Universidade Estadual de Londrina

ISABELA HARA BANDO

DETECÇÃO DE ATAQUES EM INTERNET DAS COISAS UTILIZANDO MINERAÇÃO DE FLUXOS CONTÍNUOS DE DADOS

LONDRINA
2023

ISABELA HARA BANDO

DETECÇÃO DE ATAQUES EM INTERNET DAS COISAS UTILIZANDO MINERAÇÃO DE FLUXOS CONTÍNUOS DE DADOS

Versão Preliminar de Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Bacharel em Ciência da Computação.

Orientador: Prof. Dr. Bruno Bogaz Zarpelão

LONDRINA
2023

Ficha de identificação da obra elaborada pelo autor, através do Programa de Geração Automática do Sistema de Bibliotecas da UEL

Sobrenome, Nome.
Título do Trabalho : Subtitulo do Trabalho / Nome Sobrenome. - Londrina, 2017.
100 f. : il.

Orientador: Nome do Orientador Sobrenome do Orientador.
Coorientador: Nome Coorientador Sobrenome Coorientador.
Dissertação (Mestrado em Ciência da Computação) - Universidade Estadual de Londrina, Centro de Ciências Exatas, Programa de Pós-Graduação em Ciência da Computação, 2017.
Inclui bibliografia.

1. Assunto 1 - Tese. 2. Assunto 2 - Tese. 3. Assunto 3 - Tese. 4. Assunto 4 - Tese. I. Sobrenome do Orientador, Nome do Orientador. II. Sobrenome Coorientador, Nome Coorientador. III. Universidade Estadual de Londrina. Centro de Ciências Exatas. Programa de Pós-Graduação em Ciência da Computação. IV. Título.

ISABELA HARA BANDO

DETECÇÃO DE ATAQUES EM INTERNET DAS COISAS UTILIZANDO MINERAÇÃO DE FLUXOS CONTÍNUOS DE DADOS

Versão Preliminar de Trabalho de Conclusão de Curso apresentado ao curso de Bacharelado em Ciência da Computação da Universidade Estadual de Londrina para obtenção do título de Bacharel em Ciência da Computação.

BANCA EXAMINADORA

Orientador: Prof. Dr. Bruno Bogaz Zarpelão
Universidade Estadual de Londrina

Prof. Dr. Segundo Membro da Banca
Universidade/Instituição do Segundo Membro da Banca – Sigla instituição

---

Prof. Dr. Terceiro Membro da Banca
Universidade/Instituição do Terceiro Membro da Banca – Sigla instituição

Londrina, 11 de dezembro de 2023.

Este trabalho é dedicado às crianças adultas
que, quando pequenas, sonharam em se
tornar cientistas.

AGRADECIMENTOS

Os agradecimentos principais são direcionados à Gerald Weber, Miguel Frasson, Leslie H. Watter, Bruno Parente Lima, Flávio de Vasconcellos Corrêa, Otavio Real Salvador, Renato Machnievscz[1] e todos aqueles que contribuíram para que a produção de trabalhos acadêmicos conforme as normas ABNT com LaTeX fosse possível.

Agradecimentos especiais são direcionados ao Centro de Pesquisa em Arquitetura da Informação[2] da Universidade de Brasília (CPAI), ao grupo de usuários *latex-br*[3] e aos novos voluntários do grupo *abnTeX2*[4] que contribuíram e que ainda contribuirão para a evolução do abnTeX2.

[1] Os nomes dos integrantes do primeiro projeto abnTeX foram extraídos de <http://codigolivre.org.br/projects/abntex/>

[2] <http://www.cpai.unb.br/>

[3] <http://groups.google.com/group/latex-br>

[4] <http://groups.google.com/group/abntex2> e <http://abntex2.googlecode.com/>

"Não vos amoldeis às estruturas deste mundo, mas transformai-vos pela renovação da mente, a fim de distinguir qual é a vontade de Deus: o que é bom, o que Lhe é agradável, o que é perfeito.
(Bíblia Sagrada, Romanos 12, 2))

BANDO, I. H.. **Detecção de ataques em Internet das Coisas utilizando Mineração de Fluxos Contínuos de Dados**. 2023. 31f. Trabalho de Conclusão de Curso – Versão Preliminar (Bacharelado em Ciência da Computação) – Universidade Estadual de Londrina, Londrina, 2023.

