# Detecção de Varreduras de Portas pela Análise Inteligente de Tráfego de Rede IoT

**Uelinton Brezolin**[1], **Fernando Nakayama**[1], **Michele Nogueira**[1,2]

[1]Departamento de Informática, Universidade Federal do Paraná

[2]Departamento de Ciência da Computação, Universidade Federal de Minas Gerais

{uelintonbrezolin, fernandonakayama}@ufpr.br, michele@dcc.ufmg.br

***Abstract.*** *Port scanning is a technique to identify the state of a network port. It finds open ports and vulnerabilities in the network or system. Port scanning is a first step in different attack vectors. Hence, it is essential to detect port scans to limit their impacts. Traditional methods for detecting port scans are limited because they rely on static rules and prior knowledge of the structure of the network. This work presents a new method for detecting port scanning in the Internet of Things (IoT) communication, relying on machine learning techniques. The method uses specific traffic features to create a profile of attack behavior. Through neural networks, the developed model identifies port scanning regardless of the network topology. Results show up to 90% efficiency in identifying a port scanning.*

***Resumo.*** *A varredura de portas é uma técnica para identificar o estado de uma porta de rede. Essa técnica encontra portas abertas e vulnerabilidades na rede ou sistema. A varredura de portas é um primeiro passo em diferentes vetores de ataque. Portanto, é essencial detectar essas varreduras de portas para limitar os seus impactos. Os métodos tradicionais para detectar varreduras de portas são limitados porque se baseiam em regras estáticas e no conhecimento prévio da estrutura da rede. Este trabalho apresenta um novo método para a detecção de varredura de portas em comunicação na Internet of Things (IoT), utilizando técnicas de aprendizado de máquina. O método usa recursos de tráfego específicos para criar um perfil de comportamento de ataque. Por meio de uma rede neural, o modelo desenvolvido identifica a varredura de portas independentemente da topologia da rede. Os resultados mostram uma eficiência de até 90% na identificação de uma varredura de portas.*

## 1. Introdução

Os incidentes de segurança afligem continuamente os a deptos da vida digital de usuários de dispositivos da Internet das Coisas (do inglês, *Internet of Things - IoT*). Embora parte dos incidentes necessite da participação dos usuários, como os ataques que envolvem engenharia social ou *malwares*, grande parte resulta de ataques cibernéticos passivos [CERT.BR 2023]. As técnicas de varredura de portas permitem que um atacante colete informações sobre os dispositivos de uma rede como o número das portas de serviço, as configurações de rede, as características de topologia e os tipos de sistemas operacionais, indicando os possíveis alvos. Neste contexto, os incidentes de segurança relacionados à varredura de portas trazem problemas para as redes. A varredura de portas é o

processo de tentativa de conexão com várias portas nos dispositivos de uma rede com o objetivo de determinar as portas operacionais e quais serviços estão sendo executados. Ao identificar essas informações, o atacante explora quaisquer vulnerabilidades conhecidas nos serviços ativos, sendo prejudicial para a segurança da rede.

A varredura de portas é frequentemente o primeiro passo em potenciais ataques cibernéticos [Pittman 2023]. Esse procedimento, que envolve a busca por portas abertas em dispositivos conectados a uma rede, causa uma série de problemas de segurança. A detecção dessas varreduras é vital para proteger a integridade da rede e dos dados armazenados nela. Quando um atacante realiza uma varredura de portas bem-sucedida, ele pode identificar quais serviços estão ativos em um dispositivo específico. Isso fornece ao invasor informações valiosas sobre possíveis vulnerabilidades nos sistemas alvo. Uma vez identificadas as portas abertas, o atacante explora essas vulnerabilidades para obter acesso não autorizado aos sistemas ou lançar ataques mais sofisticados, como injeção de *malware*, sequestro de dados confidenciais ou interrupção dos serviços. Além disso, a varredura de portas também prepara ataques futuros, permitindo ao invasor mapear a rede e identificar pontos fracos para exploração posterior. Outro problema é que, mesmo sem explorar imediatamente as vulnerabilidades encontradas, a simples detecção de uma varredura de portas indica que a rede está sendo alvo de um ataque potencial [Ge et al. 2023].

Na literatura, alguns trabalhos se concentram na detecção de varreduras de portas em redes IoT. [Verma et al. 2022] apresentaram uma abordagem inovadora para a segurança de dispositivos IoT em ambientes de rede local sem fio (WLAN) para detecção de ataques de varredura de portas. [Tang et al. 2020] demonstraram um sistema de penetração NAT baseado em proxy reverso e o algoritmo *ProDASA*, adaptando a frequência de varredura com base no nível de ocupação dos dispositivos alvos. No entanto, essas abordagens se concentram na detecção da disponibilidade ou abertura da porta, negligenciando o ataque que ocorre independentemente da disponibilidade da porta. Por outro lado, [Baah et al. 2022] propuseram uma solução para detecção de varredura de portas utilizando aprendizado de máquina. Em [Lent et al. 2022] aplicou-se um modelo de aprendizado profundo para identificação de varredura de portas com bons resultados na classificação dos ataques. Finalmente, [Hartpence and Kwasinski 2020] aplicaram redes neurais sequenciais para identificar e mitigar a incidência de ataques de varredura de portas. A principal deficiência nas propostas atuais baseadas em aprendizado de máquina é que elas precisam conhecer as características exclusivas da rede e, consequentemente, para cada configuração de rede, um novo modelo precisa ser treinado.

Este trabalho propõe um método para detecção de varredura de portas em redes IoT baseado na análise de tráfego e apoiado em modelos de rede neural. Esses modelos têm a capacidade de lidar com grandes volumes de dados e conseguem identificar padrões complexos e tomar decisões automatizadas. O método detecta o ataque através de uma abordagem simplificada, eficiente, de baixo processamento e que dispensa a necessidade de conhecimento prévio das características da rede a ser observada. O método consiste em três etapas: *(i)* coleta e processamento do tráfego da rede; *(ii)* classificação do tráfego; e *(iii)* a análise dos resultados, identificando atividades de varreduras de portas. Seguindo as etapas, o método coleta, filtra e organiza o tráfego da rede por pacotes. A partir dos pacotes, o método extrai as características de rede e calcula métricas estatísticas para cada conjunto de dados. O método infere uma estimativa de encontrar perfis de comporta-

mentos semelhantes a ataques de varredura de portas sem a necessidade der informações prévias da estrutura da rede. Por fim, o método analisa os resultados da inferência e detecta a presença ou não do perfil de varredura de portas.

