ZETIn: Infraestrutura baseada em Zero Trust para Segurança e Conectividade em Redes em Malha Ad-Hoc

Guilherme Nunes Nasseh Barbosa[1] **e Diogo Menezes Ferrazani Mattos**[1]

[1] LabGen/MídiaCom – TET/IC/PPGEET/UFF
Universidade Federal Fluminense (UFF)
Niterói, RJ – Brasil

Abstract. *The increasing complexity of cyber threats and the dynamic nature of mesh networks demand adaptive security solutions to ensure resilience, secure connectivity, and automated service configuration. Traditional security approaches struggle with the constantly changing topology and ad-hoc nature of these networks, making Zero Trust a promising paradigm for enhancing security while maintaining flexibility. This paper proposes ZETIn, a Zero Trust Infrastructure that leverages Generative AI and LLM-based chatbots to configure security services in the Cloud Continuum, integrating a consensus protocol to enable seamless communication across mesh networks while enforcing strict security policies. Compared to previous approaches, our experimental results show that the Zero Trust architecture introduces minimal overhead while significantly improving connectivity, security, and automated service orchestration, making it a viable solution for mission-critical ad-hoc mesh networks.*

Resumo. *O aumento da complexidade das ameaças cibernéticas e a natureza dinâmica das redes mesh exigem soluções de segurança adaptativas para garantir resiliência, conectividade segura e configuração automatizada de serviços. As abordagens tradicionais de segurança enfrentam dificuldades com a topologia em constante mudança e a natureza ad-hoc dessas redes, tornando o conceito de Confiança Zero um paradigma promissor para reforçar a segurança sem comprometer a flexibilidade. Este artigo propõe o ZETIn, uma Infraestrutura de Confiança Zero que utiliza IA Generativa e chatbots baseados em LLMs para configurar serviços de segurança no Cloud Continuum, incorporando um protocolo de consenso para viabilizar a comunicação fluida em redes mesh, mantendo políticas de segurança rigorosas. Em comparação com abordagens anteriores, nossos resultados experimentais demonstram que a arquitetura de Confiança Zero introduz um overhead mínimo, ao mesmo tempo que melhora significativamente a conectividade, a segurança e a orquestração automatizada de serviços, tornando-se uma solução viável para redes mesh ad-hoc de missão crítica.*

1. Introdução

O cenário de ameaças cibernéticas apresenta um crescimento expressivo, impulsionado pelo modelo *Ransomware-as-a-Service* (RaaS) e pela sofisticação das técnicas

Este trabalho foi financiado pelo CNPq, CAPES, FAPERJ, RNP, Prefeitura Municipal de Niterói/FEC/UFF (Edital PDPA 2020) e INCT ICONIOT. Ferramentas de Inteligência Artificial Generativa, incluindo ChatGPT, Grammarly e Llama3.1, foram empregadas na revisão textual deste trabalho.

de ataque, como o roubo de credenciais. A Symantec reporta que entre 2022 e 2023, houve um aumento significativo nos ataques de *ransomware*, com picos significativos em setembro e outubro de 2023[1]. Em razão do aumento dos ataques cibernéticos, o *National Institute of Standards and Technology* (NIST) desenvolveu a *Zero Trust Architecture* (ZTA) [Sheikh et al., 2021, Zivi e Doerr, 2022]. A ZTA busca estabelecer um modelo de segurança que minimiza a confiança implícita, exigindo autenticação contínua, controle de acesso rigoroso e microsegmentação da rede por meio do conceito de *Software Defined Perimeter* (SDP) [Syed et al., 2022]. No entanto, a implementação da *Zero Trust* em ambientes de rede dinâmicos e descentralizados ainda representa um desafio, especialmente em cenários em que modelos tradicionais baseados em perímetro se mostram ineficazes.

Paralelamente, a rápida evolução de sistemas autônomos de inteligência artificial (IA), a importância de práticas de segurança centradas em pessoas e o papel de tecnologias que aumentam a produtividade dos desenvolvedores são apontadas como tendências atuais pela Gartner[2]. Nesse cenário, soluções que aliem redes em malha sem fio (*mesh networks*), IA Generativa e princípios de *Zero Trust* surgem como altamente promissoras para ambientes dinâmicos e críticos. Enquanto a IA Generativa ultrapassa o pico de expectativas infladas e começa a demonstrar retorno de investimento em aplicações reais, estratégias de segurança baseadas em *Zero Trust* e na abordagem de segurança e privacidade centrado no ser usuário tornam-se fundamentais para enfrentar ameaças cada vez mais complexas. As redes em malha sem fio, por sua vez, oferecem resiliência e flexibilidade que potencializam a adaptabilidade de sistemas autônomos e distribuídos.

Contudo, um dos principais desafios das arquiteturas de redes de próxima geração refere-se às redes em malha sem fio para aplicações críticas, amplamente utilizadas em sistemas autônomos, operações militares e monitoramento ambiental. Essas redes operam em ambientes altamente dinâmicos, em que os nós frequentemente mudam de estado e a conectividade é imprevisível. Garantir comunicação segura, aplicação e adaptação de políticas em tempo real nessas condições é complexo, pois os modelos de segurança tradicionais apresentam dificuldades para fornecer resiliência e autenticação de baixa latência em topologias em constante evolução.

Este artigo propõe o *ZETIn*, uma plataforma *Zero Trust* que utiliza IA Generativa para configurar serviços de segurança de rede com base em comandos em linguagem natural. O sistema conta com um *chatbot* interativo, integrado a modelos de linguagem de larga escala (*Large Language Models* - LLMs), permitindo que operadores descrevam configurações de segurança por meio de comandos intuitivos. Essas entradas são então traduzidas em políticas de segurança aplicáveis dinamicamente à rede em malha sem fio (*Wireless Ad-hoc Mesh Networks*). A plataforma é construída sobre o arcabouço *OpenZiti Zero Trust*, garantindo criptografia de ponta a ponta, autenticação robusta e gerenciamento automatizado de políticas de segurança entre dispositivos.

Trabalhos anteriores exploraram implementações de *Zero Trust* em redes tradicionais, mas frequentemente dependem de configurações estáticas ou intervenções manuais,

[1]Disponível em `https://www.symantec.broadcom.com/hubfs/Symantec_Ransomware_Threat_Landscape_2024.pdf`.