RESUMO

O desenvolvimento da tecnologia nos últimos anos aumentou significativamente o uso de dispositivos inteligentes interconectados, isto é, a Internet das Coisas (*Internet of Things* - IoT). Devido ao grande fluxo de informações compartilhadas e armazenadas, garantir a segurança das redes IoT é essencial, e também um enorme desafio, tendo em vista os diversos tipos de ciberataques que ameaçam essas redes atualmente. Grande parte das soluções estudadas para detecção de ataques cibernéticos utilizam algoritmos de aprendizado em lote, cujo treinamento é feito a partir de uma base de dados estática, o que leva à perda de eficácia com o surgimento de novos comportamentos na rede. Os dados monitorados passam, constantemente, por muitas mudanças naturais e por isso, é necessário que os algoritmos sejam capazes de se adaptar com uma maior facilidade. Levando em conta esse contexto, este projeto visa estudar e implementar diferentes algoritmos de mineração de fluxo de dados contínuos para detecção de intrusões em redes IoT, utilizando o aprendizado incremental. A eficácia de detecção de cada um destes algoritmos será avaliada em múltiplos conjuntos de dados disponibilizados publicamente.

Palavras-chave: Inteligência Computacional. Fluxos contínuos de dados. Sistema de Detecção de Intrusão. Aprendizado de Máquina.

BANDO, I. H.. **Attack detection in Internet of Things using Continuous Stream Data Mining**. 2023. 31p. Final Project – Draft Version (Bachelor of Science in Computer Science) – State University of Londrina, Londrina, 2023.

ABSTRACT

The development of technology in recent years has significantly increased the use of interconnected smart devices, that is, the Internet of Things (IoT). Due to the large flow of shared and stored information, ensuring the security of IoT networks is essential, and also a huge challenge, given the several forms of cyberattacks that threaten these networks today. Most of the solutions studied for detecting cyber attacks use batch learning algorithms, which are trained from a static database, which leads to loss of effectiveness as new behaviors emerge in the network. The monitored data is constantly undergoing many natural changes and therefore it is necessary that the algorithms are able to adapt more easily. In this context, this project aims to study and implement different continuous data stream mining algorithms for intrusion detection in IoT networks, using incremental learning. The detection effectiveness of each of these algorithms will be evaluated on multiple publicly available datasets.

Keywords: Computational Intelligence. Continuous Data Streams. Intrusion Detection System. Machine Learning.

LISTA DE ILUSTRAÇÕES

LISTA DE TABELAS

LISTA DE ABREVIATURAS E SIGLAS

IoT	Internet of Things
IDS	Intrusion Detection System
SVM	Support Vector Machine
OCSVM	One-class Support Vector Machine

SUMÁRIO

1 INTRODUÇÃO

Com o avanço tecnológico e o aumento constante da conectividade, a Internet das Coisas (*Internet of Things* - IoT) emergiu como um campo de grande potencial de desenvolvimento. Além disso, a IoT está se tornando cada vez mais integrada à vida cotidiana das pessoas, estando presente em uma ampla variedade de dispositivos inteligentes, desde eletrodomésticos e sistemas de segurança até veículos autônomos e equipamentos médicos. Portanto, garantir a segurança das redes IoT e dos sistemas é um desafio relevante, que tem sido objeto de estudos ao longo dos anos [2, 3, 4, 5].

Devido à diversidade de ameaças existentes, à falta de conscientização dos usuários e à ausência de atualizações regulares de software, torna-se imperativo desenvolver modelos capazes de se adaptar às mudanças e depender cada vez menos da intervenção humana.

Embora os algoritmos de aprendizado em lote (*batch*) tenham sido amplamente utilizados na detecção de ataques em redes [6, 7, 8], eles possuem uma limitação significativa: são treinados com um conjunto de dados estático e têm dificuldade em lidar com mudanças na rede, como mudanças de conceito (*concept drift*), exigindo a atualização ou retreinamento do modelo [9, 10]. Como as redes de computadores estão em constante evolução e os padrões de tráfego podem variar com o tempo, modelos treinados com dados do passado podem perder sua eficácia rapidamente, resultando em taxas de detecção de ataques mais baixas.

Dada essa limitação, outra abordagem que pode ser explorada é o uso de algoritmos de aprendizado em fluxo contínuo de dados. Esses algoritmos são projetados para lidar com situações que exigem aprendizado incremental, permitindo que o modelo seja ajustado à medida que novos dados são recebidos em um fluxo contínuo, sem a necessidade de processar novamente todo o conjunto de dados. Estudos recentes já começaram a investigar essa abordagem [11, 12, 13], embora abranjam apenas uma parte dos algoritmos potenciais que podem ser utilizados, indicando que ainda há muito a ser explorado nesse campo.