A avaliação do método proposto seguiu uma abordagem empírica, composta por três experimentos principais. No primeiro, testou-se o método apenas com tráfego de varredura de portas; no segundo, apenas com tráfego normal; e no terceiro, com ambos os tipos de tráfego. Foram selecionados seis conjuntos de dados para treinamento e testes, sendo quatro de varredura de portas e dois de tráfego normal, incluindo o conjunto *IoT Traffic Traces* e um conjunto localmente criado com dispositivos IoT. A avaliação ocorreu no ponto de acesso dos dispositivos, focando na detecção de comportamentos maliciosos que possam resultar em ataques à rede. Utilizando características extraídas dos conjuntos de dados, foram criadas amostras específicas para cada tipo de tráfego. A aprendizagem dos perfis de comportamentos maliciosos, como varreduras de portas, foi realizado e as amostras foram classificadas. A utilização de conjuntos de dados com finalidades distintas foi fundamental para permitir que as técnicas de rede neural identificassem perfil de comportamentos maliciosos antes do início dos ataques, como a varredura de portas. Os resultados demonstraram uma eficiência na detecção de perfis de comportamento de até 90%, sem a necessidade de conhecimento prévio da estrutura da rede em operação.

Este artigo procede da seguinte forma. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 detalha o método proposto. A Seção 4 descreve a metodologia de avaliação e apresenta os resultados obtidos, incluindo uma discussão sobre os resultados e as limitações do trabalho. Por fim, a Seção 5 conclui o trabalho.

## 2. Trabalhos Relacionados

Diferentes trabalhos detectam atividades de varreduras de portas em tráfego de dispositivos conectados à Internet. Alguns autores, como [Verma et al. 2021], contribuem para maximizar a segurança de dispositivos em Redes locais sem fios (do inglês, *Wireless Local Area Network* – WLAN) habilitados para IPv6. Essa proposta inclui uma varredura na Internet com reconhecimento de rede, buscando um limiar de varredura ideal para reforçar a segurança dos dispositivos. Os autores em [Abu Bakar and Kijsirikul 2023] propõem um mecanismo baseado em DPDK (Data Plane Development Kit) para aprimorar a segurança da rede. O mecanismo supera métodos tradicionais, evidenciando a importância de técnicas avançadas de varredura para identificar e mitigar riscos.

Visando detectar ataques de varredura de portas, os autores em [Jony et al. 2023] apresentam uma técnica para detectar ataques de privação de Protocolo de Configuração de Host Dinâmico (do inglês, *Dynamic Host Configuration Protocol* – DHCP) através de varreduras de portas, superando limitações de métodos existentes. Seu método utiliza pacotes SYN para portas TCP populares, demonstrando detecção precisa e validação eficaz com o *Multivendor Network Emulation Software* (EVE-NG). Os autores em [Verma et al. 2020] enfrentam os desafios de proteger dispositivos em uma rede através das varreduras de portas em toda a Internet com seu algoritmo "estimador". Classificando ambientes de rede de área local sem fio (do inglês, *Wireless Local Area Network* WLAN) com base em medições históricas e em tempo real, sua abordagem contribui para a escolha de estratégias de varredura otimizadas. Entretanto, essas abordagens utilizam a varredura de portas para identificação de ataques, não sendo eficazes para técnicas de

varreduras de portas que ocorrem antes mesmo do ataque ser efetivado.

No contexto de varredura de portas para dispositivos IoT, [Verma et al. 2022] apresentam uma abordagem inovadora para a segurança de dispositivos IoT em ambientes de WLAN dinâmicos, utilizando o tempo de ida e volta e as respostas de pacotes de sondagem para classificar estados ambientais WLAN. Sua metodologia demonstra uma precisão superior a 90% na identificação desses estados, destacando-se pela eficácia na melhoria do desempenho de varreduras de portas em larga escala na Internet. [Tang et al. 2020] focam nos desafios da segurança de IoT, propondo um sistema de penetração NAT baseado em proxy reverso e o algoritmo ProDASA. Este algoritmo adapta a frequência de varredura com base no nível de ocupação dos dispositivos-alvo, equilibrando desempenho e segurança. Experimentos reais e simulações computacionais validam a viabilidade e vantagens dessa abordagem. No entanto, os autores se concentram em métodos de varredura de portas, deixando de considerar que o atacante pode realizar varreduras adicionais e identificar portas que esses métodos não conseguiram detectar.

Existem trabalhos na literatura que empregam técnicas para identificar ataques de varredura de portas e apoiar outros mecanismos. As primeiras iniciativas nessa direção partiram dos autores em [Brahmi et al. 2012] que empregaram técnicas de mineração de dados para otimizar a detecção de varreduras de portas em sistemas de detecção de intrusão. Em [Jemili et al. 2007] os autores empregaram redes Bayesianas para auxiliar na tomada de decisão considerando comportamento anômalo, incluindo varreduras de portas, e legítimo em sistemas de detecção de intrusão. Os autores em [Zhang et al. 2008] aplicaram a mineração de dados e árvores de decisão para identificação de anomalias em sistemas de detecção de intrusão híbridos que lidam com assinaturas e anomalias. Um dos problemas com essas abordagens é que, no caso de mecanismos baseados em assinaturas o mecanismo se torna pouco flexível para detecção e no caso do comportamento anômalo o mecanismo segrega os incidentes em duas categorias somente.

No campo de aprendizagem de máquina, os trabalhos oferecem suporte a outros mecanismos como os sistemas de detecção de intrusão. Entretanto, nota-se que existem esforços mais direcionados para mecanismos que funcionem de maneira autônoma. Os autores em [Baah et al. 2022] propuseram uma solução específica para detecção de varredura de portas em que inicialmente avaliavam a acurácia de sete classificadores de aprendizado de máquina e posteriormente melhoravam os resultados empregando a análise de componentes principais. Os autores em [Lent et al. 2022] propuseram um sistema autônomo para detecção de varredura de portas e ataques de negação de serviço. O sistema trabalha com predição e detecção de anomalias e emprega redes neurais recorrentes na predição e lógica difusa para detecção, caso a predição falhe. Em [Hartpence and Kwasinski 2020] os autores empregaram redes neurais sequenciais e tráfego de rede para classificar pacotes de rede, separar os datagramas TCP, determinar o tipo de pacote TCP e finalmente, detectar a varredura de portas. Os autores utilizaram a estrutura propícia das redes neurais recorrentes para segmentar a tarefa de aprender com o ambiente de rede e realizar a classificação. [Al-Haija et al. 2021] apresentaram um novo esquema abrangente de detecção de ataques de varredura de portas usando regressão logística. O modelo proposto utiliza técnicas de aprendizado de máquina supervisionado para prever e detectar ataques de varredura de portas em redes de comunicação. Os trabalhos referenciados possuem uma característica em comum, as tarefas de treinamento e

classificação para um ambiente de rede específico. Caso o ambiente mude drasticamente, o retreinamento do modelo torna-se necessário.

