentro da camada de modelo sejam feitas de forma confiável, mesmo quando há falhas na rede ou no processo.

Na terceira camada do protocolo tem a camada de transporte, essa camada tem várias responsabilidades, como organizar os canais, detectar falhas, representar os dados e fazer o processo de "embrulhamento" e "desembrulhamento" das mensagens. Entre os requisitos dessa camada, alguns dos mais importantes são: ter uma forma de representar os dados de maneira compacta e rápida para ser embalada e desembrulhada, conseguir lidar com mensagens de qualquer tamanho sem grandes limitações, garantir que as sessões não sejam perdidas mesmo com falhas na rede ou no sistema, e ser capaz de trabalhar de maneira assíncrona e independente das linguagens de programação usadas (CARMO, 2012).

2.5 Vulnerabilidades e ataques comuns em protocolos de aplicação IoT

Entre os potenciais ataques que ameaçam a segurança de dispositivos foram mapeados, dentre eles: ataques de interceptação de dados (*eavesdropping*), falsificação de endereço IP (IP *spoofing*), negação de serviço (DoS) e *Distributed Denial of Service*/Negação de serviço distribuída (DDoS), ataques do homem no meio (*man-in-the-middle*) e envenenamento de serviços (*poisoning*) (SILVA, 2024).

O *spoofing* de IP é uma prática que envolve a alteração de pacotes de dados na rede, com o objetivo de enganar um sistema ou rede. Essa técnica consiste em falsificar o endereço IP de origem, ou seja, ocultar o endereço verdadeiro de um pacote, fazendo com que ele pareça ter sido enviado de um local ou origem diferente da real (RIOLINO, 2023).

O ataque *Denial of Service* (DoS) tem como objetivo impedir o funcionamento de um serviço ao sobrecarregá-lo com um grande volume de requisições desnecessárias. Isso consome os recursos do sistema, dificultando ou impossibilitando o atendimento de solicitações legítimas. Para mitigar esse tipo de ataque, servidores geralmente identificam padrões suspeitos de tráfego e bloqueiam o acesso vindo da fonte responsável pela sobrecarga (BARROS, 2021).

Os ataques *Distributed Denial of Service* (DDoS) seguem a mesma lógica do DoS, mas utilizam múltiplos dispositivos para sobrecarregar um sistema com requisições simultâneas. Isso torna a defesa mais difícil, pois bloquear apenas uma origem não é suficiente para interromper o ataque. Geralmente, essas ações são motivadas por vingança ou ativismo, já que, diferentemente de outros tipos de ataques cibernéticos, não há um ganho financeiro direto envolvido (BARROS,

2021).

Nos ataques do tipo *Man in The Middle*/Homem do meio (MITM), um invasor intercepta a comunicação entre duas partes, conseguindo acessar e monitorar os dados transmitidos. Existem várias formas de executar esse ataque, mas todas podem ser evitadas se houver um mecanismo de autenticação que garanta a identificação exclusiva do remetente das mensagens. Além disso, uma maneira eficaz de proteção é utilizar criptografia para codificar as informações trocadas, impedindo que sejam lidas por terceiros não autorizados (BARROS, 2021).

3 METODOLOGIA

A metodologia deste trabalho é composta por uma revisão bibliográfica da literatura sobre os protocolos da camada de aplicação da pilha TCP/IP em sistemas IoT. O objetivo é identificar entre os principais protocolos da camada de aplicação usados por dispositivos IoT - MQTT, CoAP e AMQP - suas vulnerabilidade e ataques para dispositivos IoT.

Estudos anteriores que abordam protocolos da camada de aplicação em IoT, discutindo suas características, vantagens e limitações. O artigo de Santo *et al* (2018), revisa os principais problemas de segurança em protocolos de comunicação para IoT (*Internet of Things*/Internet das Coisas (IoT)), categorizando-os nas camadas do modelo *Transmission Control Protocol/Internet Protocol* / Protocolo de Controle de Transmissão/Protocolo da Internet (TCP/IP): física, rede, transporte e aplicação. Para cada camada, o estudo identifica ameaças como negação de serviço, interceptação de dados e ataques de autenticação. Protocolos como *Constrained Application Protocol* (CoAP) e *Message Queuing Telemetry Transporte*/Protocolo de Mensagens de Telemetria de Fila (MQTT) são analisados quanto a vulnerabilidades e medidas de segurança, destacando-se o uso do *Datagram Transport Layer Security* (DTLS) para segurança no transporte. O artigo conclui que, para ambientes de IoT, as vulnerabilidades mais comuns são ataques de *Denial of Service*/Ataque de Negação de Serviço (DoS) e de exposição de dados, e sugere aprimoramentos futuros na segurança desses protocolos (SANTO *et al.*, 2018).

No trabalho de Rocha Junior (2022), aborda vulnerabilidades em sistemas IoT e propõe medidas de prevenção, classificando os sistemas IoT em três camadas: percepção, transporte e aplicação. As vulnerabilidades incluem ataques físicos, de autenticação, negação de serviço e invasões *man-in-the-middle*, especialmente no protocolo MQTT. Como formas de prevenção, são sugeridos métodos de camuflagem de hardware, criptografia de dados, uso de *fog computing* para reduzir armazenamento na nuvem e técnicas de aprendizado de máquina para detectar ataques de negação de serviço. O artigo também discute a necessidade de protocolos de segurança padronizados e conscientização dos usuários sobre práticas seguras (JUNIOR, 2022).

Na revisão bibliográfica contém conceitos sobre segurança da informação, Internet das Coisas, arquitetura e camadas da IoT e os protocolos da camada de aplicação que serão abordados nesse trabalho, como MQTT, CoAP e AMQP. Essa revisão busca identificar as principais teorias, estudos e práticas relevantes que fundamentam a pesquisa, permitindo uma base teórica para compreender as vulnerabilidades desses protocolos.

A revisão bibliográfica da literatura sobre os protocolos da camada de aplicação

em dispositivos IoT será elaborada de maneira organizada e planejada para a realização da revisão. Isso ajudará a manter a qualidade e a coerência do trabalho. Essa etapa é importante para assegurar a qualidade e a fundamentação teórica do trabalho. A revisão será estruturada em quarto etapas, sendo elas:

a) primeira etapa: coleta de dados, os dados foram coletados das bibliotecas digitais;
b) segunda etapa: leitura de título, resumo e palavras chaves com avaliação baseada nos critérios definidos;
c) terceira etapa: leitura completa de artigos aprovados na fase anterior e reaplicação dos critérios de seleção para responder as questões da pesquisa;
d) quarta etapa: definição das questões da pesquisa.

Cada etapa no desenvolvimento do trabalho é importante para garantir que os objetivos sejam alcançados, com isso eles serão explicados mais detalhadamente seus processos a seguir.

3.1 Primeira etapa: coleta de dados

A coleta de dados em plataformas será realizada por meio da extração e análise de informações disponibilizadas pelos sistemas, considerando alguns critérios. Para conseguir acessar os trabalhos, foram usados mecanismos de busca que retornam estudos de várias bases de dados, com ênfase em Ciência e Tecnologia. Por isso, as seguintes bases de dados para realizar as consultas: *IEEE Xplore*, ACM *Digital Library*, S*copus* e *Web of Science*.

O *IEEE Xplore* é uma plataforma digital importante para encontrar e acessar artigos e conteúdos técnicos que foram publicados pelo IEEE (Instituto de Engenheiros Elétricos e Eletrônicos) e outros parceiros. Ele tem mais de 6 milhões de documentos e outros tipos de materiais de algumas das publicações mais conhecidas do mundo nas áreas de engenharia elétrica, ciência da computação e ciências parecidas (XPLORE, 2025).

A ACM (*Association for Computing Machinery*) é uma das maiores organizações voltadas para a computação no mundo. Ela reúne professores, pesquisadores e profissionais da área para trocar ideias, compartilhar informações e discutir desafios da tecnologia Ela publica pesquisas e artigos de especialistas em computação e tecnologia da informação, ajudando profissionais a lidar com problemas e tomar decisões importantes. Além disso, a organização também divulga revistas, boletins informativos e registros de conferências. Outro destaque da ACM é a criação de recomendações para currículos em universidades e escolas, ajudando na

formação de alunos que querem seguir carreira na área de tecnologia (MACHINERY, 2025).

Já o *Scopus* é um banco de dados que reúne resumos e citações de pesquisas de diversas áreas. Ele é organizado por especialistas e permite que pesquisadores, bibliotecários e outras pessoas encontrem informações importantes de forma rápida. O *Scopus* contém conteúdos de mais de 7.000 editoras, com mais de 91 milhões de registros, 94.000 perfis institucionais e 17 milhões de autores. Suas ferramentas avançadas de busca ajudam a encontrar pesquisas recentes, acompanhar tendências e identificar especialistas em diferentes assuntos. Além disso, ele oferece recursos que permitem visualizar, comparar e exportar dados de forma simples (BLOG, 2025).

Por fim, o *Web of Science* que auxilia na revisão dando acesso a pesquisas conhecidas e respeitadas pela comunidade científica. Como as fontes são de qualidade, os resultados da revisão sistemática acabam sendo mais confiáveis e precisos (CLARIVATE, 2025).

Essas bases de dados foram escolhidas para garantir que a pesquisa seja o mais completa, atualizada e confiável possível. Com a combinação dessas fontes, é possível ter uma visão geral do que já foi estudado na área, tornando a revisão mais precisa e relevante.

Com a finalidade de limitar as buscas às informações mais relevantes para a pesquisa, foi definida uma lista de palavras-chave e seus respectivos sinônimos, conforme demonstrado na Tabela 6. A partir desses termos, foi possível criar uma *string* de busca, apresentada na Figura 12, que foi aplicada nos campos de título, resumo e palavras-chave do autor. Para garantir a consistência da pesquisa, a mesma *string* de busca foi utilizada nos mesmos campos em todas as bases de dados selecionadas. O período de 2020 a 2024 foi escolhido devido à sua relação com a atualidade do tema proposto.

Tabela 6 – Palavras-chave e sinônimos

Palavra-chave	**Sinônimos**
Internet of Things	*IoT, Industry 4.0, Industrial Internet of Things, IIoT, Industrial IoT*
Protocol	*Protocols*
Vulnerabilities and attacks	*Vulnerabilities and attack, Vulnerability and attack, Vulnerability and attacks*
MQTT, CoAP, AMQP	*Message Queuing Telemetry Transport*, MQTT *protocol*, MQTT *protocols, Constrained Application Protocol, CoAP protocol, CoAP protocols*, *Advanced Message Queuing Protocol*, AMQP *protocol*, AMQP *protocols*
Security	*Protection, safety*

Fonte: Elaborada pela autora.