[2]Disponível em `https://www.gartner.com/en/newsroom/press-releases/2024-08-21-gartner-2024-hype-cycle-for-emerging-technologies-highlights-developer-productivity-total-experience-ai-and-security`.

limitando sua adaptabilidade a ambientes dinâmicos. Em contraste, a abordagem proposta automatiza a aplicação da segurança e reduz o erro humano ao empregar configuração de políticas baseada em IA. Os resultados experimentais mostram que o *ZETIn* introduz um impacto mínimo no desempenho e aprimora a eficiência da aplicação de segurança. Esses resultados destacam a adequação da abordagem proposta para otimizar a segurança e a comunicação em ambientes operacionais autônomos e de alto risco.

O restante do artigo está organizado da seguinte forma: A Seção 2 elenca os trabalhos relacionados. Os principais componentes da *Zero Trust Architecture* são apresentados na Seção 3. O problema de estabelecimento de comunicação segura e confiável em redes em malha sem fio é analisado na Seção 4. A arquitetura de rede em malha sem fio é proposta na Seção 5. A Seção 6 avalia a proposta e discute os resultados obtidos. A Seção 7 conclui o trabalho.

2. Trabalhos Relacionados

Trabalhos anteriores utilizam os conceitos da *Zero Trust Architecture* para abordar desafios específicos em empresas e grandes organizações. A proposta deste trabalho se destaca por integrar uma plataforma *Zero Trust* com IA generativa para configurar dinamicamente serviços em redes em malha sem fio, dedicadas a missões críticas. Em um trabalho anterior, Barbosa e Mattos apresentam a arquitetura inicial da plataforma *ZETIn* e realizam uma demonstração de prova de conceito [Barbosa e Mattos, 2024]. Contudo, no trabalho inicial, não há a definição do roteamento sobre a rede em malha sem fio e a plataforma não é avaliada nem tem seu desempenho comparado entre casos de uso.

Al-Hammuri *et al.* propõem um sistema de pontuação baseado em *Zero Trust* para evitar erros médicos em sistemas de informação em saúde baseados em nuvem, utilizando aprendizado de máquina e autenticação baseada em microsserviços [Al-hammuri et al., 2024]. A abordagem proposta neste artigo, por outro lado, foca na configuração e gestão automática da rede, utilizando *OpenZiti* e *Ollama* para implementar uma solução segura e flexível que processa comandos em linguagem natural, facilitando a configuração automatizada e segura de redes e serviços. Kroculick explora metodologias teóricas para o desenvolvimento de arquiteturas *Zero Trust* [Kroculick, 2024]. Em contrapartida, a proposta ZETIn adota uma abordagem mais prática, incorporando um chatbot baseado em LLMs que permite que operadores com menor conhecimento técnico configurem dinamicamente redes complexas, tornando a solução mais acessível e aplicável em cenários reais. Tanimoto *et al.* abordam a escalabilidade de perímetros definidos por software (SDP) em diferentes organizações, propondo modelos como hierárquico e de ponte [Tanimoto et al., 2023]. Em contraste, a proposta ZETIn combina SDP com IA generativa para não apenas escalar, mas também adaptar automaticamente a configuração da rede de acordo com mudanças no ambiente.

Chandramouli e Butcher investigam o controle de acesso em aplicações nativas de nuvem utilizando *service mesh* e infraestrutura baseada em proxy [Chandramouli e Butcher, 2023]. A proposta deste artigo, por outro lado, integra a detecção automatizada de serviços e sua configuração por meio de interações em linguagem natural, aprimorando a usabilidade e a segurança em redes em malha sem fio. Outros trabalhos propõem soluções específicas para avaliação de confiança e gerenciamento de redes [Khowaja et al., 2024, Alboqmi et al., 2023], a proposta deste

artigo é mais abrangente. A proposta ZETIn oferece uma solução holística que não apenas avalia a confiança, mas também configura e gerencia dinamicamente a rede de forma segura. Utilizando *OpenZiti* e IA generativa, a proposta garante uma infraestrutura resiliente e adaptável para diversos cenários críticos.

Além disso, Al-Shareeda e Manickam discutem os desafios das redes *ad hoc*, particularmente as *Vehicular Ad-Hoc Networks* (VANETs) [Al-Shareeda e Manickam, 2023]. Essas redes enfrentam ameaças críticas, como ataques *Sybil*, onde múltiplas identidades falsas podem comprometer a integridade das comunicações, e ataques de *Man-in-the-Middle*, que permitem interceptação e alteração de mensagens entre veículos. Ataques de *Distributed Denial of Service* (DDoS) também representam um grande risco, sobrecarregando a rede e reduzindo a qualidade do serviço. Além disso, a preservação da privacidade exige esquemas robustos de autenticação que protejam as identidades dos usuários sem comprometer a segurança. Zabeeulla *et al.* destacam a crescente necessidade de segurança em VANETs devido à proliferação de veículos autônomos e conectados [Zabeeulla et al., 2023]. Os autores introduzem uma abordagem híbrida de segurança que integra diferentes técnicas para detectar e mitigar ataques, principalmente DDoS. Kashyap *et.al* [Kashyap et al., 2019] propõem uma rede em malha para ambientes de IoT em casas e cidades inteligentes, em que dispositivos ativos e passivos atuam como nós, melhorando a cobertura e reduzindo zonas de sombra. A rede é auto configurável, o que melhora a escalabilidade e robustez sem necessidade de infraestrutura adicional. No entanto, o sistema enfrenta desafios como interferência, complexidade de gerenciamento, latência e consumo de energia elevado. Além disso, o artigo não aborda questões relativas à segurança, o que deixa a rede potencialmente vulnerável a ataques, sobretudo para uso em cidades inteligentes.

As pesquisas anteriores investigaram desafios e soluções para redes *ad hoc*, incluindo segurança em VANETs e adaptação da comunicação em redes móveis críticas. No entanto, a proposta deste artigo apresenta uma abordagem mais abrangente e robusta ao integrar uma rede em malha sem fio com uma arquitetura *Zero Trust* baseada em IA generativa. Essa combinação garante maior consistência, adaptabilidade e segurança para sistemas autônomos em operações críticas.