Portanto, diversos trabalhos abordam varreduras de portas com o objetivo de reconhecer a topologia de uma rede e categorizar portas abertas que, por sua vez, são vulneráveis. Entretanto, enquanto as portas abertas não forem identificadas e as ações corretivas não forem realizadas, o sistema permanecerá vulnerável. Dessa forma, as varreduras visam reconhecer a estrutura de uma rede e as portas de serviço abertas e vulneráveis, sem necessariamente identificar o atacante que está conduzindo os ataques. Portanto, é necessário um método que possa detectar varreduras de portas em uma rede de computadores de maneira eficaz, sem depender de conhecimento prévio da estrutura da rede. Considerando também a proliferação de dispositivos de baixa capacidade computacional nas redes atuais, surge a necessidade de um método de detecção leve e otimizado. Isso representa um avanço nas técnicas de detecção descritas na literatura.

Entre os estudos que abordam a detecção de atividades de varredura de portas em redes de dispositivos conectados à Internet, alguns enfrentam limitações significativas. Muitos desses trabalhos focam na segurança de dispositivos IoT e consideram redes sem fio como principal meio de transmissão de dados. No entanto, vários estudos não abordam a detecção de varreduras de portas conduzidas por atacantes de forma adequada. Além disso, algumas abordagens que se baseiam em técnicas tradicionais, como o uso de assinaturas ou a categorização binária de incidentes, restringem a flexibilidade e a adaptabilidade dos métodos, dificultando a detecção de novos tipos de ataques. Métodos mais recentes, que utilizam aprendizado de máquina, têm se mostrado promissores, mas frequentemente exigem treinamento contínuo e tendem a ser menos eficazes quando ocorrem mudanças significativas no ambiente de rede, necessitando de retreinamentos frequentes.

O método proposto neste estudo se destaca por sua capacidade de detectar atividades de varredura de portas realizadas por atacantes sem a necessidade de conhecimento prévio da estrutura da rede. Dessa forma, ele oferece uma solução eficiente em termos de recursos computacionais e demonstra alta precisão em diversos cenários, incluindo tráfego misto, varreduras de portas e tráfego normal. Essas características tornam-no uma ferramenta robusta e versátil para a detecção de varreduras de portas, adicionando uma camada adicional de segurança para redes de computadores modernas.

## 3. Proposta

Este trabalho apresenta um método de detecção varredura de portas em redes de IoT com o objetivo de detectar varreduras de portas. A detecção é feita a partir da análise do tráfego de rede, utilizando aprendizado profundo. O método busca identificar atividades de atacantes que utilizam a varredura de portas para vasculhar a rede procurando um ponto vulnerável. O método adquire conhecimento sobre padrões do comportamento de varreduras de portas em conjuntos de dados disponíveis na literatura que contêm esse tipo de varredura, a fim de alimentar uma rede neural capaz de identificar esses comportamentos no tráfego sem depender de rótulos prévios e conhecimentos da estrutura de rede que vai atuar. Para isso, o método compreende três etapas apresentadas na Figura 1: *(i)* coleta e processamento do tráfego da rede; *(ii)* classificação do tráfego; e *(iii)* a análise dos resultados, identificando atividades de varreduras de portas. Para o correto funcionamento do método, recomenda-se a implantação do método em um dispositivo com acesso

a todo o tráfego transmitido na rede, como um *gateway* ou *access point*, por exemplo. As próximas subseções detalham cada etapa do método.

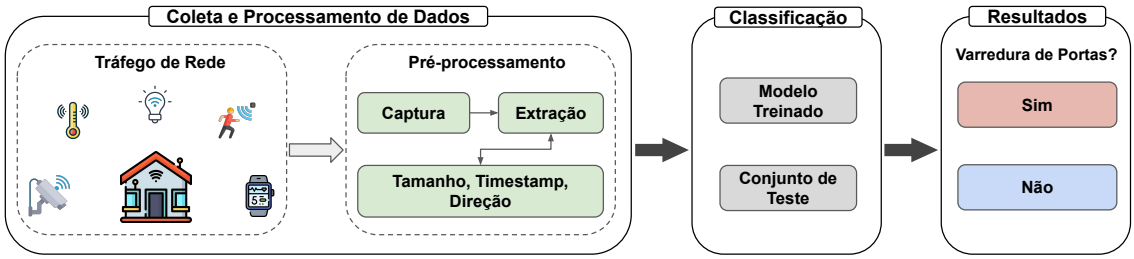


**Figura 1. Etapas de execução do sistema**

### 3.1. Coleta e processamento do tráfego da rede

Esta etapa compreende duas principais ações: *(i)* coleta e filtragem do tráfego da rede, e *(ii)* extração das características dos pacotes de rede. Na ação *(i)*, o método proposto realiza a coleta e filtragem do tráfego da rede sem fio por meio de um *sniffer* operando em modo promíscuo conectado a uma placa de rede ou ao roteador de borda (também referido como *sink node*, *gateway* ou *access point* - AP) da infraestrutura alvo. O *sniffer* utiliza uma interface de rede sem fio, como *Wi-Fi* ou IEEE 802.15.4, programada para capturar pacotes de acordo com a pilha de protocolos Transmission Control Protocol/Internet Protocol (TCP/IP). A ação *(ii)* baseia-se no tráfego da rede capturado, separando as amostras do tráfego bruto em pacotes, incluindo os pacotes transportados pelo Protocolo de Controle de Transmissão (do inglês, Transmission Control Protocol – TCP) e pelo Protocolo de datagrama do usuário (do inglês, User Datagram Protocol – UDP). Dessa forma, essa etapa consiste em calcular uma série de características estatísticas sobre os conjuntos de amostras de pacotes extraídos. As características estatísticas abrangem a média ($\mu$), a soma ($sum$) e o desvio padrão ($DP$) de atributos de rede, como tamanho, *timestamp* e direção dos pacotes. Ao final do processo, cada dispositivo possui suas próprias amostras de pacotes, acompanhadas de suas propriedades estatísticas.

### 3.2. Classificação de varredura de portas

Na segunda etapa do método, procede-se à classificação do tráfego para identificar o comportamento de varredura de portas, independentemente da estrutura da rede. Nesta fase, o tráfego de rede filtrado na etapa anterior é utilizado como entrada, juntamente com um modelo previamente treinado que incorpora comportamentos de tráfego normal e varredura de portas. O objetivo é identificar o comportamento de varredura de portas independentemente da configuração e infraestrutura específica da rede, baseado em um modelo já aprendido. Para atingir esse fim, é essencial contar com um conjunto de dados contendo ambos os tipos de tráfego. A seleção cuidadosa desses conjuntos de dados é fundamental para garantir que a rede neural seja treinada com informações pertinentes e representativas dos comportamentos em questão. Por fim, os ajustes na rede neural são realizados de forma automática para otimizar sua capacidade de identificar e compreender os padrões complexos presentes nos dados, possibilitando assim uma modelagem eficaz do comportamento de varredura de portas.