A *string* de busca da Figura 12 emprega os operadores lógicos *AND* e *OR* que busca

Figura 12 – *String* de busca

> *(“Internet of Things” **OR** “IoT” **OR** “Industry 4.0” **OR** “Industrial Internet of Things” **OR** “IIoT” **OR** “Industrial IoT”) **AND** (“Protocols” **OR** “Protocol”) **AND** (“Vulnerabilities and attack” **OR** “Vulnerability and attack” **OR** “Vulnerability and attack” **OR** “Vulnerabilities and attacks”) **AND** (“Constrained Application Protocol” **OR** “Advanced Message Queuing Protocol” **OR** “Message Queuing Telemetry Transport” **OR** “MQTT protocol, CoAP, AMQT” **OR** “MQTT, CoAP, AMQT protocols”) **AND** (“protection” **OR** “safety” **OR** “Security”)*

Fonte: elaborada pela autora

um equilíbrio entre abrangência e relevância da pesquisa. O operador *AND* serve para que estejam presentes todos os aspectos relevantes da pesquisa (IoT, protocolos, vulnerabilidades, segurança) nos artigos selecionados, enquanto o operador *OR* é utilizado para que se mantenham alternativas quanto a termos sinônimos. Este procedimento aumenta a possibilidade de localizar artigos que correspondam aos diferentes aspectos do problema de pesquisa e ao mesmo tempo mantém o foco no tema principal.

Em suma, esta seleção de caracteres é uma técnica que é capaz de não somente encontrar uma abrangente faixa de artigos relacionados com o estudo dos protocolos de IoT, suas vulnerabilidades e seus mecanismos de segurança, mas também providenciar resultados que sejam especificamente relevantes para a questão concreta de pesquisa.

3.2 Segunda etapa: seleção e critérios

Para realizar a seleção dos artigos é necessário estabelecer alguns critérios para a filtragem dos trabalhos, com isso essa etapa vai reavaliar os trabalhos que as bases de dado retornou como resultados, e a partir disso será realizado a leitura de título, resumo e palavras chaves.

Nessa etapa de seleção os trabalhos selecionados após as buscas devem atender aos critérios de inclusão (CI) ou critérios de exclusão (CE), os quais serão considerados CI:

a) **CI1:** Trabalhos que apresentem vulnerabilidades e ataques nos protocolos na camada de aplicação, nos protocolos: MQTT, CoAP e AMQP.

b) **CI2:** Trabalhos que apresentam mecanismos de segurança em protocolos das camadas de aplicação em dispositivos IoT.

Em contrapartida serão considerado critérios de exclusão (CE) os:

a) **CE1:** Trabalhos referentes a protocolos que não seja da camada de aplicação.

b) **CE2:** Trabalhos que abordam os protocolos estudados, mas não são relacionados a pesquisa proposta.
c) **CE3:** Trabalhos que falem de IoT, mas que o foco não são os protocolos e nem as vulnerabilidades.
d) **CE4:** Trabalhos que falem de segurança, mas que não sejam voltados para IoT ou os protocolos.
e) **CE5:** Trabalhos em desacordo com o tema proposto.
f) **CE6:** Trabalhos que não tem acesso livre para leitura.

Após a aplicação dos critérios de inclusão (CI) e de exclusão (CE), os trabalhos selecionados serviram de base para a pesquisa sobre falhas, vulnerabilidades e segurança dos protocolos utilizados em sistemas IoT. Com essa seleção, será viável analisar de forma mais profunda e objetiva os protocolos MQTT, CoAP e AMQP, além das medidas de segurança utilizadas para proteção desses sistemas. Portanto, os artigos selecionados serão pertinentes e auxiliarão na resposta às questões da pesquisa de forma mais objetiva.

3.3 Terceira etapa: resumo dos trabalhos selecionados

Nesta etapa, os artigos que atendem aos critérios definidos na etapa anterior serão analisados e resumidos. A análise será relacionadas nas características de segurança e privacidade dos protocolos MQTT, CoAP e AMQP em sistemas IoT. Com isso, serão extraídas as informações essenciais para responder às perguntas da pesquisa.

É importante ressaltar que para garantir a qualidade da pesquisa somente serão levados em consideração, na análise, os estudos que atenderem aos critérios de inclusão. Assim, assegura que a seleção dos estudos esteja focada e adequada aos objetivos do trabalho, eliminando aqueles que embora possuam temas próximos, ou palavras-chaves de acordo com a *string* de busca estabelecida, não geram diretamente dados sobre vulnerabilidades e segurança dos protocolos da camada de aplicação de sistemas IoT.

3.4 Quarta etapa: definição das perguntas da pesquisa

Nessa etapa, serão definidas as perguntas que orientam a pesquisa, permitindo a identificação dos principais pontos a serem analisados, para organização e análise das informações. As perguntas centrais da pesquisa são:

a) **Questão 1:** Quais dos três protocolos da camada de aplicação (MQTT, CoAP, AMQP) são mais utilizados pela indústria e pela academia em sistemas IoT?

b) **Questão 2:** Quais os níveis de segurança e privacidade oferecidos por esses protocolos?

c) **Questão 3:** Quais as falhas de segurança conhecidas nesses protocolos e como elas podem ser exploradas?

d) **Questão 4**: Existem possíveis medidas para resolver essas falhas de segurança?

Essa metodologia tem como objetivo garantir uma abordagem objetiva e bem estruturada para entender como os protocolos MQTT, CoAP e AMQP se comportam no contexto de segurança e privacidade em sistemas IoT, analisando tanto suas aplicações quanto as vulnerabilidades conhecidas.

4 RESULTADOS

Nesta seção, serão apresentados os resultados decorrentes da execução das etapas desenvolvidas na metodologia. Cada etapa vai ser explicada com base nos dados analisados, dos critérios estabelecidos e pontos observados, mostrando como os objetivos definidos se ligam aos resultados obtidos.

4.1 Resultados da primeira etapa

Ao analisar as bases de bados: *IEEE Xplore, ACM, Scopus e Web of Science*, entre o período de 2020 a 2024 usando a mesma *string* de busca para a pesquisa, Figura 12, tem o resultado visualizado na Tabela 7.

Tabela 7 – Resultados da coleta de dados

BASE	**RESULTADOS**
IEEE *Xplore*	56
ACM	20
Scopus	11
Web of Science	1
Total	88

Fonte: Elaborada pela autora.

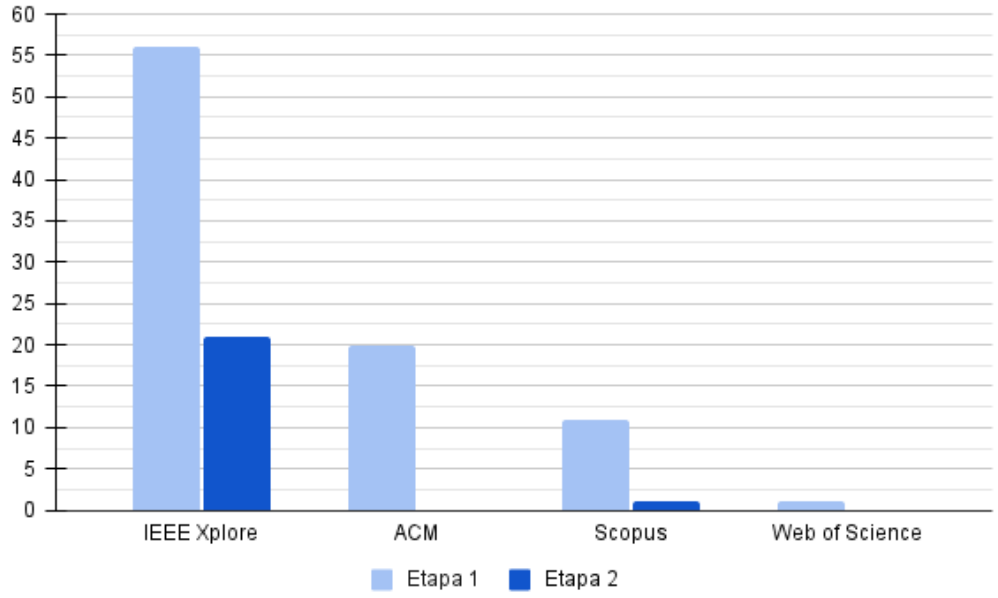
Nessa etapa, tem-se como resultado 88 trabalhos que seguiram os critérios definidos para a pesquisa. A base de dados que mais apresentou resultados compatíveis com os termos buscados foi a *IEEE Xplore* o que pode ser explicado pelo seu foco em publicações científicas e tecnológicas, reunindo uma grande quantidade de artigos sobre o tema estudado.

4.2 Resultados da segunda etapa

No Apêndice A, são apresentados os dados detalhados dos artigos pesquisados, incluindo informações sobre sua origem, data de publicação, palavras-chaves e os critérios utilizados para determinar sua inclusão ou exclusão na seleção. Através desses critérios, foi possível identificar quais artigos atendem aos requisitos necessários para avançar à terceira etapa da pesquisa, garantindo apenas os estudos mais alinhados com os objetivos da pesquisa fossem mantidos para a análise detalhada.

Após a aplicação dos critérios de inclusão (CI) e de exclusão (CE) sobre os 88 resultados obtidos, os artigos selecionados foram pesquisados com base na sua relevância e

Figura 13 – Resultados da segunda etapa



Fonte: elaborada pela autora.

adequação ao tema. Desses resultados, 56 artigos pertenciam à base de dados IEEE *Xplore*, dos quais 21 atenderam aos critérios estabelecidos. Na ACM foram encontrados 20 artigos e nenhum atendeu aos critérios desejados. Na *Scopus*, 11 artigos foram encontrados, dos quais, apenas 1 atendia aos critérios estabelecidos e na *Web of Science* foi encontrado apenas 1 artigo e ele não atendia aos critérios estabelecidos. A Figura 13 apresenta o resultado da primeira etapa em comparação com o resultado da aplicação dos critérios da segunda etapa.

O maior número de artigos selecionados da IEEE *Xplore* pode ser atribuído à sua ampla cobertura em áreas tecnológicas e de engenharia, bem como ao fato de ter maior número de publicações de alta qualidade e relevância para as áreas de IoT e segurança de protocolos. Desta forma a IEEE *Xplore* torna-se uma fonte particularmente essencial para as pesquisas nessas áreas.