3. Principais componentes da *Zero Trust Architecture*

O modelo de confiança zero (*Zero Trust*) é estruturado em diversos componentes lógicos, projetados para garantir que nenhuma entidade, seja interna ou externa, seja confiável por padrão. A interação entre esses componentes ocorre em um plano de controle dedicado, enquanto os dados das aplicações trafegam separadamente no plano de dados. Entre os principais componentes destaca-se o *Policy Decision Point* (PDP), que é subdividido em dois elementos lógicos: o *Policy Engine (PE)* e o *Policy Administrator (PA)*. O *Policy Engine* é responsável por avaliar as políticas e determinar as decisões de acesso com base em atributos como identidade, contexto e regras predefinidas. Já o *Policy Administrator* implementa essas decisões, gerenciando as configurações de controle de acesso e aplicando as políticas nos componentes do plano de dados. Essa separação clara entre os planos de controle e de dados é essencial para garantir um gerenciamento seguro e eficiente dos recursos [Stafford, 2020]. Outro componente importante é o *Trust Algorithm (TA)*, sendo esse utilizado para conceder ou negar acesso a um ativo. O TA recebe entradas de várias fontes, incluindo banco de dados de políticas, informações observáveis sobre

os sujeitos, padrões históricos de comportamento dos usuários, fontes de inteligência de ameaças e outros metadados. Essas entradas são categorizadas para avaliar a solicitação de acesso com base em critérios pré-definidos ou uma pontuação de confiança. Existem variações no TA, como a avaliação baseada em critérios versus a baseada em pontuação e a consideração singular versus contextual das solicitações[Stafford, 2020].

Além dos componentes principais dos planos de dados e controle, outras fontes de dados podem ser utilizadas para fornecer regras de entrada e tomar decisões de acesso[Abdalla et al., 2024], tais como:

- **Sistema contínuo de diagnóstico e mitigação - CDM**. Este subsistema desempenha um papel fundamental ao coletar informações atualizadas sobre o estado atual dos ativos. Sua principal função é implementar atualizações críticas em configurações e componentes de *software*, avaliando se os ativos estão executando componentes adequadamente, verificando a integridade dos componentes de *software*, e identificar quaisquer componentes não autorizados, sinalizando possíveis vulnerabilidades.
- ***Threat intelligence feed(s)***. Fornecem informações valiosas de diversas fontes, tanto internas quanto externas, auxiliando o PDP de confiança zero a tomar decisões de acesso informadas. Esses feeds atualizam constantemente o PDP sobre ameaças emergentes, vulnerabilidades recém-descobertas, relatórios de malware e ataques recentes a outros ativos. Ao aproveitar essa inteligência, o PDP pode identificar rapidamente riscos potenciais e impedir o acesso de fontes suspeitas.
- **Logs de rede e de atividades**. Atuam como um repositório de dados, coletando informações detalhadas sobre o tráfego de rede, acessos e eventos do sistema. Esses dados, atualizados em tempo quase real, alimentam sistemas de detecção e resposta a incidentes, permitindo a identificação rápida de ameaças e a tomada de decisões para a proteção da rede.

Com a integração de diversas fontes de dados, o modelo de confiança zero impulsiona, de forma acelerada, o desenvolvimento de soluções tanto comerciais quanto *open source* para a proteção de ambientes em cenários cada vez mais dinâmicos e complexos. Soluções comerciais de grandes fornecedores, como Zscaler, Palo Alto Networks, Fortinet e Microsoft, se destacam ao oferecer serviços empresariais robustos e integrados. Paralelamente, projetos de código aberto, como OpenZiti e HashiCorp Boundary, ganham destaque pela flexibilidade e pela ampla possibilidade de customização, atendendo a demandas específicas e promovendo inovações colaborativas no campo da segurança.

4. Conciliação da Arquitetura de Confiança Zero e Redes em Malha Sem Fio

A plataforma ZETIn baseia-se em soluções de código aberto para implementar a arquitetura de confiança zero (*Zero Trust Architecture* - ZTA), a rede em malha (*mesh network*) e o ambiente de Inteligência Artificial. O OpenZiti[3] provê redes de sobreposição seguras, implementando ZTA por meio de um Ziti Controller que gerencia o perímetro definido por software (*Software Defined Perimeter* - SDP) e de Roteadores de Borda que atuam como *Gateways* SDP. O OpenZiti garante controle de acesso rigoroso, otimização do tráfego de rede e escalabilidade, facilitando a expansão organizacional e a segurança robusta de IoT [Teerakanok et al., 2021, Diaz Rivera et al., 2022].

[3]Disponível em `https://openziti.io/`.

4.1. Arcabouço de Confiança Zero OpenZiti

OpenZiti é uma plataforma de código aberto desenvolvida para implementar uma rede baseada na arquitetura de confiança zero [Teerakanok et al., 2021, Diaz Rivera et al., 2022]. A plataforma fornece redes seguras sobrepostas à rede física (*overlay*). A plataforma oferece uma solução de rede definida por software que permite a criação de redes seguras, invisíveis e segmentadas, garantindo que os recursos só sejam acessíveis por usuários e dispositivos autenticados e autorizados. Composta pelo Controlador Ziti, o Edge Router e ferramentas para o desenvolvimento de aplicações, a plataforma gerencia políticas de segurança, facilita a comunicação segura e permite a integração das funcionalidades de segurança nas aplicações, tornando a implementação da ZTA adaptável às necessidades das organizações e mais flexível.

O Controlador Ziti (*Ziti Controller*) é responsável por ser o controlador do Perímetros Definido por *Software*, enquanto o *Edge Router* atua como *Gateway* do SDP. Desenvolvido em Golang, OpenZiti emprega três tipos de políticas: *service*, *service edge router* e *edge router*. Definir uma *service police* envolve as etapas: *bind* e *dial*. A etapa Bind identifica a entidade de serviço, enquanto a etapa Dial especifica os clientes autorizados a acessar esses serviços [Diaz Rivera et al., 2022]. As funcionalidades principais do OpenZiti incluem controle de acesso rigoroso, otimização do tráfego de rede e escalabilidade para suportar a expansão organizacional. A plataforma aplica políticas de autenticação e autorização de forma contínua, melhorando a eficiência e a segurança da infraestrutura de TI. O OpenZiti é versátil, permitindo a integração de segurança em aplicações legadas, proteção de recursos em ambientes de nuvem e oferecendo robusta segurança para dispositivos IoT. Com essa abordagem, o OpenZiti possibilita a construção de redes seguras e resistentes a ataques, protegendo continuamente os recursos críticos da organização.