Assim, o método já incorpora um modelo treinado, ou seja, já aprendeu com os comportamentos de varredura de portas. Esse modelo já treinado seguiu a coleta e processamento de dados semelhante à ocorrida na etapa 1, para realizar a extração de características dos pacotes. No entanto, para esse propósito, utilizou-se o tráfego de rede proveniente de conjuntos de dados disponíveis na literatura. Durante o treinamento, apenas dados de varredura de portas e tráfego normal são fornecidos à rede neural, sendo rotulados como $0$ para varredura de portas e $1$ para tráfego normal. É importante destacar que os rótulos se restringem apenas à fase de aprendizado, onde os dados são rotulados para orientar a rede neural. Após essa etapa, o modelo treinado pode ser aplicado de forma autônoma na detecção de varredura de portas.

A detecção, por sua vez, é feita de acordo com a capacidade de classificação da rede neural retornada pelo classificador, utilizando como entrada o modelo treinado e o tráfego de rede capturado na etapa 1. Dessa forma, o modelo classifica cada amostra (pacote) uma a uma, com uma probabilidade variando de $0$ a $1$, indicando se está mais associada à varredura de portas ou ao tráfego normal. Posteriormente, o método realiza uma média ponderada entre as amostras classificadas como varredura de portas e tráfego normal, chegando a uma média que indica a probabilidade do tráfego em questão ser uma varredura de portas ou não.

### 3.3. Análise de resultados

Na terceira etapa, direciona-se o foco do método para a análise dos resultados gerados pelo classificador na etapa anterior, com o intuito de identificar a presença de varreduras de portas na rede em operação e alertar os administradores responsáveis pela sua gestão. Nesse estágio, são utilizados os resultados da classificação da etapa 2, que consistem na média ponderada do conjunto de probabilidades atribuídas a cada amostra de tráfego processada durante a etapa 1, visando a identificação de atividades de varredura de portas.

Após calcular a média ponderada das probabilidades obtidas pelo classificador, procedemos à análise individual dos resultados de cada classificação, discernindo entre tráfego normal e varredura de portas. Esses resultados são apresentados em uma escala de probabilidades variando de $0$ a $100$ para cada classe. Uma alta probabilidade atribuída à varredura de portas, indicando uma porcentagem significativa de tráfego identificado como tal, sinaliza a possível presença de atividade maliciosa. Se essa atividade maliciosa for detectada, método notificará imediatamente o administrador da rede, que tomará as medidas adequadas de controle. Estas podem incluir a alocação de recursos, gerenciamento de pacotes, controle de tráfego e ajustes no roteamento, entre outras ações necessárias para mitigar os riscos associados à varredura de portas e garantir a segurança da rede que o método está operando.

## 4. Avaliação de desempenho

A avaliação de desempenho do método seguiu uma abordagem empírica, envolvendo três experimentos principais. No primeiro experimento, foi adotada a abordagem de testar o método apenas com tráfego de rede de varredura de portas. No segundo experimento, a abordagem foi testar o método apenas com tráfego normal (sem ataques). Por fim, no terceiro experimento, foi adotada a abordagem de testar o método com ambos os tipos de tráfego. Para isso, um grupo de *datasets* foi selecionado para o treinamento e testes do método. Esse grupo de *datasets* foram fundamentais para validar o método em

questão, visto a gama de tráfego, totalizando seis *datasets* utilizados. Dentre estes, quatro continham tráfego de varredura de portas [Ferrag et al. 2022, Sharafaldin et al. 2018, Almseidin et al. 2022, Huang et al. 2022], enquanto dois consistiam em tráfego normal, sendo um deles o *IoT Traffic Traces* [Sivanathan et al. 2019], e o outro um *dataset* localmente criado com dispositivos IoT.

A avaliação do método foi conduzida no ponto de acesso dos dispositivos. Em redes sem fio, os dispositivos conectados transmitem dados pessoais em forma de pacotes padronizados por protocolos populares, tornando difícil determinar quem está interceptando ou "farejando" (sniffing/ouvindo sem autorização) o tráfego da rede. Nesse cenário, um dispositivo que não implementa adequadamente medidas de segurança pode servir como uma vulnerabilidade potencial, permitindo que usuários mal-intencionados realizem ataques, controlem os dispositivos remotamente, roubem dados pessoais e/ou causem prejuízos financeiros.

Com base nos conjuntos de *datasets*, as características de rede são extraídas para criar amostras específicas para cada tipo de tráfego, normal e varredura de portas. O aprendizado do comportamento da varredura de portas, que abre brechas para os ataques, é realizado, e os comportamentos são classificados. A motivação para o uso desses conjuntos de dados com finalidades distintas tornou-se essencial, permitindo que as técnicas de rede neural identifiquem, inicialmente, os comportamentos maliciosos de varredura de portas presentes no tráfego da rede, antes mesmo do início dos ataques que se beneficiam dessas técnicas, caracterizados como *port scans*.

### 4.1. Características dos conjuntos de dados de varredura de portas

O *dataset* Edge-IIoTset é um conjunto abrangente de dados de segurança cibernética que se concentra em aplicações de IoT e a Internet Industrial das Coisas (IIoT) [Ferrag et al. 2022]. Ele foi gerado usando um ambiente de teste IoT/IIoT especialmente construído, contendo uma ampla variedade de dispositivos, sensores, protocolos e configurações de nuvem/borda. O conjunto de dados inclui mais de 10 tipos de dispositivos IoT, como sensores digitais de baixo custo para temperatura e umidade, sensor ultrassônico, sensor de detecção de nível de água, medidor de pH, sensor de umidade do solo, sensor de frequência cardíaca, sensor de chama, entre outros. Além disso, o *dataset* identifica e analisa catorze tipos de ataques relacionados aos protocolos de conectividade IoT e IIoT, os quais são categorizados em cinco ameaças, incluindo ataques de negação de serviço (DoS/DDoS), coleta de informações, ataque *Man-in-the-Middle*, ataques de injeção e ataques de varredura. Portanto, o *dataset* oferece uma ampla variedade de ataques em redes IoT, principalmente varredura de portas, possibilitando o aprendizado do comportamento da mesma.