4.3 Resultados da terceira etapa

Nesta etapa, os trabalhos que seguiram os critérios estabelecidos serão reavaliados. Aqueles que atendem aos requisitos necessários são selecionados e utilizados na próxima etapa para responder às questões da pesquisa. Os trabalhos foram classificados em ordem alfabética de acordo com seus respectivos títulos. Na Tabela 8 são apresentados os 22 títulos dos trabalhos, seguidos por seus autores e ano de publicação que atenderam os critérios de seleção e serão

utilizados para responder as perguntas da pesquisa.

Tabela 8 – Lista de artigos selecionados

Título	**Autor(es)**	**Ano**
A Review on the Study on MQTT Security Challenge	Fu Chen, Yujia Huo, Jianming Zhu, Dan Fan	2020
A Systematized Security and Communication Protocols Stack Review for Internet of Things	Manasha Saqib, Bhat Jasra, Ayaz Hassan Moon	2020
Addressing cybersecurity threats in prosumer-based nanogrids with MQTT communication	Pablo José Hueros-Barrios, Fco. Javier Rodríguez, Pedro Martín, Miguel Gayo, Ignacio Fernández	2022
Evaluating the Reliability of MQTT with Comparative Analysis	Yifeng Liu, Eyhab Al-Masri	2021
Fuzzing attacks for vulnerability discovery within MQTT protocol	G. Casteur, A. Aubaret, B. Blondeau, V. Clouet, A. Quemat, V. Pical, R. Zitouni	2020
Publish Subscribe System Security Requirement: A Case Study for V2V Communication	Hemant Gupta, Amiya Nayak	2024
Secure IoT Communication with a Blockchain Enabled Decentralized MQTT Network	Vladyslav Makarian	2023
Secure MQTT Authentication and Message Exchange Methods for IoT Constrained Device	Fathan Abdul Shodiq, Rizka Reza Pahlevi, Parman Sukarno	2021
Securing Internet of Vehicles Protocols using ASCON and GIFT-COFB	Wissal BenMassaoud, Darshan M, Rutvij H. Jhaveri, Gautam Srivastava	2023
Securing MQTT Ecosystem: Exploring Vulnerabilities, Mitigations, and Future Trajectories	Shams Ul Arfeen Laghari, Wenhao Li, Selvakumar Manickam, Priyadarsi Nanda, Ayman Khallel Al-Ani, Shankar Karuppayah	2024
Securing the Internet of Things: Exploring MQTT Vulnerabilities and Threats in Pakistan's IoT Landscape	Ammar Hassan, Junaid Aslam, Shahzaib Tahir, Imran Rashid	2024

The Security Method in MQTT Protocol for Internet of Things	Chia-Fen Hsieh, Chih-Kai Chang	2021
Vulnerabilities and Security Issues in IoT Protocols	Raghad M. Abdulghani, Marwa M. Alrehili, Abrar A. Almuhanna, Omar H. Alhazmi	2020
Vulnerability Assessment of MQTT Protocol in Internet of Things (IoT)	Meenaxi M Raikar, Meena S M	2021
BlockMQ: Blockchain-Assisted MQTT Authentication	Md Atiab Jobayer Purno, Rahul Bangar, Shubham Kaushal, M. Brindha, G. Saravana Ilango	2024
Incident Response in Smart Agriculture: An MQTT Case Study	Santoshi Rudrakar, Parag H. Rughani, Lakshminarayana Sadineni	2024
Pirates of the MQTT: Raiding IIoT Systems with a Rogue Client	Wael Alsabbagh, Samuel Amogbonjaye, Chaerin Kim, Peter Langendörfer	2024
Role of PKI in Securing AMQP Communication	Sanjay Adiwal, S Shuaib Ahmed, Balaji Rajendran, Mohammad Misbahuddin, S D Sudarsan	2024
Securing Shared Subscriptions in MQTTv5 for IoT Networks: Vulnerability Analysis and Mitigation	Graziano Rizzo, Mattia Giovanni Spina, Floriano De Rango	2024
Securing the IoT Application Layer From an MQTT Protocol Perspective: Challenges and Research Prospects	Lakshminarayana, Sujitha and Praseed, Amit and Thilagam, P. Santhi	2024
SGANFuzz: A Deep Learning-Based MQTT Fuzzing Method Using Generative Adversarial Networks	Zhiqiang Wei, Xijia Wei, Xinghua Zhao, Zongtang Hu, Chu Xu	2024
A Comparative Study of Post-Quantum Cryptographic Algorithm Implementations for Secure and Efficient Energy Systems Monitoring	Gandeva Bayu Satrya, Yosafat Marselino Agus, Adel Ben Mnaouer	2023

Fonte: Elaborada pela autora.

4.4 Resultados da quarta etapa

Esta etapa apresenta os resultados obtidos após a análise dos dados coletados, com objetivo em responder as questões de pesquisa.

Questão 1: De acordo com os artigos analisados, o MQTT é o protocolo mais popular tanto na indústria quanto na academia, sendo amplamente adotado em sistemas IoT devido à sua leveza e eficiência na comunicação. O artigo (CHEN *et al.*, 2020) destaca que o MQTT é amplamente utilizado para transmissão de dados na camada de aplicação da IoT, sendo um dos protocolos mais adotados no setor industrial devido à sua simplicidade e baixo consumo de recursos.

Essa popularidade também é reforçada pelo artigo (LIU; AL-MASRI, 2021), que menciona o MQTT como um dos principais protocolos de comunicação IoT, adotado em diversas aplicações devido à sua confiabilidade e flexibilidade na troca de mensagens. Além disso, o artigo (MAKARIAN, 2023) aponta que o modelo tradicional do MQTT é centralizado, o que facilita sua implementação e o torna a escolha principal para muitas aplicações IoT.

Apesar de sua ampla adoção, algumas pesquisas indicam que o protocolo ainda apresenta vulnerabilidades. O artigo (ABDULGHANI *et al.*, 2020) enfatiza que, mesmo sendo amplamente utilizado na indústria e em pesquisas acadêmicas, o MQTT possui falhas que precisam ser mitigadas. Confirmando essa análise, o artigo (RAIKAR; M, 2021) ressalta que, embora seja um dos protocolos mais utilizados em IoT, o MQTT ainda necessita de melhorias em segurança.

Diante dos resultados alcançados, é possível perceber que, dentre os protocolos que são objeto deste estudo, o MQTT é aquele que apresenta maior popularidade tanto no setor industrial quanto no acadêmico, isto se deve a sua leveza e eficiência em ambientes de redes monolíticas, é a escolha preferida para sistemas IoT, conforme evidenciado nas trabalhos analisados. A Tabela 9 faz uma síntese dos trabalhos e como esses pontos principais do MQTT são discutidos nos artigos.

Questão 2: Sobre os níveis de privacidade, os protocolos analisados apresentam diferentes abordagens para garantir a proteção dos dados transmitidos em redes IoT. Níveis de segurança e privacidade do MQTT o artigo (CHEN *et al.*, 2020) destaca que o MQTT pode adotar TLS/SSL para criptografia. Já o artigo (LIU; AL-MASRI, 2021) analisa diferentes configurações de segurança do MQTT, mostrando que a criptografia TLS v1.3 e TLSMA (TLS com autenticação mútua) oferecem boa proteção sem comprometer significativamente o desempenho.

Tabela 9 – Principais observações sobre o MQTT

Referência	Pontos destacados
(CHEN *et al.*, 2020)	MQTT é amplamente utilizado na camada de aplicação da IoT devido à sua simplicidade e baixo consumo de recursos.
(LIU; AL-MASRI, 2021)	Protocolo confiável e flexível, adotado em diversas aplicações IoT.
(MAKARIAN, 2023)	O modelo centralizado facilita a implementação e torna a escolha principal para IoT.
(ABDULGHANI *et al.*, 2020)	Apesar da utilização generalizada, possui vulnerabilidades que precisam ser mitigadas.
(RAIKAR; M, 2021)	Reforça a necessidade de melhorias na segurança.

Fonte: Elaborada pela autora.

Ademais, o artigo (SAQIB BHAT JASRA, 2020) menciona que o MQTT não possui segurança embutida, dependendo de configurações adicionais, como autenticação por usuário/senha e criptografia TLS. Além disso, o artigo (HUEROS-BARRIOS *et al.*, 2022) sugere que a autenticação baseada em senhas e o uso de ACLs (listas de controle de acesso) ajudam a reduzir os riscos, mas ainda são insuficientes sem criptografia adicional. No artigo (SHODIQ *et al.*, 2021) propõe uso de *JSON Web Token* (JWT) e criptografia leve (XXTEA) para aumentar a segurança sem comprometer o desempenho dos dispositivos.

Em relação aos níveis de segurança e privacidade do CoAP, o artigo (SAQIB BHAT JASRA, 2020) menciona que o CoAP utiliza DTLS (*Datagram* TLS) para garantir criptografia e autenticação, oferecendo um nível de segurança adequado para redes de baixa potência. Já no artigo (BENMASSAOUD *et al.*, 2023) menciona que o CoAP pode ser protegido contra ataques de espionagem e MITM com criptografia leve baseada em ASCON e GIFT-COFB.

Tabela 10 – Privacidade e segurança dos protocolos

Protocolo	Referência	Medidas de segurança
MQTT	(CHEN *et al.*, 2020)	Suporte para TLS/SSL.
MQTT	(LIU; AL-MASRI, 2021)	TLS v1.3 e TLSMA garantem boa proteção sem impacto significativo no desempenho.
MQTT	(SAQIB BHAT JASRA, 2020)	MQTT não possui segurança embutida, sendo necessárias autenticação e criptografia adicionais.
MQTT	(HUEROS-BARRIOS *et al.*, 2022)	O uso de senhas e ACLs ajuda, mas não é suficiente sem criptografia adicional.
MQTT	(SHODIQ *et al.*, 2021)	O uso de *JSON Web Token* (JWT) e XXTEA melhora a segurança sem impactar o desempenho.
CoAP	(SAQIB BHAT JASRA, 2020)	Utiliza DTLS para criptografia e autenticação.
CoAP	(BENMASSAOUD *et al.*, 2023)	Proteção contra ataques MITM com criptografia baseada em ASCON e GIFT-COFB.
AMQP	(ADIWAL *et al.*, 2024)	Suporte à autenticação robusta e criptografia TLS de ponta a ponta.
AMQP	(LAGHARI *et al.*, 2024)	A segurança depende da configuração correta dos certificados digitais.