4.2. Rede em Malha B.A.T.M.A.N.

A implementação da rede em malha utiliza o B.A.T.M.A.N.[4] (*Better Approach to Mobile Ad-hoc Networking*), protocolo de roteamento com comportamento proativo para redes sem fio em malha (*Wireless Ad-hoc Mesh Networks*). O B.A.T.M.A.N. mantém informações sobre nós acessíveis, determinando o melhor vizinho de um salto para cada destino, facilitando o roteamento multi-saltos[5]. Mensagens de Originadores (*Originator Messages* - OGMs) são enviadas e reencaminhadas periodicamente para que todos os nós as recebam, permitindo estimar a qualidade das rotas por meio do número de OGMs recebidos.

As redes em malha, baseadas no protocolo B.A.T.M.A.N., formam arquiteturas de rede descentralizadas que aprimoram a comunicação em redes sem fio. Diferentemente de redes tradicionais, que dependem de um único ponto central de falha, as redes em malha distribuem dados por múltiplos nós, criando um sistema resiliente em que a informação pode encontrar vários caminhos para chegar ao destino. O protocolo B.A.T.M.A.N. exerce papel fundamental ao gerenciar dinamicamente o roteamento de dados, de maneira eficiente, especialmente em ambientes móveis e não infraestruturadas. O protocolo avalia

[4]Disponível em `https://www.open-mesh.org/projects/open-mesh/wiki`.

[5]A proposta deste artigo é agnóstica em relação à versão do B.A.T.M.A.N. utilizada, desde que haja conectividade IP entre os nós da rede.

continuamente as melhores rotas para transmissão de dados, garantindo que a rede se adapte a mudanças de topologia, como a adição ou remoção de nós.

Em uma rede em malha B.A.T.M.A.N., cada nó atua tanto como cliente quanto como roteador, encaminhando dados para seus vizinhos. Essa capacidade de autorrecuperação torna as redes em malha ideais para cenários em que a infraestrutura pode ser instável ou inexistente, como em situações de recuperação de desastres, áreas rurais ou grandes eventos. O B.A.T.M.A.N. reduz a complexidade do roteamento ao não depender de um conhecimento global da rede, concentrando-se na visão local de cada nó. Essa abordagem minimiza a sobrecarga e permite que a rede seja escalável de maneira mais eficaz, proporcionando uma comunicação robusta e eficiente mesmo com o crescimento do número de nós.

4.3. Arcabouço para Grandes Modelos de Linguagem

A implementação de Inteligência Artificial (IA)[6] usa o Open WebUI[7] e o Ollama[8], criando um ambiente amigável e seguro. O Open WebUI é uma interface web autohospedada que suporta diversos motores de LLM, incluindo Ollama e APIs compatíveis com o OpenAI. O Ollama, por sua vez, aprimora as funcionalidades de IA ao processar comandos em linguagem natural para configurações dinâmicas da rede. Essa integração fornece uma solução adaptável para gerenciar serviços de rede em ambientes de missão crítica. Contudo, um aspecto importante ao utilizar grandes modelos de linguagem é a segurança. Códigos gerados automaticamente por IA generativa podem apresentar vulnerabilidades específicas, que nem sempre coincidem com as encontradas em códigos produzidos por desenvolvedores humanos, exigindo práticas de validação e testes adicionais para garantir níveis adequados de segurança [Hamer et al., 2024].

4.4. Protocolo de Consenso Raft

O Raft é um mecanismo de consenso baseado no modelo líder-seguidor, garantindo tolerância a falhas de nós (*Crash-Failure Fault* - CFT). Seu funcionamento é dividido em três fases: eleição de líder, replicação de logs e segurança. Inicialmente, os nós são seguidores, mas caso não detectem um líder, iniciam uma eleição, em que candidatos coletam votos via RPC *RequestVotes* até que a maioria simples escolha um novo líder. O líder eleito gerencia as interações com clientes, recebendo e replicando entradas de log para os seguidores via *RPC AppendEntries*, garantindo a sincronização do estado da rede. Além disso, ele mantém a integridade dos seguidores por meio de números de épocas, impedindo nós desatualizados de aceitar requisições inválidas. A segurança do Raft impede que um seguidor desatualizado seja eleito líder, evitando inconsistências nos logs. Suas vantagens incluem implementação simples, eleições eficientes e justiça na escolha de líderes, mas exige alto armazenamento e não tolera falhas bizantinas.

5. Arquitetura de Redes em Malha Sem Fio com Confiança Zero Proposta

A proposta da arquitetura ZETIn consiste em integrar o protocolo Raft em redes em malha para sincronizar aplicações, como rotas calculadas localmente, criando um

[6]Este trabalho não avalia o desempenho de modelos de IA generativa.
[7]Disponível em `https://openwebui.com/`.
[8]Disponível em `https://ollama.com/`.