O conjunto de dados CIC-IDS2017 [Sharafaldin et al. 2018] é uma coleção abrangente de dados de tráfego de rede cuidadosamente elaborada para refletir as condições do mundo real em termos de diversidade e volume de tráfego. Ele abrange atividades benignas e sete tipos comuns de ataques, incluindo DoS, DDoS, *Brute Force*, XSS, *SQL Injection*, Infiltração, *Port Scan* e *Botnet*. Mais de 80 características de tráfego de rede foram extraídas e calculadas para todas as atividades benignas e intrusivas, utilizando o software CICFlowMeter. O conjunto de dados é totalmente rotulado e foi analisado para selecionar os melhores conjuntos de características para detectar diferentes tipos de

ataques. Além disso, o conjunto de dados fornece informações sobre os horários dos ataques, protocolos utilizados, diversidade de ataques, heterogeneidade dos dados capturados e metadados explicativos. Esses atributos fazem do conjunto de dados uma fonte valiosa para avaliar e aprimorar técnicas de detecção de intrusões.

O Multi-Step Cyber-Attack Dataset (MSCAD) [Almseidin et al. 2022] é um conjunto de dados desenvolvido como referência para sistemas de detecção de intrusão. Ele abrange cenários de ataques de vários passos, como quebra de senha, DDoS baseado em volume e varredura de portas. O MSCAD foi empregado para treinar IDS e avaliar seu desempenho. Em comparação com outros conjuntos de dados, o MSCAD passou com sucesso por doze critérios-chave de avaliação, sendo reconhecido como promissor para treinamento, teste e avaliação de métodos de detecção de ataques de múltiplos passos.

O conjunto de dados de ataques de varredura de portas em *testbed* de emulação e *testbed* com *hardware* em *Loop* [Huang et al. 2022] (do inglês, *dataset of* ***port scanning attacks*** *on emulation testbed and hardware-in-the-loop testbed*) é gerado realizando quatro cenários de ataques de varredura de portas em um sistema de supervisão e aquisição de dados (SCADA) de 8 subestações em três ambientes diferentes, incluindo o minimega no Laboratório Nacional Sandia (SNL), o Emulador de Pesquisa Aberta Comum (CORE) na Universidade Texas A&M e o Testbed RESLab com *hardware* em loop na Universidade Texas A&M. O objetivo desses experimentos é reproduzir os cenários de ataque em diferentes ambientes e validar o sistema de comunicação emulado com sistema baseado em *hardware*, especialmente os controladores programáveis industriais (PLCs). O conjunto de dados gerado pode ser benéfico para a comunidade estudar o comportamento dos ataques de varredura de portas e validar as metodologias para detectar e prevenir ataques de varredura de portas.

### 4.2. Características dos conjuntos de dados IoT

O conjunto de dados *IoT Traffic Traces* [Sivanathan et al. 2019] abrange a coleta do tráfego de rede de uma casa inteligente composta por 28 dispositivos IoT e não-IoT. A coleta ocorreu entre 22 de setembro de 2016 e 12 de outubro de 2016, totalizando 20 dias de captura de tráfego, 29.3 GB de dados e 1.629.879 pacotes. A captura foi dividida em 20 arquivos *pcap*, correspondentes a cada dia de coleta. O *dataset* registra o tráfego de rede de diversos dispositivos, como assistentes virtuais, lâmpadas inteligentes, sensores de movimento, monitores de pressão arterial, monitores de bebê, balanças Wi-Fi, notebooks, smartphones, porta-retratos digitais, caixas de som portáteis, interruptores de luz inteligentes, câmeras de monitoramento Wi-Fi, *gateways*, impressoras, entre outros. Devido à diversidade de dispositivos no conjunto de dados, as características do tráfego de rede variam consideravelmente, incluindo diferentes tamanhos de pacotes. Portanto, o *dataset* oferece uma ampla gama de dados sensíveis coletados pelos dispositivos, possibilitando a avaliação da detecção de vulnerabilidades do mecanismo proposto.

No cenário experimental, os dispositivos estabelecem uma conexão com os serviços na Internet por meio de uma rede local *Wi-Fi*. Tais dispositivos possuem as configurações de fábrica e operam conforme suas respectivas aplicações. A Figura 2 mostra uma foto tirada do ambiente experimental com os dispositivos considerados nas análises. As câmeras de monitoramento transmitem as gravações constantemente para o servidor na nuvem de cada fabricante. O *smartphone* acessa as gravações das câmeras

por meio das aplicações em nuvem. A *smartTV* conecta-se com o serviço de *stream* do *Yotube* e o *Playstation* 4 está conectado a um jogo *online*. Enquanto os dispositivos geram tráfego, um laptop contendo o sistema operacional *Ubuntu Desktop* na versão 20.04 LTS observa a rede sem fio por meio de uma placa de rede e da ferramenta *Wireshark* [Orebaugh et al. 2007]. O tráfego de rede foi coletado por 2 horas, gerando um total de 638166 pacotes e 2,9 GB de dados.

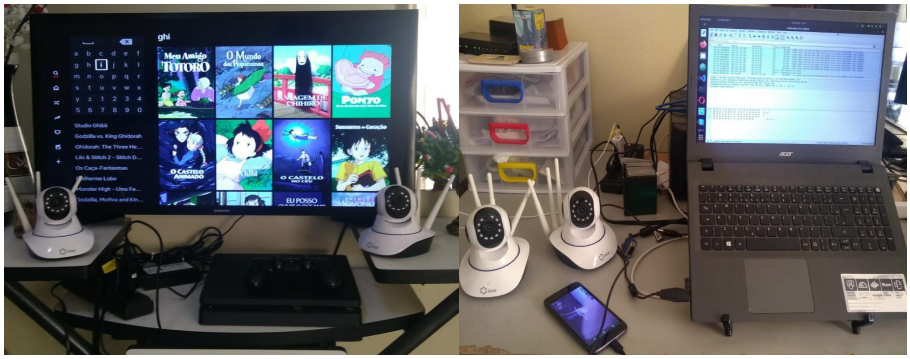

**Figura 2. Cenário Experimental**

### 4.3. Detalhamento da criação do comportamento e classificação

Para cada conjunto de dados, o método extrai o conjunto de características descrito da Subseção 3.1. Essa extração de características do tráfego é realizada por meio da ferramenta Tshark[1], dos *dataset* servem de entrada para a avaliação do método. Assim, por meio das bibliotecas Numpy[2] e Pandas[3] da linguagem Python v3, extraem-se as características da rede, criam-se as amostras e computam-se as medidas estatísticas do tráfego. As características de rede consideradas envolvem as características tamanho do pacote, *timestamp* e direção do pacote 0 para ida e 1 para volta. Para o cálculo das medidas estatísticas, foram amostras com 3 pacotes, visto que alguns dispositivos geraram pouca quantidade de pacotes de rede (exemplo, 10 pacotes por dia). A partir das amostras, são computadas as medidas estatísticas da média, soma e desvio padrão.