Fonte: Elaborada pela autora.

Em contrapartida, quando ao nível de segurança e privacidade do AMQP, no artigo (ADIWAL *et al.*, 2024) afirma que o AMQP oferece um nível de segurança superior em comparação ao MQTT e CoAP, suportando autenticação robusta baseada em certificados e criptografia TLS de ponta a ponta. Além disso, o artigo (LAGHARI *et al.*, 2024) menciona que o AMQP pode oferecer segurança forte, mas depende da configuração adequada dos certificados digitais.

A Tabela 10 apresenta um resumo sintetizado sobre segurança e privacidade nos protocolos MQTT, CoAP e AMQP, destacando as principais medidas de proteção identificadas nos trabalhos selecionados.

Questão 3: Em relação às falhas de segurança conhecidas e como podem ser exploradas no protocolo MQTT, o artigo (CHEN *et al.*, 2020) aponta que o MQTT é suscetível a ataques DoS, *Spoofing*, MITM e *replay* devido à falta de segurança embutida. Da mesma forma, (CASTEUR *et al.*, 2020) explora o uso de ataques de *fuzzing* para identificar falhas, como *buffer overflow* e comportamentos inesperados do *broker*.

A insegurança do MQTT também é abordada em (HUEROS-BARRIOS *et al.*, 2022), que alerta para a possibilidade de ataques de replay e alteração de pacotes sem que o destinatário perceba. No artigo (ALSABBAGH *et al.*, 2024), destaca-se que a falta de autenticação permite que dispositivos não autorizados publiquem mensagens maliciosas, viabilizando ataques de controle remoto.

Além disso, (WEI *et al.*, 2024) identifica seis vulnerabilidades no protocolo, incluindo *buffer overflow* e pacotes malformados capazes de derrubar *brokers* MQTT. Já (RAIKAR; M, 2021) avalia falhas estruturais do MQTT, como a ausência de criptografia padrão, tornando-o vulnerável a ataques MITM. Por fim, o estudo (LAKSHMINARAYANA *et al.*, 2024) aborda ataques de *replay*, onde comandos antigos podem ser reutilizados indevidamente, representando um risco significativo para a segurança do protocolo.

Já em relação às falhas de segurança conhecidas e como elas podem ser exploradas no protocolo CoAP, o artigo (SAQIB BHAT JASRA, 2020) menciona que o CoAP é vulnerável a ataques MITM e *replay*, pois opera sobre UDP. Da mesma forma, (HASSAN *et al.*, 2024) alerta que o CoAP pode permitir ataques de espionagem e manipulação de pacotes.

O estudo (ABDULGHANI *et al.*, 2020) reforça essas vulnerabilidades, destacando que a estrutura do CoAP o torna suscetível a ataques MITM e *replay* devido ao uso do UDP. Além disso, o artigo (GUPTA; NAYAK, 2024) aponta que a ausência de autenticação forte no CoAP pode facilitar a interceptação de pacotes e a falsificação de mensagens, comprometendo a

segurança das comunicações.

Em relação às falhas de segurança conhecidas e como podem ser exploradas no protocolo AMQP, o artigo (ADIWAL *et al.*, 2024) afirma que o AMQP pode sofrer ataques DoS e *spoofing* caso a autenticação baseada em certificados não seja corretamente configurada. Da mesma forma, (PURNO *et al.*, 2024) destaca que o AMQP pode ser alvo de ataques MITM caso os certificados digitais não sejam gerenciados de forma adequada, comprometendo a integridade da comunicação. A Tabela 11 apresenta um resumo das principais falhas de segurança identificadas nos protocolos MQTT, CoAP e AMQP, com base na literatura analisada.

Tabela 11 – Falhas de segurança dos protocolos

Protocolo	**Referência**	**Falhas de segurança**
MQTT	(CHEN *et al.*, 2020)	Suscetível a ataques DoS, *spoofing*, MITM e *replay*.
MQTT	(CASTEUR *et al.*, 2020)	Identificação de *buffer overflow* e falhas via *fuzzing*.
MQTT	(HUEROS-BARRIOS *et al.*, 2022)	Possibilidade de ataques de repetição e alteração de pacotes.
MQTT	(ALSABBAGH *et al.*, 2024)	A falta de autenticação permite a publicação de mensagens maliciosas.
MQTT	(WEI *et al.*, 2024)	Seis vulnerabilidades encontradas, incluindo *buffer overflow* e pacotes malformados.
MQTT	(RAIKAR; M, 2021)	Ausência de criptografia padrão expõe ataques MITM.
MQTT	(LAKSHMINARAYANA *et al.*, 2024)	Comandos antigos podem ser reutilizados em ataques de repetição.
CoAP	(SAQIB BHAT JASRA, 2020)	Vulnerável a ataques MITM e *replay*.
CoAP	(HASSAN *et al.*, 2024)	Pode permitir espionagem e manipulação de pacotes.
CoAP	(ABDULGHANI *et al.*, 2020)	O uso do UDP aumenta a vulnerabilidade aos ataques MITM e *replay*.
CoAP	(GUPTA; NAYAK, 2024)	Falta de autenticação forte facilita interceptação de pacotes e falsificação de mensagens.
AMQP	(ADIWAL *et al.*, 2024)	Pode sofrer ataques DoS e *spoofing* caso a autenticação não seja configurada corretamente.
AMQP	(PURNO *et al.*, 2024)	Vulnerável a ataques MITM se os certificados digitais não forem gerenciados corretamente.

Fonte: Elaborada pela autora.

Questão 4: Para a pergunta "Existem possíveis medidas para resolver essas falhas de segurança?", algumas soluções são propostas, para o protocolo MQTT, o artigo (CHEN *et al.*, 2020) sugere a implementação de criptografia leve com ECC e RSA para minimizar o consumo de recursos, além do uso de mecanismos de detecção de intrusão (IDS) baseados em aprendizado de máquina para identificar tráfego anômalo e prevenir ataques como DoS e MITM. Também destaca a aplicação de *blockchain* para autenticar dispositivos IoT de forma descentralizada e o uso de HMAC para garantir a integridade e confidencialidade das mensagens.

No mesmo contexto, o artigo (SHODIQ *et al.*, 2021) propõe autenticação forte com JWT (*JSON Web Token*) e criptografia leve com XXTEA para proteger as mensagens MQTT sem

comprometer o desempenho. Já (LIU; AL-MASRI, 2021) enfatiza a importância da adoção de TLS v1.3 e TLSMA para proteger a comunicação sem impactos significativos no desempenho.

Além disso, (CASTEUR *et al.*, 2020) recomenda habilitar TLS na porta 8883, uma vez que muitos servidores ainda operam sem criptografia na porta 1883, tornando-se alvos fáceis de interceptação. Por fim, (PURNO *et al.*, 2024) sugere a utilização de *blockchain* para autenticação descentralizada, eliminando a dependência de um único *broker* vulnerável.

O artigo (SAQIB BHAT JASRA, 2020) sugere o uso de DTLS (Datagram TLS) para criptografia e autenticação, além de um controle de acesso rigoroso para evitar que dispositivos não autorizados se comuniquem na rede (*Security for IoT Protocols*). Por outro lado, o artigo (BENMASSAOUD *et al.*, 2023) propõe a utilização de criptografia leve para o CoAP, empregando os algoritmos ASCON e GIFT-COFB, que são otimizados para dispositivos IoT com restrições de *hardware*.

O artigo (ADIWAL *et al.*, 2024) sugere o uso obrigatório de TLS 1.3 para proteger os dados em trânsito, autenticação baseada em certificados digitais para evitar ataques MITM e monitoramento contínuo do tráfego para detectar atividades suspeitas. Além disso, o artigo (LAGHARI *et al.*, 2024) reforça que o AMQP pode ser protegido contra ataques MITM por meio de um sistema de autenticação forte e controle granular de acesso.

Tabela 12 – Medidas de segurança propostas para os protocolos

Protocolo	**Referência**	**Medidas de segurança propostas**
MQTT	(CHEN *et al.*, 2020)	Criptografia leve (ECC, RSA); IDS baseado em aprendizado de máquina para detecção de tráfego anômalo; uso de *blockchain* para autenticação descentralizada; HMAC para integridade e confidencialidade.
MQTT	(SHODIQ *et al.*, 2021)	Autenticação forte com JWT (*JSON Web Token*); criptografia leve com XXTEA para proteger mensagens sem comprometer o desempenho.
MQTT	(LIU; AL-MASRI, 2021)	Adoção de TLS v1.3 e TLSMA para comunicação segura, sem impactos significativos no desempenho.
MQTT	(CASTEUR *et al.*, 2020)	Habilitação de TLS na porta 8883, evitando vulnerabilidades comuns na porta 1883.
MQTT	(PURNO *et al.*, 2024)	Uso de *blockchain* para autenticação descentralizada, eliminando a dependência de um único *broker*.
CoAP	(SAQIB BHAT JASRA, 2020)	Implementação de DTLS para criptografia e autenticação; controle de acesso rigoroso para impedir a comunicação de dispositivos não autorizados.
CoAP	(BENMASSAOUD *et al.*, 2023)	Criptografia leve utilizando algoritmos ASCON e GIFT-COFB, otimizados para dispositivos IoT com restrições de hardware.
AMQP	(ADIWAL *et al.*, 2024)	Uso obrigatório de TLS 1.3; autenticação baseada em certificados digitais para prevenir ataques MITM; monitoramento contínuo do tráfego para identificar atividades suspeitas.
AMQP	(LAGHARI *et al.*, 2024)	Sistema de autenticação forte e controle granular de acesso para mitigar ataques MITM.

Fonte: Elaborada pela autora.

A Tabela 12 apresenta algumas estratégias para aumentar a segurança dos protocolos de comunicação em ambientes IoT. As soluções propostas buscam reduzir falhas comuns através de técnicas como criptografia leve (ECC, RSA e XXTEA), garantindo proteção sem prejudicar o desempenho dos dispositivos. Essas abordagens aumentam a integridade, a confidencialidade e a proteção dos dados nas redes IoT, contribuindo para um ambiente mais seguro e confiável.

5 CONCLUSÕES E TRABALHOS FUTUROS

A Internet das Coisas está mudando a forma como os dispositivos se comunicam e interagem no mundo digital, permitindo maior automação, eficiência e conectividade em vários campos, incluindo indústria, saúde e casas inteligentes. No entanto, esta conectividade também traz desafios significativos à segurança da informação, uma vez que muitos dispositivos têm recursos computacionais limitados, dificultando a implementação de mecanismos de proteção fortes.