Neste projeto, serão estudados e implementados diferentes algoritmos de mineração de fluxo de dados contínuos, com o objetivo de avaliar o potencial de cada um deles na identificação de ataques nas redes IoT. Para tanto, primeiramente, será realizado um levantamento de conjuntos de dados públicos que contenham tráfego de rede IoT e atendam dois requisitos: a presença de mudanças naturais de comportamento, para avaliar a capacidade de adaptação dos algoritmos, e diversidade de ataques, para compreender melhor o potencial de detecção frente a diferentes ameaças. Na sequência, os algoritmos serão implementados e testados sobre os conjuntos selecionados, buscando avaliar principalmente

métricas de desempenho preditivo. Por fim, serão investigadas técnicas para diminuição da demanda por exemplos rotulados para o treinamento dos modelos de aprendizado.

2 FUNDAMENTAÇÃO TEÓRICO-METODOLÓGICA E ESTADO DA ARTE

2.1 Internet das Coisas

O termo "Internet das Coisas"foi popularizado pelo britânico Kevin Ashton, em 1999 [14], e desde então seu uso vem crescendo cada vez mais. Atualmente, a IoT é um paradigma tecnológico que se refere à interconexão de dispositivos do nosso cotidiano por meio da Internet. Esses dispositivos estão equipados com sensores, atuadores e tecnologia de comunicação que permitem a coleta, troca e análise de dados, possibilitando assim, a automação de tarefas e a tomada de decisões em tempo real [15].

Para entender a melhor a respeito dessa rede interconectada, é imperativo compreender sua arquitetura, que é fundamentada em camadas ou fases, que podem variar em nome e quantidade dependendo da abordagem ou modelo específico, mas uma arquitetura típica pode incluir três camadas principais, conforme a apresentada na Figura 1 [1].

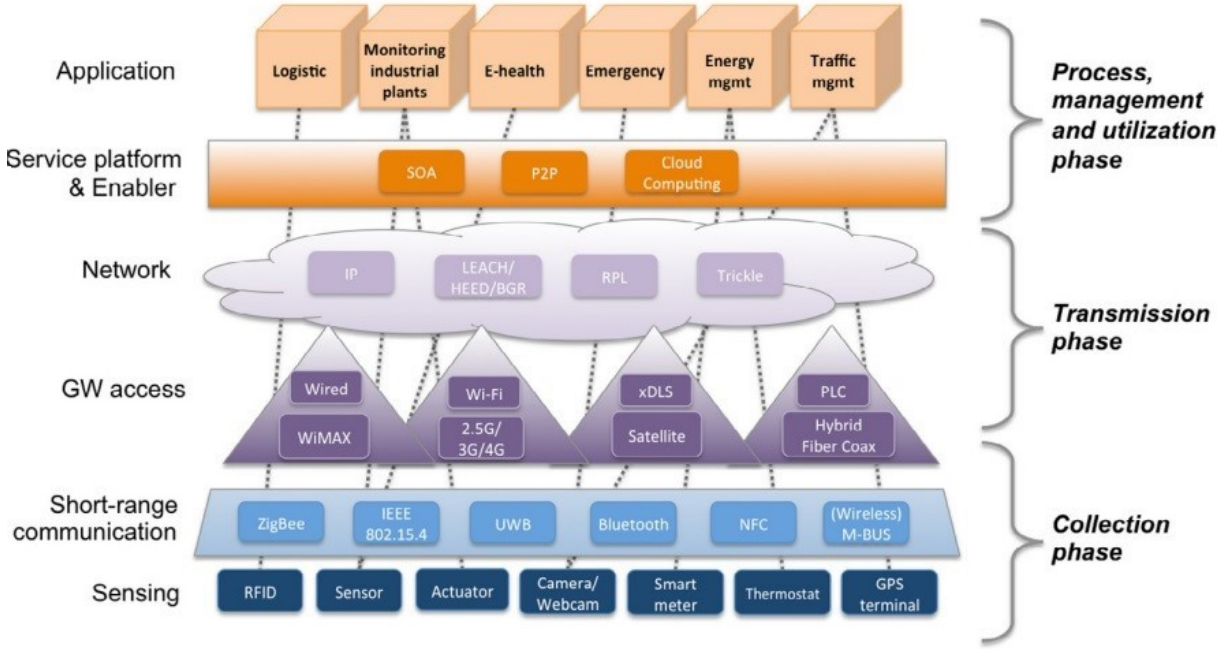

Figura 1 – Representação horizontal para aplicações de IoT. [1]