Cada conjunto de dados possui requisitos específicos e tipos de tráfego distintos. Além dos registros de varreduras de portas, esses conjuntos também incluem outros tipos de ataques, como DDos, Brute Force e *Man-in-the-middle*, entre outros. Para este estudo, apenas o tráfego de varredura de portas foi selecionado dos conjuntos de dados que continham esse tipo de ataque, de acordo com as descrições fornecidas pelos autores, que incluíam data, hora, minutos e segundos da execução da varredura, juntamente com os IPs e MACs dos agentes envolvidos no ataque, sendo realizado uma filtragem destes pacotes para realizar os experimentos.

Após a filtragem dos conjuntos de dados, foi realizada uma fusão dos dados filtrados de cada *dataset* para fins de treinamento e teste da rede neural. Um dos principais problemas relacionados à identificação de varreduras de portas através de modelos de

[1]Tshark - Wireshark: https://www.wireshark.org/docs/man-pages/tshark.html. Acessado em Jan/2024.
[2]Biblioteca Numpy: https://numpy.org/. Acessado em Jan/2024.
[3]Biblioteca Pandas: https://pandas.pydata.org/. Acessado em Jan/2024.

aprendizado de máquina considerando-se o tráfego de rede é a ausência de dados específicos para treinamento e teste dos modelos. A fusão de conjuntos de dados tem como objetivo melhorar a generalização e a qualidade dos dados, além de aumentar a precisão na categorização das varreduras de portas. A fusão de conjuntos de dados é empregada em diversas áreas do aprendizado de máquina, como a análise de sentimentos em redes sociais [Fortuna et al. 2018] e cibersegurança [Abbiati et al. 2021], e envolve o uso de diferentes características dos conjuntos de dados, bem como a utilização de múltiplos algoritmos para sua construção. A Tabela 1 apresenta essa divisão, na qual a rede neural foi treinada com ambos os tipos de tráfego: tráfego normal e tráfego de varredura de portas (ataque). O teste foi realizado com conjuntos de dados contendo tanto tráfego normal quanto varredura de portas, porém, a rede neural não tinha conhecimento prévio desse tráfego como entrada. O objetivo desse teste era validar a capacidade do método de identificar varreduras de portas com base apenas no comportamento desse tipo de ataque, sem conhecimento prévio do tráfego de entrada e da estrutura da rede atuante.

**Tabela 1. Distribuição dos *Datasets***

| **Treino** | | **Teste** |
|---|---|---|
| **Ataque** | **Normal** | |
| CIC-IDS2017 | *IoT Traffic Traces* | Cenário Experimental |
| Edge-IIoTset | – | MSCAD |
| *Port Scanning Attacks* | – | – |

### 4.4. Configuração do aprendizado

A rede neural utilizou as bibliotecas, Keras[4] e TensorFlow[5] para seu funcionamento. A rede neural foi configurada com uma arquitetura sequencial de três camadas densamente conectadas, adequada para o método proposto de detecção de tráfego de varreduras de portas e fornecendo resultados confiáveis. A escolha da função de ativação *ReLU* nas camadas de entrada e oculta permite a introdução de não linearidades essenciais para capturar padrões complexos nos dados. Essa propriedade é crucial para identificar comportamentos distintos associados ao tráfego de varreduras de portas, que podem ser sutis e não lineares. Além disso, a função de ativação *sigmoide* na camada de saída é apropriada para problemas de classificação binária como é o caso, onde o objetivo é discernir entre duas classes possíveis: tráfego de varredura de portas e não tráfego de varredura. Ao produzir uma saída entre $0$ e $1$, a função *sigmoide* fornece uma interpretação direta da probabilidade de uma determinada entrada pertencer à classe positiva, facilitando a tomada de decisões. Essa abordagem é bem estabelecida em problemas de classificação binária e é especialmente eficaz em cenários onde há uma necessidade de modelar relações não lineares nos dados e obter probabilidades interpretáveis para as classes-alvo. Portanto, essa rede neural é altamente apropriada e tem o potencial de fornecer resultados robustos na detecção de tráfego de varreduras de portas.

[4]Keras: https://keras.io/. Acessado em Jan/2024

[5]TensorFlow: https://www.tensorflow.org/. Acessado em Jan/2024

### 4.5. Resultados

Esta subseção expõe os resultados do método de detecção de varredura de portas proposto neste trabalho. Na Tabela 2, os resultados pertinentes à avaliação empírica do método descrito na seção anterior são apresentados. Esses resultados foram obtidos conforme delineado na Tabela 1, na qual foram realizados três tipos de experimentos. O primeiro experimento consistiu na utilização exclusiva de tráfego de varreduras de portas, o segundo experimento empregou exclusivamente tráfego normal, enquanto o terceiro experimento empregou ambos os tipos de tráfego.

O primeiro experimento foi realizado utilizando exclusivamente o tráfego de varredura de portas extraído e filtrado do conjunto de dados MSCAD [Almseidin et al. 2022], com o objetivo de demonstrar a capacidade do método em detectar com precisão o comportamento de um ataque de varredura de portas sem ter prévio conhecimento do tráfego de entrada. Conforme ilustrado na Tabela 2, com o tráfego de ataque como entrada, observamos um total de $38,852$ amostras (pacotes) para as quais o método não possui conhecimento prévio. Neste contexto, notamos que o método apresenta uma classificação satisfatória em relação à quantidade de amostras classificadas como varredura de portas, identificando corretamente $33,809$ amostras como *port scan*, o que representa verdadeiros positivos, enquanto apenas $4,943$ amostras foram erroneamente classificadas, caracterizando falsos negativos. O método demonstra uma taxa de acerto de $87\%$ na identificação do tráfego de varredura de portas como entrada, o que valida sua capacidade de detecção desse tipo de tráfego sem prévio conhecimento do mesmo.

**Tabela 2. Resultados da Avaliação do Método**

| Exp. | Entrada | Quant. Amostras | Tipo | Classificação | Métricas |
|---|---|---|---|---|---|
| 1 | Tráfego Ataque | 38,852 | PortScan | 33,909 | VP |
| | | | Erro | 4,943 | FN |
| 2 | Tráfego Normal | 1,000,000 | Normal | 729,989 | VN |
| | | | Erro | 270,011 | FP |
| 3 | Ambos Tráfegos | Ataque 38,852<br>Normal 1,000,000 | PortScan | 303,920 | VP + FP |
| | | | Normal | 734,932 | VN + FN |

O segundo experimento foi conduzido exclusivamente com o tráfego normal de dispositivos IoT, extraído e filtrado do conjunto de dados IoT *Traffic Traces*[Sivanathan et al. 2019], com o propósito de evidenciar a capacidade do método em distinguir entre tráfego normal e tráfego de varredura de portas, mesmo sem prévio conhecimento do tráfego de entrada. Conforme destacado na Tabela 2, ao utilizar o Tráfego Normal como entrada, registramos um total de $1,000,000$ amostras (pacotes) para as quais o método não tinha conhecimento prévio. Nesse contexto, observamos que o método apresenta uma classificação satisfatória em relação à quantidade de amostras classificadas como tráfego normal, identificando corretamente $729,989$ amostras como tráfego normal, o que representa verdadeiros negativos, enquanto apenas $270,011$ amostras foram erroneamente classificadas, caracterizando falsos positivos. É relevante notar que o método demonstra uma taxa de acerto de 73% na identificação do tráfego normal como entrada, validando sua capacidade de distinguir entre tráfego normal e tráfego de varredura de portas para uma detecção precisa do ataque.