Diante dessa situação, este trabalho tem como objetivo analisar as vulnerabilidades presentes nos protocolos de aplicação MQTT, CoAP e AMQP que são amplamente utilizados para comunicação entre dispositivos IoT. Foi realizada uma revisão sistemática da literatura para identificar os principais riscos associados a esses protocolos e avaliar as estratégias mais eficazes para mitigar ataques cibernéticos e proteger a integridade dos dados transmitidos.

A relevância deste estudo reside na crescente popularidade dos dispositivos IoT e na necessidade de tornar estas tecnologias mais seguras e confiáveis. A falta de padronização de segurança e a adoção de protocolos sem criptografia adequada tornam os dispositivos alvos fáceis para invasores que podem explorar vulnerabilidades para realizar roubo de dados, ataques de negação de serviço (DoS/DDoS) e interceptação de mensagens (*man-in-the-middle* - MITM).

Para verificar se os objetivos foram alcançados, a funcionalidade dos protocolos estudados é analisada detalhadamente, destacando suas vulnerabilidades. Os resultados obtidos confirmam que o MQTT é um protocolo leve e eficiente, amplamente utilizado na área de IoT devido ao seu baixo consumo de recursos. Porém, sua segurança é limitada por não possuir mecanismos nativos de criptografia e autenticação, tornando-o vulnerável a ataques caso medidas adicionais como TLS (*Transport Layer Security*) e autenticação forte não sejam implementadas.

Além disso, como o protocolo CoAP utiliza UDP para transmissão de dados, existe um alto risco de ataques de *spoofing*, DoS e interceptação de mensagens, pois o protocolo não garante a entrega confiável de pacotes de dados. Para mitigar estas vulnerabilidades, recomenda-se a implementação de DTLS (*Datagram Transport Layer Securit*y), que adiciona uma camada de proteção às comunicações.

Por sua vez, o AMQP se destaca por oferecer segurança e confiabilidade superiores, pois suporta criptografia e controle de acesso mais granular. Porém, seu alto consumo de recursos pode dificultar sua adoção em dispositivos IoT com *hardware* limitado, tornando-o mais adequado para aplicações onde a segurança é maior que a eficiência energética.

Portanto, os protocolos MQTT, CoAP e AMQP possuem vulnerabilidades que podem comprometer a segurança da IoT, foi confirmada. Este trabalho constatou que a ausência de criptografia, a falta de autenticação confiável e o uso de protocolos de transferência de dados inseguros representam riscos significativos para a integridade e confiabilidade da IoT.

Diante disso, para trabalhos futuros torna-se importante ampliar o período de publicação da análise e assim construir uma base de dados mais abrangente. Adicionalmente, é necessário considerar a extensão do estudo a outras plataformas de pesquisas científicas que complementem as fontes utilizadas neste estudo. Também é importante adotar uma gama mais ampla de sinônimos e termos relacionados para ampliar o escopo dos resultados e identificar estudos relevantes que possam ter sido excluídos deste trabalho.

REFERÊNCIAS

ABDULGHANI, R. M.; ALREHILI, M. M.; ALMUHANNA, A. A.; ALHAZMI, O. H. Vulnerabilities and security issues in iot protocols. In: **2020 First International Conference of Smart Systems and Emerging Technologies (SMARTTECH)**. [*S. n.*], 2020. p. 7–12. Disponível em: https://ieeexplore.ieee.org/document/9283795. Acesso em: 07 jan. 2025.

ABNT. **NBR ISO/IEC 27002:2005 Tecnologia da informação: técnicas de segurança**. Rio de Janeiro, 2006.

ADIWAL, S.; AHMED, S. S.; RAJENDRAN, B.; MISBAHUDDIN, M.; SUDARSAN, S. D. Role of pki in securing amqp communication. In: **2024 IEEE International Conference on Public Key Infrastructure and its Applications (PKIA)**. [*S. n.*], 2024. p. 1–8. Disponível em: https://ieeexplore.ieee.org/document/10729177. Acesso em: 07 jan. 2025.

AL-FUQAHA, A.; GUIZANI, M.; MOHAMMADI, M.; ALEDHARI, M.; AYYASH, M. Internet of things: A survey on enabling technologies, protocols, and applications. **IEEE Communications Surveys & Tutorials**, IEEE, v. 17, n. 4, p. 2347–2376, 2015. Disponível em: https://ieeexplore.ieee.org/document/7123563. Acesso em: 10 jan. 2025.

AL., L. M. A. et. A state of art for smart gateways issues and modification. **Asian J. Res. Comput. Sci.**, v. 7, n. 4, p. 1–13, Apr. 2021. Disponível em: https://www.researchgate.net/publication/350823622_A_State_of_Art_for_Smart_Gateways_Issues_and_Modification. Acesso em: 25 jan. 2025.

ALBERTIN, A. L. A. E. R. M. D. M. A internet das coisas irá muito além das coisas. **GV Executivo**, p. 6, 2017. Disponível em: https://periodicos.fgv.br/gvexecutivo/article/view/68668/66258. Acesso em: 10 dez. 2024.

ALSABBAGH, W.; AMOGBONJAYE, S.; KIM, C.; LANGENDöRFER, P. Pirates of the mqtt: Raiding iiot systems with a rogue client. In: **2024 8th Cyber Security in Networking Conference (CSNet)**. [*S. n.*], 2024. p. 248–253. Disponível em: https://ieeexplore.ieee.org/document/10851733. Acesso em: 07 jan. 2025.

ASHTON, K. That 'internet of things' thing. **RFID Journal**, 2009. Disponível em: https://www.rfidjournal.com/articles/view?4986. Acesso em: 10 dez. 2024.

ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. **Computer Networks**, Elsevier, v. 54, 2010. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S1389128610001568. Acesso em: 15 jan. 2025.

BARROS, E. de. **ESTUDO DE VULNERABILIDADES EM DISPOSITIVOS IOT TCP/IP**. 2021. Disponível em: https://lume.ufrgs.br/handle/10183/222479. Acesso em: 21 jan. 2025.

BASTOS, A.; CAUBIT, R. **Gestão de Segurança da Informação: ISO 27001 e 27002 - Uma Visão Prática**. Rio Grande do Sul: Zouk, 2009.

BEAL, A. **Segurança da Informação: Princípios e Melhores Práticas para a Proteção dos Ativos de Informação nas Organizações**. Reimpressão 2008. São Paulo: Atlas, 2005.

BENMASSAOUD, W.; M, D.; JHAVERI, R. H.; SRIVASTAVA, G. Securing internet of vehicles protocols using ascon and gift-cofb. In: **2023 IEEE 97th Vehicular**

Technology Conference (VTC2023-Spring). [*S. n.*], 2023. p. 1–7. Disponível em: https://ieeexplore.ieee.org/document/10199329. Acesso em: 07 jan. 2025.

BLOG, S. **About Scopus Blog**. Elsevier, 2025. Disponível em: https://blog.scopus.com/about. Acesso em: 05 fev. 2025.

BUJARI, A.; FURINI, M.; MANDREOLI, F.; MARTOGLIA, R.; MONTANGERO, M.; RONZANI, D. Standards security and business models: Key challenges for the iot scenario. **Mobile Netw. Appl.**, v. 23, n. 1, p. 147–154, Feb. 2018. Disponível em: https://doi.org/10.1007/s11036-017-0835-8. Acesso em: 10 jan. 2025.

CARMO, T. de Russo e. **Uso do padrão AMQP para transporte de mensagens entre atores remotos**. Dissertação (Mestrado) – Instituto de Matemática e Estatística, Universidade de São Paulo, 2012. Disponível em: https://www.ime.usp.br/~reverbel/students/master_theses/thadeu_de_russo_e_carmo.pdf. Acesso em: 03 fev. 2025.

CASTEUR, G.; AUBARET, A.; BLONDEAU, B.; CLOUET, V.; QUEMAT, A.; PICAL, V.; ZITOUNI, R. Fuzzing attacks for vulnerability discovery within mqtt protocol. In: **2020 International Wireless Communications and Mobile Computing (IWCMC)**. [*S. n.*], 2020. p. 420–425. Disponível em: https://ieeexplore.ieee.org/document/9148320. Acesso em: 07 jan. 2025.

CEDALO. **Complete MQTT Protocol Guide**. 2023. Disponível em: https://cedalo.com/blog/complete-mqtt-protocol-guide/#What_are_the_standard_MQTT_ports_and_why. Acesso em: 08 jan. 2025.

CHEN, F.; HUO, Y.; ZHU, J.; FAN, D. A review on the study on mqtt security challenge. In: **2020 IEEE International Conference on Smart Cloud (SmartCloud)**. [*S. n.*], 2020. p. 128–133. Disponível em: https://ieeexplore.ieee.org/document/9265962. Acesso em: 07 jan. 2025.

CLARIVATE. **Academia Government**. Clarivate, 2025. Disponível em: https://clarivate.com/academia-government/. Acesso em: 21 nov. 2024.

DIAS, C. **Segurança e Auditoria da Tecnologia da Informação**. Rio de Janeiro: Axcel Books, 2000.

DIGITAL, A. **IoT Protocols: A Comprehensive Guide**. 2025. Disponível em: https://www.a1.digital/news/iot-protocols-a-comprehensive-guide/. Acesso em: 17 jan. 2025.

DINCULEANÃ, D.; CHENG, X. Vulnerabilities and limitations of mqtt protocol used between iot devices. **Applied Sciences**, v. 9, n. 5, p. 848, 2019. Disponível em: https://www.mdpi.com/2076-3417/9/5/848. Acesso em: 10 jan. 2025.

DIZDAREVIC, J.; CARPIO, F.; JUKAN, A.; MASIP-BRUIN, X. Survey of communication protocols for internet-of-things and related challenges of fog and cloud computing integration. **ACM Comput. Surveys**, v. 51, n. 6, p. 1–29, 2019. Disponível em: https://doi.org/10.1145/3292674. Acesso em: 25 jan. 2025.

EMNIFY. **Segurança IoT: Tudo o que você precisa saber**. 2023. Disponível em: https://www.emnify.com/pt-br/glossario-iot/seguranca-iot. Acesso em: 07 jan. 2025.

FERREIRA, P. B. V. **Arquitetura REST em smartphones Android**. Dissertação (Master's thesis) – Instituto Politécnico do Porto, Porto, Portugal, 2015. Disponível em: https://recipp.ipp.pt/bitstream/10400.22/8226/1/DM_PedroFerreira_2015_MEI.pdf. Acesso em: 07 jan. 2025.