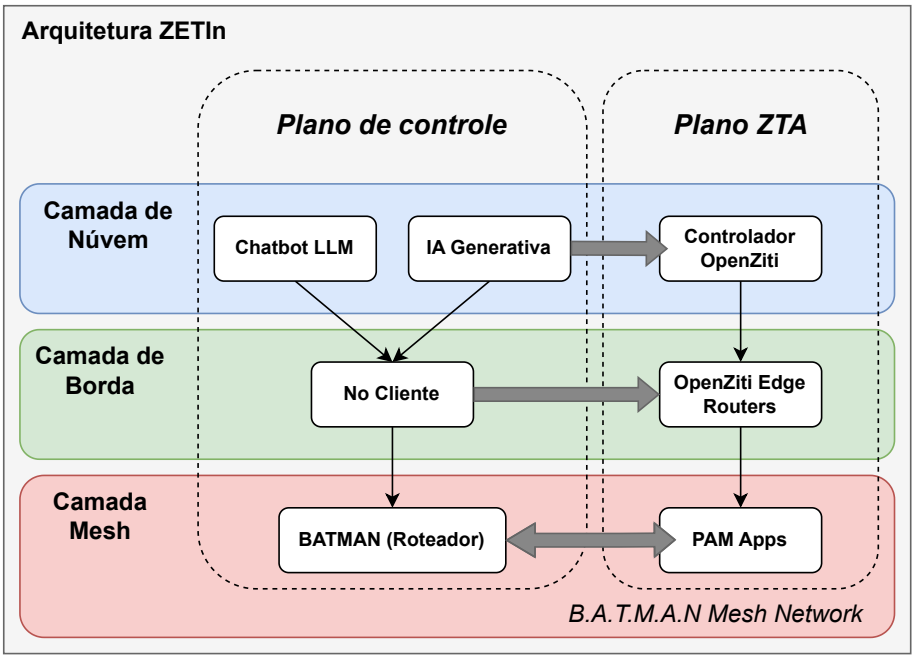


Figura 1. Arquitetura ZETIn para gerenciar redes em malha em ambientes de missão crítica, utilizando IA Generativa e *Zero Trust*. A camada de Nuvem inclui um controlador OpenZiti, um motor de IA Generativa e um *chatbot* baseado em LLM para interação com o usuário. A camada de Borda possui nós *fog* e roteadores OpenZiti, aplicando políticas de segurança. A rede em malha emprega B.A.T.M.A.N. para comunicação, protegida pelos princípios de Zero Trust. Os usuários interagem via *chatbot*, que configura dinamicamente a rede e os serviços, garantindo comunicação criptografada e segurança contínua.

ambiente de alta disponibilidade para sistemas autônomos em operações críticas. Ao aproveitar a natureza descentralizada das redes em malha sem fio, a arquitetura adapta-se dinamicamente a mudanças e garante continuidade de operação mesmo em caso de falhas de nós. O Raft gerencia o a replicação de estados, enquanto a arquitetura ZETIn lida com a comunicação entre os nós, permitindo a sincronização em tempo real das aplicações.

A consistência em toda a rede é garantida pelo algoritmo de consenso Raft que assegura a distribuição uniforme dos dados de configuração e dos estados das aplicações. Assim, a solução proposta oferece uma infraestrutura resiliente e escalável que suporta sistemas autônomos dinâmicos em ambientes nos quais redes estáticas tradicionais não são viáveis. Por meio dessa abordagem, busca-se avançar os sistemas de alta disponibilidade para aplicações críticas em redes ad-hoc sem fio. Cabe destacar que embora o protocolo Raft seja baseado em rodadas e enfrente desafios para redes altamente dinâmicas, a proposta não requer um consenso absoluto e imediato sobre a topologia. Um consenso eventual é suficiente para garantir que os pacotes sejam corretamente encaminhados mesmo enquanto as tabelas de roteamento estão convergindo.

A Figura 1 ilustra a arquitetura ZETIn, oferecendo uma solução robusta, segura e dinâmica para configurar e gerenciar redes em malha em ambientes de missão crítica, a partir de princípios de Inteligência Artificial (IA) Generativa e *Zero Trust*. A arquitetura é dividida em dois planos principais: o Plano de Controle e o Plano ZTA. Além disso,

há um plano de encaminhamento, não exibido na figura, responsável pela transmissão efetiva de dados. Para implementar o componente de *Zero Trust*, foi utilizado o arcabouço OpenZiti. Para implementação do *Chatbot* e IA Generativa, foram usados o OpenwebUI, como interface gráfica e o Ollama como servidor para execução os modelos LLM.

Na camada de Nuvem do Plano de Controle, o controlador OpenZiti gerencia a política de segurança *Zero Trust*, garantindo autenticação e autorização contínuas de todos os acessos. O motor de IA Generativa processa comandos em linguagem natural, permitindo a configuração dinâmica de serviços de rede. O Ollama realiza o motor de IA Generativa. A proposta visa aproveitar modelos como LLaMA 3.1, Gemma2 e DeepSeek-R1, proporcionando alta adaptabilidade e capacidade de resposta na rede. Além disso, um *chatbot* baseado em LLM fornece uma interface para interação com o usuário, possibilitando a emissão de comandos e consultas em linguagem natural, que são então interpretados e executados pelo motor de IA. Esse *chatbot* em LLM é implantado com OpenWebUI.

A **camada de Borda**, parte crucial do Plano de Controle, inclui nós de borda que agregam dados de dispositivos IoT e servem como intermediários entre a Nuvem e a borda da rede. Esses nós fornecem capacidades de processamento local e aumentam a escalabilidade e a capacidade de resposta da rede. Igualmente importantes nesta camada são os roteadores de borda OpenZiti, que desempenham um papel fundamental na segurança da rede. Eles aplicam políticas de segurança, garantindo que toda comunicação seja criptografada e que apenas dispositivos autenticados e autorizados possam acessar os recursos da rede. Esta medida de segurança robusta aumenta significativamente a segurança da rede.

No Plano ZTA, a rede em malha utiliza *Batman-adv* para comunicação eficiente entre dispositivos. A segurança na rede em malha é assegurada pelos princípios de *Zero Trust*, definidos pelo controlador OpenZiti e aplicados pelos roteadores de borda. Os usuários interagem com o sistema através da interface do *chatbot*, executada na Nuvem, que interpreta comandos em linguagem natural para configurar dinamicamente a rede e seus serviços. Essa interação permite ajustes em tempo real na gestão da rede, reforçando sua adaptabilidade e robustez. A aplicação contínua de políticas de segurança e a criptografia dos canais de comunicação asseguram que a rede permaneça protegida contra acessos não autorizados e ameaças em potencial.

A **camada de *mesh*** na arquitetura ZETIn desempenha um papel essencial na distribuição das relações de vizinhança e na garantia de conectividade segura para aplicações em um ambiente de *Zero Trust*. Para isso, ela utiliza o roteador B.A.T.M.A.N. que estabelece e mantém conexões dinâmicas entre os nós da rede em malha. A distribuição dessas relações de vizinhança é gerenciada através do protocolo de consenso Raft, garantindo consistência e resiliência na comunicação entre os nós, mesmo diante de falhas. Além disso, a camada *mesh* é responsável por prover conectividade segura para as *PAM Apps*, que são as aplicações que utilizam a infraestrutura de *Zero Trust* criada pelo OpenZiti para garantir sua segurança. Isso assegura que todas as comunicações dentro da rede sejam criptografadas e autenticadas, mitigando riscos de ataques e acessos não autorizados. Assim, essa camada permite a formação de uma rede robusta, adaptável e segura, essencial para aplicações em ambientes críticos.

Ao combinar IA Generativa com uma abordagem *Zero Trust*, a arquitetura apre-

senta uma solução completa e confiável para redes em malha voltadas a ambientes de missão crítica. Ela proporciona alta disponibilidade por meio do algoritmo de consenso Raft e garante segurança e escalabilidade ao explorar a natureza descentralizada das redes ad-hoc. Esse conjunto de técnicas reforça a confiança na capacidade do sistema em atender às exigências de cenários críticos e em evolução constante [Teerakanok et al., 2021, Diaz Rivera et al., 2022].