O terceiro experimento foi conduzido utilizando ambos os tráfegos: varredura de portas MSCAD [Almseidin et al. 2022] e tráfego normal [Sivanathan et al. 2019], simulando um cenário real. O objetivo foi demonstrar que o método é capaz de distinguir entre tráfego normal e tráfego de varredura de portas, validando a possibilidade de identificar essa atividade maliciosa em redes de computadores, mesmo sem qualquer conhecimento prévio do tráfego de entrada. Conforme evidenciado na Tabela 2, ao utilizar ambos os tráfegos como entrada, foram processadas um total de $1,038,852$ amostras (pacotes) para as quais o método não possui conhecimento prévio. Dentro desse conjunto de amostras, o método classificou $303,920$ como varredura de portas. Dentre essas amostras classificadas como varredura de portas, $34,000$ foram corretamente identificadas como verdadeiros positivos, enquanto aproximadamente $260,000$ amostras foram classificadas erroneamente como falsos positivos, indicando tráfego normal. O método classificou um total de $734,932$ amostras como tráfego normal, dos quais $730,000$ foram verdadeiros negativos, classificados corretamente, enquanto cerca de $4,000$ foram classificados erroneamente como falsos negativos, caracterizando tráfego de varredura de portas. Assim, o método demonstra uma alta taxa de classificação de amostras de varreduras de portas, com cerca de 90% das amostras corretamente classificadas, mesmo em um cenário de tráfego misto, ou seja, contendo tráfego normal e tráfego de varredura de portas.

Os resultados obtidos nesta seção evidenciam a eficácia do método de detecção de varredura de portas proposto neste estudo. Através dos três tipos de experimentos realizados, foi possível observar consistentes taxas de acerto na identificação de tráfego malicioso e tráfego normal, mesmo sem qualquer conhecimento prévio do tráfego de entrada. No primeiro experimento, onde empregamos exclusivamente tráfego de varreduras de portas, o método demonstrou uma taxa de acerto de 87%, evidenciando sua capacidade de detectar esse tipo de atividade maliciosa de forma precisa. No segundo experimento, com tráfego normal, alcançamos uma taxa de acerto de 73%, destacando a habilidade do método em distinguir entre padrões de tráfego benignos e maliciosos. O terceiro experimento, simulando um cenário real com ambos os tipos de tráfego, revelou uma taxa de classificação ainda mais robusta, com cerca de 88% das amostras de varreduras de portas corretamente identificadas. Esses resultados consolidam a confiabilidade e a eficiência do método proposto, fornecendo uma valiosa ferramenta para a detecção precisa de atividades maliciosas em redes de computadores, mesmo em ambientes complexos e dinâmicos.

**Discussão dos resultados**

Os resultados obtidos a partir dos experimentos demonstram a eficácia do método de detecção de varredura de portas proposto. Cada experimento foi cuidadosamente delineado para testar diferentes aspectos do método, proporcionando uma visão abrangente de seu desempenho em diversos cenários de tráfego. No primeiro experimento, utilizamos exclusivamente tráfego de varredura de portas para avaliar a capacidade do método em identificar ataques sem conhecimento prévio do tráfego de entrada. A taxa de acerto de 87%, com $33,809$ verdadeiros positivos e $4,943$ falsos negativos, indica que o método é altamente eficiente na detecção de comportamentos maliciosos específicos de varredura de portas. Esta alta taxa de detecção é atribuída ao fato de que varreduras de portas possuem comportamento que o método proposto foi capaz de aprender com sucesso, sendo capaz de reconhecer mesmo sem conhecimento prévio do tráfego de entrada.

O segundo experimento focou no tráfego normal de dispositivos IoT, com o objetivo de testar a capacidade do método em distinguir entre tráfego benigno e malicioso. A taxa de acerto de 73%, com 729,989 verdadeiros negativos e $270,011$ falsos positivos, sugere que o método, embora eficaz, enfrenta desafios na redução de falsos positivos em tráfego exclusivamente benigno. Isso pode ocorrer porque o tráfego normal apresenta variações e comportamentos que, embora não sejam maliciosos, se assemelham a padrões de varredura de portas, levando a classificações incorretas. No terceiro experimento, que combinou tráfego de varredura de portas e tráfego normal, o método mostrou uma taxa de acerto de aproximadamente 88% na classificação correta de varreduras de portas. Com $34,000$ verdadeiros positivos e cerca de $260,000$ falsos positivos, além de $730,000$ verdadeiros negativos e aproximadamente $4,000$ falsos negativos, o método demonstrou uma robustez notável em um ambiente de tráfego misto. A alta taxa de acerto nesse cenário reflete a capacidade do método de adaptar-se a contextos mais realistas e complexos, onde a distinção entre tráfego normal e malicioso é crucial para a segurança da rede.

Para a realização dos experimentos e obtenção dos resultados, foi empregada uma máquina virtual local configurada com um processador Intel® Xeon® Silver 4114 CPU @ 2.20GHz, com arquitetura de 64 bits. A memória RAM disponível para os testes foi de 20 GB, proporcionando um ambiente robusto para o processamento de dados intensivos. O armazenamento foi dimensionado em 100 GB, atendendo eficientemente às necessidades de armazenamento de grandes volumes de dados. Importante ressaltar que a configuração utilizada não incluiu uma placa de vídeo dedicada, focando exclusivamente na capacidade de processamento e memória da máquina virtual local. Essa configuração permitiu um desempenho consistente e adequado para a execução dos experimentos propostos.

Em relação às restrições do método proposto, uma das limitações observadas foi a alta taxa de falsos positivos no experimento com tráfego exclusivamente normal, em que $270.011$ amostras foram incorretamente classificadas como varredura de portas. Esse elevado número de falsos positivos indica que o método pode confundir comportamentos benignos com atividades maliciosas. Em redes IoT, onde a variabilidade do tráfego é alta, a dificuldade do método em distinguir esses tipos de tráfego é significativa, e estamos trabalhando para resolver essa limitação. Outra limitação que está em análise é a eficácia do método contra ataques adversários ou técnicas que tentam camuflar varreduras de portas.