FIELDING, R. T. **Architectural styles and the design of network-based software architectures**. Tese (Ph.D. dissertation) – University of California, Irvine, CA, USA, 2000. Inf. Comput. Sci. Disponível em: https://ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf. Acesso em: 10 jan. 2025.

FILHO, M. R.; ISLABãO, G. I. de; SANTOS, D. A. **Aplicação do Paradigma "Internet das Coisas" para o Setor de Serviços e Cuidados com a Saúde: caminhos e desaios de desenvolvimento do setor a partir de análise patentária**. 2015. Disponível em: https://periodicos.ufba.br/index.php/nit/article/view/55738/31031. Acesso em: 10 jan. 2025.

FRIESS, P. **Internet of Things: Converging Technologies for Smart Environments and Integrated Ecosystems**. Aalborg, Denmark: River Publishers, 2013.

GUBBI, S.; BUYYA, R.; MARUSIC, S.; PALANISWAMI, M. Internet of things (iot): A vision, architectural elements, and security issues. **Future Generation Computer Systems**, Elsevier, v. 29, 2013. Disponível em: https://www.sciencedirect.com/science/article/abs/pii/S0167739X13000241. Acesso em: 10 dez. 2024.

GUPTA, B. B.; QUAMARA, M. An overview of internet of things (iot): Architectural aspects challenges and protocols. **Concurrency Comput. Pract. Exp.**, v. 32, n. 21, Nov. 2020. Disponível em: https://doi.org/10.1002/cpe.4946. Acesso em: 10 jan. 2025.

GUPTA, H.; NAYAK, A. Publish subscribe system security requirement: A case study for v2v communication. **IEEE Open Journal of the Computer Society**, v. 5, p. 389–405, 2024. Disponível em: https://ieeexplore.ieee.org/document/10636299. Acesso em: 07 jan. 2025.

HASSAN, A.; ASLAM, J.; TAHIR, S.; RASHID, I. Securing the internet of things: Exploring mqtt vulnerabilities and threats in pakistan's iot landscape. In: **2024 International Conference on Engineering Computing Technologies (ICECT)**. [*S. n.*], 2024. p. 1–6. Disponível em: https://ieeexplore.ieee.org/document/10581385. Acesso em: 07 jan. 2025.

HIVEMQ. **MQTT Essentials Part 1: Introducing MQTT**. 2024. Disponível em: https://www.hivemq.com/blog/mqtt-essentials-part-1-introducing-mqtt/. Acesso em: 08 jan. 2025.

HUEROS-BARRIOS, P. J.; RODRíGUEZ, F. J.; MARTíN, P.; GAYO, M.; FERNáNDEZ, I. Addressing cybersecurity threats in prosumer-based nanogrids with mqtt communication. In: **2022 International Conference on Electrical, Computer and Energy Technologies (ICECET)**. [*S. n.*], 2022. p. 1–6. Disponível em: https://ieeexplore.ieee.org/document/9872918. Acesso em: 07 jan. 2025.

IBM, E. **MQTT Version 3.1 Protocol Specification**. 2010. Disponível em: https://public.dhe.ibm.com/software/dw/webservices/ws-mqtt/mqtt-v3r1.html. Acesso em: 31 jan. 2025.

ISLAM, M. M.; NOORUDDIN, S.; KARRAY, F.; MUHAMMAD, G. Internet of things: Device capabilities, architectures, protocols, and smart applications in healthcare domain.

IEEE Internet of Things Journal, v. 10, n. 4, p. 3611–3641, 2023. Disponível em: https://ieeexplore.ieee.org/document/9983826. Acesso em: 21 jan. 2025.

ITU-T. **Recommendation ITU-T Y.2060: Overview of the Internet of Things**. [*S. l.*], 2012. Disponível em: https://www.itu.int/ITU-T/recommendations/rec.aspx?rec=y.2060. Acesso em: 10 jan. 2025.

JUNIOR, S. L. da R. **Segurança em Sistemas IoT: vulnerabilidades e mecanismos de prevenção**. Trabalho de Conclusão de Curso (Graduação), Florianópolis, 2022. Orientador: Prof. Richard Demo Souza, Dr. Disponível em: https://repositorio.ufsc.br/handle/123456789/233379. Acesso em: 07 jan. 2025.

KASPERSKY. **Melhores práticas para segurança IoT**. 2020. Disponível em: https://www.kaspersky.com.br/resource-center/preemptive-safety/best-practices-for-iot-security. Acesso em: 07 jan. 2025.

KITRUM. **Three-Layer Architecture in the Internet of Things**. 2023. https://kitrum.com/blog/three-layer-architecture-in-the-internet-of-things/. Acesso em: 16 nov. 2024.

LAGHARI, S. U. A.; LI, W.; MANICKAM, S.; NANDA, P.; AL-ANI, A. K.; KARUPPAYAH, S. Securing mqtt ecosystem: Exploring vulnerabilities, mitigations, and future trajectories. **IEEE Access**, v. 12, p. 139273–139289, 2024. Disponível em: https://ieeexplore.ieee.org/document/10552255. Acesso em: 07 jan. 2025.

LAKSHMINARAYANA, S.; PRASEED, A.; THILAGAM, P. S. Securing the iot application layer from an mqtt protocol perspective: Challenges and research prospects. **IEEE Communications Surveys Tutorials**, v. 26, n. 4, p. 2510–2546, 2024. Disponível em: https://ieeexplore.ieee.org/document/10458418. Acesso em: 07 jan. 2025.

LIGHT, R. **Understanding MQTT Quality of Service or also known as MQTT QoS**. 2022. https://cedalo.com/blog/understanding-mqtt-qos/. Acesso em: 08 jan. 2025.

LIU, Y.; AL-MASRI, E. Evaluating the reliability of mqtt with comparative analysis. In: **2021 IEEE 4th International Conference on Knowledge Innovation and Invention (ICKII)**. [*S. n.*], 2021. p. 24–29. Disponível em: https://ieeexplore.ieee.org/document/9574783. Acesso em: 07 jan. 2025.

LYRA, M. R. **Segurança e Auditoria em Sistemas de Informação**. Rio de Janeiro: Ciência Moderna, 2008.

MACHINERY, A. for C. **ACM History**. ACM, 2025. Disponível em: https://www.acm.org/about-acm/acm-history. Acesso em: 05 fev. 2025.

MAGRINI, E. **A Internet das Coisas**. Fundação Getulio Vargas, 2018. P. 20. Disponível em: https://repositorio.fgv.br/server/api/core/bitstreams/b50af2ba-b001-4b1d-a1ad-5df985f6d1bb/content. Acesso em: 10 dez. 2024.

MAKARIAN, V. Secure iot communication with a blockchain-enabled decentralized mqtt network. In: **2023 IEEE 12th International Conference on Intelligent Data Acquisition and Advanced Computing Systems: Technology and Applications (IDAACS)**. [*S. n.*], 2023. v. 1, p. 828–833. Disponível em: https://ieeexplore.ieee.org/document/10348690. Acesso em: 07 jan. 2025.

MARTINS, I. R.; ZEM, J. L. Estudo dos protocolos de comunicação mqtt e coap para aplicações machine-to-machine e internet das coisas. **Revista Tecnológica da Fatec Americana**, v. 3, n. 1, p. 64–87, 2015. Disponível em: https://ric.cps.sp.gov.br/bitstream/123456789/107/1/estudo_protocolos_comunicacao_mqtt_coap_aplicacoes_machine_internet_coisas.pdf. Acesso em: 31 jan. 2025.

MOHAMED, K. S. Iot networking and communication layer. In: ____. **The Era of Internet of Things**. Cham, Switzerland: Springer Int., 2019. p. 49–70.

NIST. **Networks of Things**. [*S. l.*], 2016. Disponível em: https://nvlpubs.nist.gov/nistpubs/SpecialPublications/NIST.SP.800-183.pdf. Acesso em: 10 jan. 2025.

NIST. **Considerations for Managing Internet of Things (IoT) Cybersecurity and Privacy Risks**. [*S. l.*], 2019. Disponível em: https://doi.org/10.6028/NIST.IR.8228. Acesso em: 15 jan. 2025.

PALMACCIO, M.; DICUONZO, G.; BELYAEVA, Z. S. The internet of things and corporate business models: A systematic literature review. **J. Bus. Res.**, v. 131, p. 610–618, Jul. 2021. Disponível em: https://doi.org/10.1016/j.jbusres.2020.09.069. Acesso em: 10 jan. 2025.

PURNO, M. A. J.; BANGAR, R.; KAUSHAL, S.; BRINDHA, M.; ILANGO, G. S. Blockmq: Blockchain assisted mqtt authentication. In: **2024 First International Conference on Pioneering Developments in Computer Science Digital Technologies (IC2SDT)**. [*S. n.*], 2024. p. 522–527. Disponível em: https://ieeexplore.ieee.org/document/10696441. Acesso em: 07 jan. 2025.

RADWARE. **Mirai - DDoSPedia**. 2025. Disponível em: https://www.radware.com/security/ddos-knowledge-center/ddospedia/mirai/. Acesso em: 07 jan. 2025.

RAIKAR, M. M.; M, M. S. Vulnerability assessment of mqtt protocol in internet of things (iot). In: **2021 2nd International Conference on Secure Cyber Computing and Communications (ICSCCC)**. [*S. n.*], 2021. p. 535–540. Disponível em: https://ieeexplore.ieee.org/document/9478156. Acesso em: 07 jan. 2025.

RIOLINO, L. **Spoofing de IP: o que é e como identificar e prevenir a fraude**. 2023. Disponível em: https://www.locaweb.com.br/blog/temas/seguranca-digital/spoofing-de-ip-o-que-e/. Acesso em: 21 jan. 2025.

ROISENBERG, L. **Análise Comparativa do MQTT com Outros Protocolos como CoAP, AMQP e WebSocket**. 2024. Disponível em: https://blog.lri.com.br/analise-comparativa-do-mqtt-com-outros-protocolos-como-coap-amqp-e-websocket/#AMQP_Uma_visao_geral_do_protocolo. Acesso em: 03 fev. 2025.

SANTO, W.; ORDOñEZ, E.; RIBEIRO, A. Uma revisão sistemática sobre a segurança nos protocolos de comunicação para internet das coisas. **Journal on Advances in Theoretical and Applied Informatics**, v. 4, n. 1, p. 1–9, 2018. ISSN 2447-5033. Disponível em: https://revista.univem.edu.br/jadi/article/view/2482. Acesso em: 07 jan. 2025.