6. Avaliação e Resultados

A proposta foi avaliada utilizando oito dispositivos Raspberry Pi 4, com 2GB de RAM e 4 núcleos, organizados em uma rede em malha sem fio baseada no protocolo B.A.T.M.A.N. Esse protocolo foi escolhido por sua baixa sobrecarga em comparação com outras alternativas [Zheng et al., 2022]. Três desses dispositivos foram configurados como *gateways* de rede, responsáveis por fornecer a rota padrão e o acesso à Internet, enquanto os cinco restantes atuam como clientes, podendo hospedar diferentes tipos de aplicações. Cada um dos cinco nós possui um túnel OpenZiti configurado, que estabelece comunicação com o controlador OpenZiti para obter políticas específicas de *Zero Trust*. O ambiente de testes conta ainda com a instanciação de dois *Edge Routers* instanciados em nuvens públicas, uma localizada no Brasil e outra localizada nos Estados Unidos da América.

No controlador OpenZiti, foram definidas duas políticas de *Edge Router*, uma para o Brasil e outra Estados Unidos. Dessa forma, os cinco nós podem se conectar tanto ao *Edge Router* localizado no Brasil quanto nos Estados Unidos. Além disso, todos os nós implementam o mecanismo de consenso Raft para obtenção de endereços IP e distribuição dinâmica da topologia de rede. Periodicamente, cada nó avalia a proximidade de outros nós e, ao ingressar em um *cluster*[9], o novo nó atualiza sua lista de vizinhos e propaga essa informação para toda a rede através do protocolo Raft. Todo o código foi desenvolvido em Python.

Durante a avaliação, foram mensurados a vazão (*throughput*) e a latência em três cenários distintos: (i) com os cinco nós conectados ao *Edge Router* do Brasil, (ii) conectados ao *Edge Router* dos Estados Unidos e (iii) comunicando-se apenas pela rede local, sem o uso do OpenZiti. Além dessas métricas de rede, também foram monitorados o consumo de CPU e memória dos dispositivos nos testes realizados exclusivamente na rede local, a fim de avaliar o impacto do encaminhamento de dados sobre os recursos computacionais de cada dispositivo. As políticas de *Zero Trust*, armazenadas no controlador OpenZiti, são configuradas para cada dispositivo por meio de um chatbot baseado em um modelo de LLM, especificamente o LLaMA 3.1. O chatbot é executado utilizando a plataformas OpenwebUI e Ollama em um *cluster* Kubernetes equipado com seis GPUs NVIDIA GeForce RTX 4060 Ti, cada uma com 16 GB de memória GDDR5. Para o teste de vazão (*throughput*), foi utilizado o iperf3 [10] e para a latência, o hping3 na porta 80.

A partir da Figura 3(a) e 3(b), referentes ao uso de CPU e memória respectivamente, foi possível observar que todos os nós mantiveram um nível estável de consumo de recursos computacionais. No caso do uso de CPU, os valores médios ficaram aproximadamente entre 25% e 30%. Já o uso de memória permaneceu abaixo de 14%, su-

[9]Refere-se a um agrupamento de nós em uma rede B.A.T.M.A.N..

[10]Disponível em `https://iperf.fr/`.

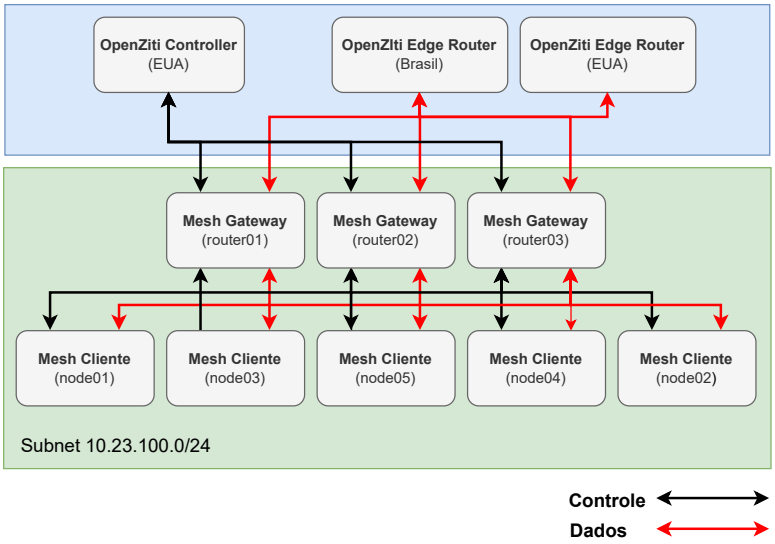
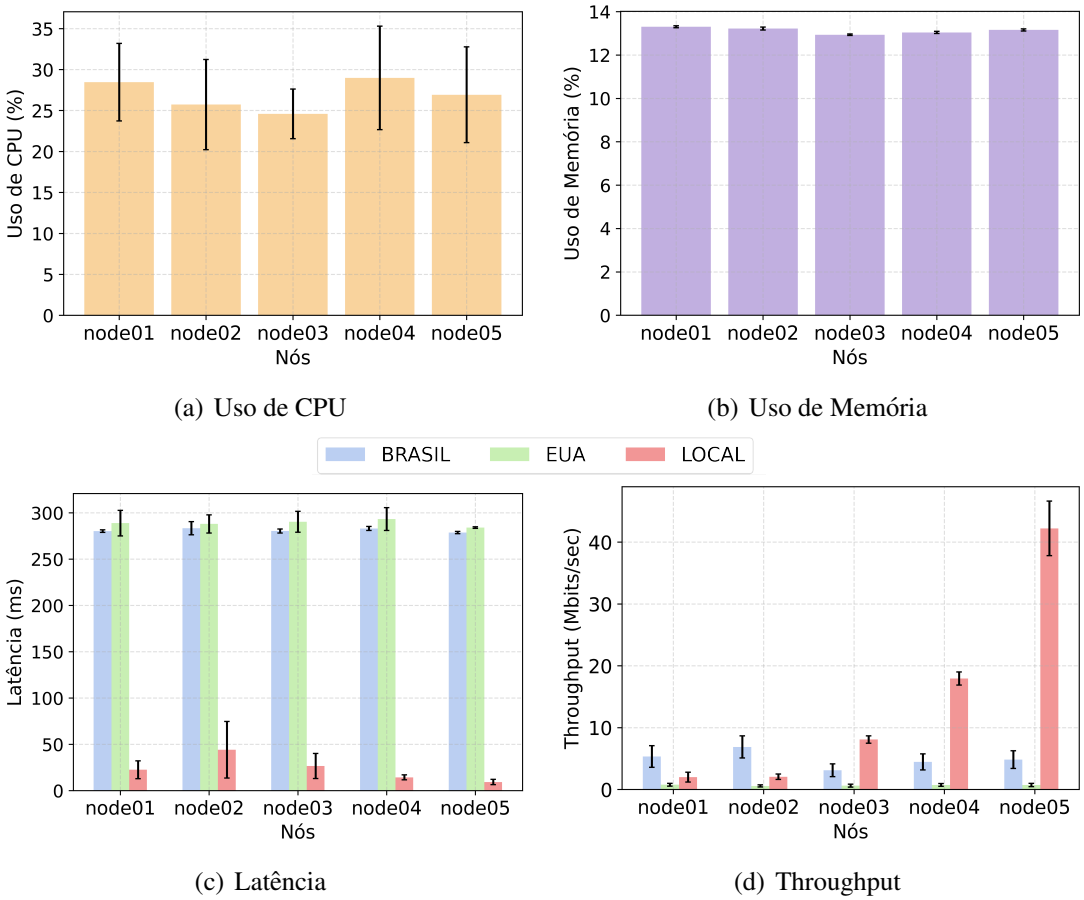