## 5. Conclusão

Este artigo apresentou um método baseado em uma abordagem inovadora que combina análise de tráfego de rede com aprendizado profundo para detectar varreduras de portas em redes de comunicação IoT, demonstrando eficácia mesmo sem conhecimento prévio do tráfego de entrada. O método opera em três etapas principais: coleta e processamento do tráfego da rede, classificação do tráfego e análise dos resultados. Os resultados dos testes realizados mostraram taxas de acerto consistentes, com uma taxa de acerto de 87% para tráfego exclusivamente de varredura de portas. No cenário mais desafiador, com ambos os tipos de tráfego, o método apresentou uma taxa de classificação ainda mais robusta, identificando corretamente cerca de 88% das amostras de varredura de portas. Essa abordagem inovadora oferece uma solução eficaz e automatizada para detectar e mitigar ameaças de segurança cibernética em ambientes de rede complexos, distinguindo entre tráfego malicioso e tráfego benigno e alertando os administradores de rede sobre possíveis atividades suspeitas.

## 6. Agradecimentos

Os autores agradecem o apoio da UFPR, UFMG e da Fundação de Amparo à Pesquisa do Estado de São Paulo (FAPESP – #2021/06733-6, #2022/14299-7, #2022/15573-5). Este trabalho foi realizado com apoio da Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Código de Financiamento 001.

## Referências

Abbiati, G., Ranise, S., Schizzerotto, A., and Siena, A. (2021). Merging datasets of cybersecurity incidents for fun and insight. *Frontiers in Big Data*, 3.

Abu Bakar, R. and Kijsirikul, B. (2023). Enhancing network visibility and security with advanced port scanning techniques. *Sensors*, 23(17).

Al-Haija, Q. A., Saleh, E., and Alnabhan, M. (2021). Detecting port scan attacks using logistic regression. In *2021 4th International Symposium on Advanced Electrical and Communication Technologies (ISAECT)*, pages 1–5.

Almseidin, M., Al-Sawwa, J., and Alkasassbeh, M. (2022). Multi-step cyber-attack dataset (mscad for intrusion detection).

Baah, E. K., Yirenkyi, S., Asamoah, D., Oppong, S. O., Opoku-Mensah, E., Partey, B. T., Sackey, A. K., Kornyo, O., and Obu, E. (2022). Enhancing port scans attack detection using principal component analysis and machine learning algorithms. In *International Conference on Frontiers in Cyber Security*, pages 119–133. Springer.

Brahmi, H., Brahmi, I., and Ben Yahia, S. (2012). Omc-ids: at the cross-roads of olap mining and intrusion detection. In *Advances in Knowledge Discovery and Data Mining: 16th Pacific-Asia Conference, PAKDD, Kuala Lumpur, Malaysia, May 29–June 1, 2012, Proceedings, Part II 16*, pages 13–24. Springer.

CERT.BR (2023). Estatísticas dos Incidentes Reportados ao CERT. br. Disponível em: https://stats.cert.br/. Acessado em Janeiro, 2024.

Ferrag, M. A., Friha, O., Hamouda, D., Maglaras, L., and Janicke, H. (2022). Edge-iiotset: A new comprehensive realistic cyber security dataset of iot and iiot applications: Centralized and federated learning.

Fortuna, P., Ferreira, J., Pires, L., Routar, G., and Nunes, S. (2018). Merging datasets for aggressive text identification. In Kumar, R., Ojha, A. K., Zampieri, M., and Malmasi, S., editors, *Proceedings of the First Workshop on Trolling, Aggression and Cyberbullying (TRAC-2018)*, pages 128–139, Santa Fe, New Mexico, USA. Association for Computational Linguistics.

Ge, J., Li, T., and Wu, Y. (2023). *Online Encrypted Traffic Classification Based on Lightweight Neural Networks*, pages 109–128. Wiley-IEEE Press.

Hartpence, B. and Kwasinski, A. (2020). Combating tcp port scan attacks using sequential neural networks. In *International Conference on Computing, Networking and Communications (ICNC)*, pages 256–260. IEEE.

Huang, H., Wlazlo, P., Sahu, A., Walker, A., Goulart, A., Davis, K., Swiler, L., Tarman, T., and Vugrin, E. (2022). Dataset of port scanning attacks on emulation testbed and hardware-in-the-loop testbed.

Jemili, F., Zaghdoud, M., and Ahmed, M. B. (2007). A framework for an adaptive intrusion detection system using bayesian network. In *IEEE Intelligence and Security Informatics*, pages 66–70. IEEE.

Jony, A., Miah, A. S. M., and Islam, M. N. (2023). An effective method to detect dhcp starvation attack using port scanning. In *International Conference on Next-Generation Computing, IoT and Machine Learning (NCIM)*, pages 1–6.

Lent, D. M. B., Novaes, M. P., Carvalho, L. F., Lloret, J., Rodrigues, J. J., and Proença, M. L. (2022). A gated recurrent unit deep learning model to detect and mitigate distributed denial of service and portscan attacks. *IEEE Access*, 10:73229–73242.

Orebaugh, A., Ramirez, G., Beale, J., and Wright, J. (2007). *Wireshark & Ethereal Network Protocol Analyzer Toolkit*. Syngress Publishing.

Pittman, J. M. (2023). Machine learning and port scans: A systematic review. *arXiv preprint arXiv:2301.13581*.

Sharafaldin, I., Lashkari, A. H., Ghorbani, A. A., et al. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. *ICISSp*, 1:108–116.

Sivanathan, A., Gharakheili, H. H., Loi, F., Radford, A., Wijenayake, C., Vishwanath, A., and Sivaraman, V. (2019). Classifying IoT devices in smart environments using network traffic characteristics. *IEEE Transactions on Mobile Computing*, 18(8):1745–1759.

Tang, F., Kawamoto, Y., Kato, N., Yano, K., and Suzuki, Y. (2020). Probe delay based adaptive port scanning for iot devices with private ip address behind nat. *IEEE Network*, 34(2):195–201.

Verma, S., Kawamoto, Y., and Kato, N. (2020). A novel iot-aware wlan environment identification for efficient internet-wide port scan. In *IEEE Global Communications Conference - GLOBECOM*, pages 1–6.

Verma, S., Kawamoto, Y., and Kato, N. (2021). A network-aware internet-wide scan for security maximization of ipv6-enabled wlan iot devices. *IEEE Internet of Things Journal*, 8(10):8411–8422.

Verma, S., Kawamoto, Y., and Kato, N. (2022). A smart internet-wide port scan approach for improving iot security under dynamic wlan environments. *IEEE Internet of Things Journal*, 9(14):11951–11961.

Zhang, J., Zulkernine, M., and Haque, A. (2008). Random-forests-based network intrusion detection systems. *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, 38(5):649–659.