SAQIB BHAT JASRA, A. H. M. M. A systematized security and communication protocols stack review for internet of things. **IEEE Access**, v. 9, p. 1–10, 2020. Disponível em: https://ieeexplore.ieee.org/document/9298196. Acesso em: 07 jan. 2025.

SÊMOLA, M. **Gestão da Segurança da Informação: Uma visão executiva**. 11ª reimpressão. ed. Rio de Janeiro: Elsevier, 2003.

SERRANO, T. M. **AMQP – Protocolo de Comunicação para IoT**. Brasil: Embarcados, 2018. Disponível em: https://embarcados.com.br/amqp-protocolo-de-comunicacao-para-iot/. Acesso em: 05 jan. 2025.

SHELBY, Z. **CoAP: The Web of Things Protocol**. 2014. Disponível em: https://pt.slideshare.net/slideshow/coap-tutorial/21464401#19. Acesso em: 03 fev. 2025.

SHELBY, Z.; HARTKE, K.; BORMANN, C. **The Constrained Application Protocol (CoAP)**. RFC Editor, 2014. Request for Comments 7252. Disponível em: https://www.rfc-editor.org/rfc/rfc7252.html. Acesso em: 03 fev. 2025.

SHODIQ, F. A.; PAHLEVI, R. R.; SUKARNO, P. Secure mqtt authentication and message exchange methods for iot constrained device. In: **2021 International Conference on Intelligent Cybernetics Technology Applications (ICICyTA)**. [*S. n.*], 2021. p. 70–74. Disponível em: https://ieeexplore.ieee.org/document/9689126. Acesso em: 07 jan. 2025.

SILVA, A. L. F. da. **Análise das Vulnerabilidades de Protocolos de Camada de Aplicação IoT: Uma Revisão Sistemática de Literatura**. 2024. Trabalho de Conclusão de Curso, Universidade Federal Rural do Semi-Árido. Disponível em: https://repositorio.ufersa.edu.br/items/f21281f9-14a1-4ea3-bed1-805a2d279154. Acesso em: 07 jan. 2025.

SOUZA, C. P. **Estudo dos Protocolos de Comunicação MQTT e CoAP e suas Aplicações em Machine-to-Machine e Internet das Coisas**. 2017. Disponível em: https://ric.cps.sp.gov.br/bitstream/123456789/107/1/estudo_protocolos_comunicacao_mqtt_coap_aplicacoes_machine_internet_coisas.pdf. Acesso em: 03 fev. 2025.

UPADHYAY, P.; UPADHYAY, P. D. Internet of things—a survey. **Int. J. Sci. Res. Comput. Sci. Eng. Inf. Technol.**, v. 54, n. 12, p. 417–438, May 2021. Disponível em: https://www.researchgate.net/publication/352390551_Internet_of_things_-_A_Survey. Acesso em: 07 jan. 2025.

VAILSHERY, L. S. **Número de conexões de IoT em todo o mundo 2022-2033**. 2024. Disponível em: https://www-statista-com.translate.goog/?_x_tr_sl=en&_x_tr_tl=pt&_x_tr_hl=pt&_x_tr_pto=sge&_x_tr_hist=true#statisticContainer. Acesso em: 15 dez. 2024.

WEI, Z.; WEI, X.; ZHAO, X.; HU, Z.; XU, C. Sganfuzz: A deep learning-based mqtt fuzzing method using generative adversarial networks. **IEEE Access**, v. 12, p. 27210–27224, 2024. Disponível em: https://ieeexplore.ieee.org/document/10433531. Acesso em: 07 jan. 2025.

XPLORE, I. **About IEEE Xplore**. IEEE, 2025. Disponível em: https://ieeexplore.ieee.org/Xplorehelp/overview-of-ieee-xplore/about-ieee-xplore. Acesso em: 05 fev. 2025.

ZHANG, J.; YE, Y.; HU, C.; LI, B. Architecture design and demand analysis on application layer of standard system for ubiquitous power internet of things. **Global Energy Interconnect.**, v. 4, n. 3, p. 304–314, Jun. 2021. Disponível em: https://www.sciencedirect.com/science/article/pii/S2096511721000621. Acesso em: 16 nov. 2024.

APÊNDICE A – SELEÇÃO E CRITÉRIOS

Cód	Título	Ano	*Keywords*	Critérios
01	*The Security Method in MQTT Protocol for Internet of Things*	2021	*MQTT protocol, Message Queuing Telemetry Transport, Payloads, Certification*	CI1 e CI2
01	*Securing MQTT Ecosystem: Exploring Vulnerabilities, Mitigations, and Future Trajectories*	2024	*IoT security, MQTT attacks, MQTT ecosystem, MQTT security*	CI1, CI2 e CI3
01	*A Lightweight Authentication Protocol for IoT-Based Applications Using Reconfigurable Noisy PUFs*	2024	*Authentication protocol, industrial Internet of Things (IIoT), Internet of Things (IoT), network security, noisy physically unclonable function (PUF), physical unclonable functions,smart sensors*	CE1
01	*A Review on the Study on MQTT Security Challenge*	2020	MQTT, IoT, IoT *Security*	CI1, CI2
01	*Cyber-Security of Industrial Internet of Things in Electric Power Systems*	2022	*Cyber-attacks, cyber-security, electric power systems, industrial internet of things, intrusion detection systems*	CE2
01	*An Efficient Authentication Scheme for Secure Communication between Industrial IoT Devices*	2020	*Industrial Internet of Things (IIoT), security, machine-to-machine (M2M) communications, authentication*	CE3
01	*PUF-Based Authentication System with Resilience against Multi-Faceted Attacks for Blockchain-based IoT Networks*	2024	*blockchain, bribe attack, brute force attack, dictionary attack , IoT, long-range attack, precomputing attack. PUF, sybil attack*	CE3

01	*Light-Weight Security Protocol and Data Model for Chip-to-Chip Zero-Trust*	2023	*Secure protocol and data model (SPDM), automatic verification of internet security protocols and applications (AVISPA), secure protocol animator (SPAN), formal verification (FV)*	CE3
01	*Fuzzing attacks for vulnerability discovery within MQTT protocol*	2020	*MQTT Protocol, IoT Security, Fuzzing,Docker, Containerization*	C1
01	*Securing the Internet of Things: Exploring MQTT Vulnerabilities and Threats in Pakistan's IoT Landscape*	2024	*MQTT vulnerabilities, IoT Security, Privacy, Shodan, Network Enumeration*	C1
01	*Vulnerability assessment of MQTT protocol in Internet of Things (IoT)*	2021	*Publish, IP crawler, Subscribe, Broker, Shodan search engine*	C1
01	*Machine Learning-based Vulnerability Study of Interpose PUFs as Security Primitives for IoT Networks*	2021	*physical unclonable function, Interpose PUF, machine learning, neural network, FPGA*	CE4
01	*LEAF-IIoT: Lightweight and Efficient Authentication Framework for the Industrial Internet of Things*	2024	*Communication security, IIoT, smart device, resource constrained*	CE4
01	*A Systematized Security and Communication Protocols Stack Review for Internet of Things*	2020	*Internet of Things, IoT security, Message Queue Telemetry Transport*	C1
01	*Evaluating the Reliability of MQTT with Comparative Analysis*	2021	*MQTT, Security, Reliability, Edge Computing, Internet of Things, IoT*	C2

01	*Secure MQTT Authentication and Message Exchange Methods for IoT Constrained Device*	2021	*IoT, MQTT, authentication, JWT, constraint device*	C1, C2
01	*Addressing cybersecurity threats in prosumer-based nanogrids with MQTT communication*	2022	*MQTT, threats, IDPS, nanogrid, cybersecurity*	C1, C2
01	*Vulnerabilities and Security Issues in IoT Protocols*	2020	*IoT, Security ,Privacy, Protocols, MQTT, DDS, LoRaWan*	C1, C2
01	*USAF-IoD: Ultralightweight and Secure Authenticated Key Agreement Framework for Internet of Drones Environment*	2024	*Internet of Drones (IoD), key agreement, physical unclonable functions, security, user authentatio*n	CE3, CE4
01	*Modified protocol for secure mutual authentication in IoT smart homes*	2021	*Internet of Things, Keyed-hash chain, Mutual authentication, Smart home*	CE4
01	*Reverse Sequence Hash Chain based Multicast Authentication for IoT Firmware Updates*	2021	*Authentication, multicast, Internet of Thing*s	CE4
01	*A novel approach to secure IoT cryptocurrency wallets based on Enhanced ECDSA algorithm*	2024	*Cryptocurrency, Internet of Things (IoT), Blockchain, Elliptic Curve Digital Signature Algorithm (ECDSA)*	CE4
01	*Securing IoT Communication: A Steganographic Protocol for Efficient Mutual Authentication and Data Integrity*	2024	*Internet of Things (IoT), Security, Cryptography, Steganography, Mutual Authentication, MITM, Integrity*	CE1

01	*Publish Subscribe System Security Requirement: A Case Study for V2V Communication*	2024	*Internet of Things (IoT), MQTT-SN, security requirements, Vehicle-to-Vehicle (V2V) communication*	C1, C2
01	*Assessing Initial Attacks and Defense with Retry Function in MQTT Over QUIC*	2024	*MQTT, QUIC, DoS, Initial Attack, IoT*	CE2
01	*A Lightweight PUF-Based Authentication Protocol Using Secret Pattern Recognition for Constrained IoT Devices*	2021	*Authentication, hardware security, IoT security, lightweight security, physical unclonable functions*	CE3
01	*A CRC-Based Authentication Model and ECC-Based Authentication Protocol for Resource-Constrained IoT Applications*	2024	*Authentication, vulnerability, latency, security* Π*-calculus, functionalities*	CE3
01	*A Permutation Challenge Input Interface for Arbiter PUF Variants Against Machine Learning Attacks*	2022	*IoT security , physical unclonable function , machine learning attack, arbiter PUF variants*	CE4
01	*Secure protocol buffers for Bluetooth Low-Energy communication with wearable devices*	2021	*Internet of Things, Bluetooth Low-Energy, Security, Privacy, Protocol Buffers*	CE3
01	*A Novel Lightweight Group Authentication Protocol for Internet of Things*	2023	*Embedded systems, IoT, Lightweight, Group authentication*	CE3
01	*A Monitoring and Control Gateway for IoT Edge Devices in Smart Home*	2020	*IoT Edge Devices, Security , Control, Smart Home*	CE3