As camadas são divididas da seguinte forma:

- Camada de Coleta (ou Camada de Percepção): composta pelos dispositivos sensores e atuadores distribuídos fisicamente no ambiente. Os sensores são responsáveis por captar dados essenciais como luz, temperatura, umidade e outros, em tempo real, enquanto os atuadores executam ações em resposta a essas informações, gerando

uma perspectiva geral do ambiente. Essa camada serve como ponto de entrada para os dados, representando o elo entre o mundo físico e o ambiente digital.

- Camada de Transmissão (ou Camada de Rede): responsável pela comunicação e transmissão eficiente dos dados coletados pelos sensores para a camada superior. Estão inclusas nessa camada algumas tecnologias heterogêneas, que realizam métodos de endereçamento, roteamento e permitem o acesso a rede e a entrega dos dados a aplicações e servidores externos.

- Camada de Aplicação (ou Camada de gerenciamento e utilização): onde ocorre o processamento, análise e aplicação dos dados em soluções específicas, adaptadas de acordo com as diferentes necessidades. Esta camada também representa a interface com os usuários e as operações que se beneficiam com as informações provenientes dos dispositivos IoT.

2.1.1 Segurança em Redes IoT

Atualmente, o número de dispositivos que utilizam as redes IoT tem crescido cada vez mais e assegurar a proteção dessas redes tem se tornado mais difícil, levando em conta que existem muitos fatores que podem contribuir com a violação de suas seguranças.

Quando comparada às redes tradicionais, a natureza heterogênea e distribuída dos dispositivos IoT, muitas vezes com recursos limitados de hardware e software, intensifica as vulnerabilidades e dificulta a implementação de medidas de segurança robustas [15]. Além disso, a diversidade de padrões de comunicação e protocolos em ambientes IoT contribui para a complexidade, tornando a interoperabilidade entre dispositivos e a aplicação de políticas de segurança coesas uma tarefa desafiadora [16, 17]. A vasta quantidade de dados gerados por dispositivos IoT também amplia os riscos de privacidade e exposição a ameaças cibernéticas, demandando estratégias eficientes de gerenciamento e proteção [18].

De acordo com a comunidade aberta voltada para a segurança de softwares, a *Open Worldwide Application Security Project* (OWASP)[1], os dez principais problemas de segurança, em IoT, que podemos citar são:

1. Senhas fracas, adivinháveis ou codificadas: senhas padrões e comuns, disponíveis publicamente e imutáveis, que podem facilmente ser descobertas por meio do uso da força bruta, ou seja, algoritmos que testam todas as senhas até encontrar uma correspondência;

2. Serviços de rede inseguros: serviços de rede que operam no dispositivo e são desnecessários ou vulneráveis, especialmente aqueles que estão conectados à Internet,

[1] https://owasp.org/www-pdf-archive/OWASP-IoT-Top-10-2018-final.pdf

podendo comprometer a confidencialidade, integridade/autenticidade ou disponibilidade das informações;

3. Interfaces inseguras do ecossistema: interfaces web, API de backend, nuvem ou interfaces móveis que são inseguras no ecossistema externo ao dispositivo podem resultar na vulnerabilidade do dispositivo ou de seus componentes associados. Normalmente, os problemas estão associados a ausência de autenticação/autorização, criptografia ausente ou fraca, e a falta de filtragem de entrada e saída;

4. Falta de mecanismo de atualização seguro: abrange a ausência da validação do firmware no próprio dispositivo, da entrega segura (sem criptografia durante a transmissão), a inexistência de mecanismos anti-reversão e de notificações sobre alterações de segurança decorrentes de atualizações;

5. Uso de componentes inseguros ou desatualizados: utilização de componentes ou bibliotecas de software desatualizados ou vulneráveis representa um risco significativo para a segurança do dispositivo. Isso engloba a personalização insegura de plataformas de sistema operacional, assim como a adoção de software de terceiros ou componentes de hardware provenientes de uma cadeia de suprimentos comprometida;