Figura 2. Topologia utilizada para validação da proposta composta por duas camadas principais: a camada de controle OpenZiti e a camada da rede *mesh* local. Na camada de controle (azul), o OpenZiti Controller, responsável pelo gerenciamento centralizado, e os OpenZiti *Edge Routers*, que são responsáveis pelo tráfego de dados do ambiente *Zero Trust* da rede em malha. Na camada mesh (verde), encontram-se três Mesh Gateways (router01, router02 e router03), que atuam como nós centrais, e cinco Mesh Clientes (node01, node02, node03, node04 e node05), conectados aos *gateways*. Os nós clientes se conectam à Internet através do *Mesh Gateways*, para obter as políticas do OpenZiti Controller através do tráfego de controle. Com as políticas obtidas se conectam em pelo menos um *Edge Router*, através do plano de dados.

gerindo que a rede e os protocolos de consenso não tendem a consumir muitos recursos desses dispositivos, indicando uma certa estabilidade, indispensável para redes em malha sem fio e dispositivos IoT com recursos limitados. Em diversos casos, os dispositivos IoT não garantem uma segurança adequada justamente em função da limitação desses recursos computacionais. No entanto, ao implementar um mecanismo de *Zero Trust*, essa complexidade pode ser transferida para a rede.

Na avaliação da latência, a Figura 3(c) exibe os diferentes cenários conforme a localização dos OpenZiti *Edge Routers*. Verifica-se que a latência ao utilizar roteadores de borda do Brasil (em verde) e dos EUA (em amarelo) permaneceu na faixa de 250 a 300 ms para todos os nós, com variações relativamente pequenas. Esse comportamento sugere que, ao utilizar um túnel via OpenZiti, a latência tende a não oscilar de forma significativa em função da localização geográfica do *Edge Router*, indicando uma consistência nos valores medidos entre as duas regiões. Nesse caso, ambos os roteadores de borda podem ser utilizados para aplicações que não sejam sensíveis à latência. Como de se esperar, a latência ao não utilizar o OpenZiti foi substancialmente menor, com valores majoritariamente abaixo de 50 ms. Ressalta-se que, como toda a comunicação no OpenZiti deve ser intermediada pelos *Edge Routers*, a latência dentro da rede OpenZiti tende a ser no mínimo o dobro da latência local para o RTT medido. Os testes foram realizados a partir de um nó *gateway* da rede. Os nós que não estavam conectados diretamente a este



(a) Uso de CPU (b) Uso de Memória

(c) Latência (d) Throughput

Figura 3. Resultados obtidos a partir da topologia de testes. A Figura (a) apresenta o uso de CPU (% de processamento) dos cinco nodes durante o teste de rede local. A Figura (b) apresenta o consumo de memória no mesmo cenário de testes locais. A Figura (c) retrata a latência em milissegundos nos utilizando roteadores de borda no Brasil, dos EUA e localmente. Por fim, a Figura (d) compara o throughput (em Mbits/s) para os mesmos cenários, para cada edge router.

gateway, apresentaram latência maior, como é o caso do node02. Esse maior tempo de resposta local reflete a proximidade física e maior número de saltos de rede.

Por sua vez, a Figura 3(d) apresenta a vazão, sendo possível verificar o benefício da localidade ao utilizar o OpenZiti. Ao conectar ao *Edge Router* do Brasil, foi possível atingir um valor máximo de 15.1 Mb/s pelo node02, por estar diretamente conectado a um *gateway*. Ao se utilizar o *Edge Router* localizado nos Estados Unidos, a melhor taxa foi alcançada pelo node05 com 2.65 Mb/s. Esse fato demonstra que a escolha de um *Edge Router* geograficamente próximo exerce impacto significativo no desempenho da aplicação, sobretudo daquelas que necessitam de maior taxa de transferência. Já o cenário que não utiliza o OpenZiti, apenas a rede local, apresentou taxas superiores, especialmente o node05, chegando a ultrapassar 40 Mb/s. Como esperado, quanto maior a proximidade física com o *gateway* de teste, maior será a taxa de transferência. Os node01 e node02, precisaram de mais saltos e, consequentemente, tiveram uma taxa muito inferior.

Os testes realizados evidenciaram um comportamento previsível em relação à vazão e latência. Quando os dispositivos estavam conectados ao *Edge Router* do Brasil, a vazão foi, em média, cinco vezes superior que quando conectados ao *Edge Router*

dos EUA. Esse resultado destaca a importância da proximidade geográfica para garantir melhor desempenho. Além disso, foi possível verificar a viabilidade de implementar um mecanismo de *Zero Trust* em dispositivos com recursos computacionais limitados, como é o caso de diversos dispositivos IoT. Esse mecanismo garante que a comunicação seja segura sem impactar o desempenho dos dispositivos. O maior desafio está na melhoria da rede em malha sem fio, sendo esta a responsável pela maior parte da limitação na transferência de dados. Um cliente que necessita de múltiplos saltos para acessar a Internet terá um desempenho inferior independente de utilizar uma arquitetura *Zero Trust*.