01	*Secure IoT Communication with a Blockchain-Enabled Decentralized MQTT Network*	2023	*MQTT, IoT, Blockchain, Decentralization, Federation, IoT Oracles*	C1, C2
01	*PUF-Based Mutual Authentication and Key Exchange Protocol for Peer-to-Peer IoT Applications*	2023	*Peer-to-peer Internet of Things, IoT security, physical unclonable functions, peer-entity authentication protocol, authenticated key exchange protocol, man-in-the-middle attacks*	CE3
01	*Securing Internet of Vehicles Protocols using ASCON and GIFT-COFB*	2023	*Internet of Vehicles (IoV), Message Queuing Telemetry Transport (MQTT), Security, Payload Encryption, Lightweight Cryptography*	C1, C2
01	*Towards Spoofing Resistant Next Generation IoT Networks*	2022	*5G–NR, deep learning, Internet of Things (IoT), IoT security, mmWave communication, physical layer security, wireless spoofing attack detection*	CE3
01	*Role of PKI in Securing AMQP Communication*	2024	*IoT, IoT Security, AMQP, PKI, MitM, DoS*	C1, C2
01	*Lightweight and Privacy-Preserving Remote User Authentication for Smart Homes*	2022	*Authentication protocol, Internet of Things, protocol resiliency, geometric secret sharing, smart home, PRNU*	CE5
01	*IB-RPL: Embedding Isolation and Blacklisting of Malicious Nodes in RPL for Securing IoT-LLNs*	2021	*RPL, IoT, Routing Attack, Isolation, Blacklisting*	CE1

01	*Incident Response in Smart Agriculture: An MQTT Case Study*	2024	*Internet of Things, Agricultural Internet of Things (Ag-IoT), Smart Agriculture, Incident Response Model, MQTT Protocol, IoT Security*	C1, C2
01	*Pirates of the MQTT: Raiding IIoT Systems with a Rogue Client*	2024	*Rogue Client, MQTT Protocol, Cyberattacks, Cybersecurity*	C1, C2
01	*Security-Bag: A Specification-based Intrusion Detection System Applied to Star Topology BLE Networks*	2024	*IoT networks, Bluetooth Low Energy (BLE), Specification-based Intrusion Detection System (IDS), Security-Bag*	CE3
01	*Securing the IoT Application Layer From an MQTT Protocol Perspective: Challenges and Research Prospects*	2024	*IoT, application layer, MQTT, vulnerabilities, attacks, taxonomy, testing, detection, defense*	C1, C2
01	*DAISS: Design of an Attacker Identification Scheme in CoAP Request/Response Spoofing*	2021	Não se aplica	C1
01	*A detailed Analysis of Various Security Issues in Wireless Communication Cyber— Physical Systems and the Internet-of-Things*	2022	5G, Cyber Physical System (CPS), Internet-of-Things(IoT), Machine-Type Communication(MTC)	CE5
01	*Maximizing IoT Security: An Examination of Cryptographic Algorithms*	2023	Não se aplica	CE3
01	*SGANFuzz: A Deep Learning-Based MQTT Fuzzing Method Using Generative Adversarial Networks*	2024	*MQTT, fuzz test, generative adversarial networks, time-series models, transformer, vulnerability detection*	C1

01	*Reliable, Secure, and Efficient Hardware Implementation of Password Manager System Using SRAM PUF*	2021	*Password security, PUF-based security, database hacking, authentication, hash chain*	CE5
01	*BlockMQ: Blockchain Assisted MQTT Authentication*	2024	*blockchain, iot, authentication, mqtt*	C1, C2
01	*TRUSTY: A Solution for Threat Hunting Using Data Analysis in Critical Infrastructures*	2021	*Cybersecurity, Dataset, Honeypot, Industrial Internet of Things, Multi-Armed Bandit, Reinforcement Learning, Thompson Sampling*	CE5
01	*Characterizing and Mining Traffic Patterns of IoT Devices in Edge Networks*	2021	*Internet-of-Things, measurement, smart home, network monitoring, anomaly traffic detection.*	CE3
01	*Lightweight and DPA-Resistant Post-Quantum Cryptoprocessor based on Binary Ring-LWE*	2020	*Post-quantum cryptography, lattice-based cryptography, ring learning with errors (Ring-LWE), internet of things (IoT), hardware implementation*	CE3
01	*Arbiter PUF—A Review of Design, Composition, and Security Aspects*	2023	*Arbiter PUF, cryptography, hardware security, machine learning attacks, physical unclonable function, ZedBoard*	CE3
01	*Securing Shared Subscriptions in MQTTv5 for IoT Networks: Vulnerability Analysis and Mitigation*	2024	*Wireless Sensor Network Security, MQTT, Shared Subscription*	CI1, CI2
01	*A Scalable Architecture for Monitoring IoT Devices Using Ethereum and Fog Computing*	2020	*Internet of things, device monitoring, fog, blockchain, smart contract, ethereum, ipfs*	CE3

01	*HALE-IoT: Hardening Legacy Internet of Things Devices by Retrofitting Defensive Firmware Modifications and Implants*	2023	C*ybersecurity, defensive techniques, devices, end–of–life (EOL), firmware, firmware modification, HTTPS, Internet of Things (IoT), legacy, retrofit security, secure socket layer (SSL)–proxy, Web application firewall (WAF)*	CE3
02	*Industrial Internet of Things Ecosystems Security and Digital Forensics: Achievements, Open Challenges, and Future Directions*	2024	*Industrial internet of things (IIoT), ecosystem, security, digital forensics*	CE3
02	*A Study of the Resilience of the Mosquitto MQTT Broker Running on Raspberry Pi Against Fuzzy DDoS Attacks*	2024	Não se aplica	CE6
02	A Survey of IIoT Protocols: A Measure of Vulnerability Risk Analysis Based on CVSS	2020	Não se aplica	CE3
02	*CPSIoTSec'24: Proceedings of the Sixth Workshop on CPS&IoT Security and Privacy*	2024	Não se aplica	CE3
02	*MEC-enabled 5G Use Cases: A Survey on Security Vulnerabilities and Countermeasures*	2021	Não se aplica	CE5
02	*Knowledge-based Cyber Physical Security at Smart Home: A Review*	2024	Não se aplica	CE3

02	*A Comprehensive Review of the State-of-the-Art on Security and Privacy Issues in Healthcare*	2023	Não se aplica	CE3
02	*A Survey of Cybersecurity Certification for the Internet of Things*	2020	Não se aplica	CE3
02	*ARES '24: Proceedings of the 19th International Conference on Availability, Reliability and Security*	2024	Não se aplica	CE5
02	*CCS '24: Proceedings of the 2024 on ACM SIGSAC Conference on Computer and Communications Security*	2024	Não se aplica	CE5
02	*ASIA CCS '24: Proceedings of the 19th ACM Asia Conference on Computer and Communications Security*	2024	Não se aplica	CE5
02	*ARES '23: Proceedings of the 18th International Conference on Availability, Reliability and Security*	2023	Não se aplica	CE5
02	*ARES '22: Proceedings of the 17th International Conference on Availability, Reliability and Security*	2022	Não se aplica	CE5
02	*CE5 ICCTA '22: Proceedings of the 2022 8th International Conference on Computer Technology Applications*	2022	Não se aplica	CE5

02	*CE5 ICNCC '23: Proceedings of the 2023 12th International Conference on Networks, Communication and Computing*	2024	Não se aplica	CE5
02	*CE5 RAID '24: Proceedings of the 27th International Symposium on Research in Attacks, Intrusions and Defenses*	2024	Não se aplica	CE5
02	CE5 ICIMMI '23: Proceedings of the 5th International Conference on Information Management & Machine Intelligence	2024	Não se aplica	CE5
02	*CE5 RAID '23: Proceedings of the 26th International Symposium on Research in Attacks, Intrusions and Defenses*	2023	Não se aplica	CE5
02	*CE5 CAIBDA '24: Proceedings of the 2024 4th International Conference on Artificial Intelligence, Big Data and Algorithms*	2024	Não se aplica	CE5
02	*CE5 LANC '24: Proceedings of the 2024 Latin America Networking Conference*	2024	Não se aplica	CE5
03	*An Open-Source Supervisory Control and Data Acquisition Architecture for Photovoltaic System Monitoring Using ESP32, Banana Pi M4, and Node-RED*	2024	*renewable energy; PV system; SCADA; Node-RED; Internet of Things;* MTU; RTU; MQTT	CE2

03	*A Comparative Study of Post-Quantum Cryptographic Algorithm Implementations for Secure and Efficient Energy Systems Monitoring*	2023	*Internet of Things; NTRU; post-quantum encryption; RSA; SABER; system monitoring*	CI1, C12
03	*SpecificationBased Symbolic Execution for Stateful Network Protocol Implementations in IoT*	2023	*Constrained networking, Internet of Things, network protocols, software testing, symbolic execution*	CE3
03	*Threat-modeling-guided Trust-based Task Offloading for Resource-constrained Internet of Things*	2022	*computation offloading edge computing; Internet of Things; resource constrained, threat modelling, Trust*	CE2, CE3
03	*SymEx-VP: An open source virtual prototype for OS-agnostic concolic testing of IoT firmware*	2022	Concolic testingVirtual prototypingSystemCRISC-VInternet of things	CE3
03	*Architecture and security of SCADA systems: A review*	2021	*Critical infrastructure; Cyber-physical systems; IDS; IIoT; SCADA attacks; SCADA systems security; Testbed*	CE3
03	*Insecurity of operational cellular IoT service: New vulnerabilities, attacks, and countermeasures*	2021	*cellular IoT; security; service charging*	CE3, CE4
03	Security challenges in adopting internet of things for smart network	2021	*Internet of Things (IoT) ,Smart Network, Security, Attacks*, IDS	CE3

03	*Vulnerability Analysis and Method of Selection of Communication Protocols for Information Transfer in Internet of Things Systems*	2021	*application layer protocols; communication templates; cyberattacks; Internet of Things; models of messaging*	CE6
03	*Hybrid logical security framework for privacy preservation in the green internet of things*	2020	*green IoT; ICT; authentication; confidentiality; cryptography; security framework*	CE2, CE3
03	*Updating IoT devices: Challenges and potential approaches*	2020	*Software updates, Internet of Things, Security*	CE3
04	*Hybrid Logical Security Framework for Privacy Preservation in the Green Internet of Things*	2020	*green IoT; ICT; authentication; confidentiality; cryptography; security framework*	CE2

Fonte: Elaborada pela autora.