7. Conclusão

Este artigo propôs a ZETIn, uma infraestrutura baseada em *Zero Trust* para enfrentar os desafios complexos das redes em malha sem fio (*Wireless Ad-hoc Mesh Networks*) em sistemas autônomos de missão crítica. A adoção do OpenZiti, aliado ao mecanismo de consenso Raft, possibilita a manutenção de uma comunicação segura e resiliente, mesmo em cenários de topologia dinâmica e dispositivos com recursos computacionais limitados. Os experimentos evidenciaram que a sobrecarga imposta pelas políticas de *Zero Trust* é mínima, sem comprometer significativamente o desempenho, como indicado pelos baixos índices de uso de CPU e memória, além da estabilidade observada na latência entre os dispositivos conectados a diferentes *Edge Routers*. Ao utilizar a arquitetura *Zero Trust* para a rede em malha, foi possível constatar que o desempenho não foi afetado significativamente ao utilizar *Edge Routers* em localizações distintas, resultando em benefício ao expor estes dispositivos em ambientes críticos. Embora os testes tenham demonstrado a viabilidade da proposta, algumas limitações foram identificadas. Em redes em malha altamente dinâmicas, a estabilidade do protocolo Raft pode ser impactada por mudanças frequentes na topologia. Embora ainda existam desafios relacionados à estabilidade do protocolo Raft e à necessidade de aprimoramento do *chatbot*, os resultados experimentais confirmam o potencial da solução para garantir conectividade segura e contínua. Assim, a arquitetura ZETIn apresenta-se como uma alternativa robusta para sistemas autônomos que demandam alta disponibilidade e segurança, consolidando a viabilidade do paradigma *Zero Trust* em infraestruturas *ad-hoc*. Como trabalhos futuros, pretende-se explorar a escalabilidade em redes maiores e a incorporação de algoritmos de aprendizado de máquina para a otimização contínua rede.

Referências

Abdalla, A. S., Moore, J., Adhikari, N. e Marojevic, V. (2024). Ztran: Prototyping zero trust security xapps for open radio access network deployments. *IEEE Wireless Communications*, 31(2):66–73.

Al-hammuri, K., Gebali, F. e Kanan, A. (2024). Ztcloudguard: Zero trust context-aware access management framework to avoid medical errors in the era of generative ai and cloud-based health information ecosystems. *AI*, 5(3):1111–1131.

Al-Shareeda, M. A. e Manickam, S. (2023). A systematic literature review on security of vehicular ad-hoc network (vanet) based on veins framework. *IEEE Access*, 11:46218–46228.

Alboqmi, R., Jahan, S. e Gamble, R. F. (2023). A runtime trust evaluation mechanism in the service mesh architecture. Em *2023 10th International Conference on Future Internet of Things and Cloud (FiCloud)*, p. 242–249.

Barbosa, G. N. N. e Mattos, D. M. F. (2024). ZETInChat: Zero Trust Infrastructure with Dynamic Service Deployment via Chatbot in Mesh Networks. Em *Proceedings of the 1st GENZERO Workshop*, Singapore. Springer Nature.

Chandramouli, R. e Butcher, Z. (2023). A zero trust architecture model for access control in cloud-native applications in multi-cloud environments. Relatório técnico, National Institute of Standards and Technology.

Diaz Rivera, J. J., Khan, T. A., Akbar, W., Muhammad, A. e Song, W.-C. (2022). ZT&T: Secure blockchain-based tokens for service session management in zero trust networks. Em *2022 6th Cyber Security in Networking Conference (CSNet)*, p. 1–7.

Hamer, S., d'Amorim, M. e Williams, L. (2024). Just another copy and paste? comparing the security vulnerabilities of chatgpt generated code and stackoverflow answers. Em *2024 IEEE Security and Privacy Workshops (SPW)*, p. 87–94.

Kashyap, R., Azman, M. e Panicker, J. G. (2019). Ubiquitous mesh: a wireless mesh network for iot systems in smart homes and smart cities. Em *2019 IEEE International Conference on Electrical, Computer and Communication Technologies (ICECCT)*, p. 1–5. IEEE.

Khowaja, S. A., Nkenyereye, L., Khowaja, P., Dev, K. e Niyato, D. (2024). Slip: Self-supervised learning based model inversion and poisoning detection-based zero-trust systems for vehicular networks. *IEEE Wireless Communications*, 31(2):50–57.

Kroculick, J. B. (2024). Zero trust decision analysis for next generation networks. Em *Disruptive Technologies in Information Sciences VIII*, volume 13058, p. 278–286. SPIE.

Sheikh, N., Pawar, M. e Lawrence, V. (2021). Zero trust using network micro segmentation. Em *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, p. 1–6.

Stafford, V. (2020). Zero trust architecture. *NIST special publication*, 800:207.

Syed, N. F., Shah, S. W., Shaghaghi, A., Anwar, A., Baig, Z. e Doss, R. (2022). Zero trust architecture (zta): A comprehensive survey. *IEEE Access*, 10:57143–57179.

Tanimoto, S., Yangchen, P., Sato, H. e Kanai, A. (2023). Suitable scalability management model for software-defined perimeter based on zero-trust model. *International Journal of Service and Knowledge Management*, 7(1).

Teerakanok, S., Uehara, T. e Inomata, A. (2021). Migrating to zero trust architecture: Reviews and challenges. *Security and Communication Networks*, 2021(1):9947347.

Zabeeulla, M., singh, A., Sharma, S. K. e Chauhan, S. P. S. (2023). Design and modelling of hybrid network security method for increasing security in vehicular ad-hoc network. *Measurement: Sensors*, 29:100878.

Zheng, Z., Yong, T., Li, J. e Wen, Z. (2022). Simulation research of uaanet based on batman-adv routing protocol. Em *2022 IEEE International Conference on Unmanned Systems (ICUS)*, p. 232–236.

Zivi, A. e Doerr, C. (2022). Adding zero trust in byod environments through network inspection. Em *2022 IEEE Conference on Communications and Network Security (CNS)*, p. 1–6.