6. Proteção insuficiente da privacidade: a manipulação da informação pessoal do usuário, armazenada no dispositivo ou no ecossistema, ocorre de maneira insegura, inadequada ou sem consentimento;

7. Transferência e armazenamento inseguro de dados: ausência de medidas de criptografia ou de controle de acesso para dados confidenciais em qualquer ponto do sistema, seja em repouso, em trânsito ou durante o processamento;

8. Falta de gerenciamento de dispositivos: a ausência de suporte de segurança em dispositivos implementados na produção, abrangendo áreas como gerenciamento de ativos, atualizações, desativação segura, monitoramento de sistemas e capacidades de resposta;

9. Configurações padrão inseguras: dispositivos ou sistemas são entregues com configurações padrão inseguras ou sem a capacidade de serem aprimorados em termos de segurança, uma vez que a modificação das configurações pelos operadores é restringida;

10. Falta de proteção física: ausência de salvaguardas físicas possibilita que potenciais invasores obtenham informações confidenciais, o que pode facilitar futuros ataques remotos ou a tomada de controle local do dispositivo.

2.2 Sistemas de Detecção de Intrusão

Os sistemas de detecção de intrusão (*Intrusion Detection Systems* - IDS) são ferramentas essenciais na segurança da informação, projetadas para identificar atividades suspeitas ou maliciosas em redes ou sistemas. Eles podem ser predominantemente um *software* ou um *hardware*, ou uma combinação de ambos e desempenham um papel crucial na detecção e prevenção de intrusões, auxiliando na proteção, manutenção da integridade, confidencialidade e disponibilidade dos sistemas [19].

A arquitetura geral de um IDS inclui os seguintes componentes [19, 20]:

- Sensores: responsáveis pela coleta de dados
- Mecanismo de Análise: processa os dados coletados pelos sensores e identifica padrões suspeitos ou maliciosos.
- Base de Conhecimento: contém informações sobre padrões de ataques conhecidos e perfis de comportamento típicos. Essa base é usada para comparar os padrões identificados durante a análise.
- Resposta a Incidentes: após a detecção de um ataque, o sistema pode desencadear uma resposta, como bloquear o acesso do usuário ou realizar o envio de alertas.

Um IDS pode ser classificado em *Network-based Intrusion Detection Systems* (NIDS) ou em *Host-based Intrusion Detection Systems* (HIDS). No primeiro, a implantação ocorre no perímetro da rede e os pacotes que a atravessam são examinados em tempo real, em busca de atividades maliciosas. Já no segundo, a instalação é feita diretamente nos hospedeiros (computadores ou servidores) e monitoram as atividades e eventos que ocorrem no próprio sistema operacional, incluindo chamadas de sistema, processos em execução, entre outros. [20]

Para determinar se um ataque está de fato ocorrendo, os padrões de tráfego, atividade ou código podem ser comparados com uma base de conhecimento que contém assinaturas conhecidas de ataques previamente identificados. Esse método é conhecido como baseado em assinatura (*signature-based*) e se mostra altamente eficaz para detectar ataques bem conhecidos, visto que ao encontrar uma correspondência com a base, o evento é considerado uma intrusão [21, 20].

Em contrapartida, outra possível abordagem é o método baseado em anomalias (*anomaly based*), mais eficiente na identificação de ataques menos conhecidos, uma vez que envolve a criação de um perfil do comportamento normal do sistema ou usuário. Qualquer desvio significativo desse padrão é considerado uma anomalia e pode indicar uma possível atividade maliciosa. [21, 20].

2.3 Mineração de Fluxos Contínuos de Dados

A mineração de dados é um campo da ciência de dados que lida com a análise e extração de padrões e informações relevantes a partir de um grande conjunto de dados. Este campo multidisciplinar integra técnicas de estatística, aprendizado de máquina e inteligência artificial para explorar e descobrir *insights* ocultos nos dados [22].

O processo de mineração de dados compreende diversas etapas, começando com a compreensão do domínio e a seleção dos dados relevantes, passando pelo pré-processamento para tratar ruídos e valores ausentes, até a aplicação de algoritmos de mineração para identificar padrões e estruturas nos dados [22, 23].

A mineração de dados, tradicionalmente centrada em conjuntos de dados estáticos, trabalha com dados em diferentes formatos, como texto, imagem, som, e mais recentemente, fluxos contínuos de dados em tempo real. Sendo assim, enfrenta novos desafios quando aplicada a dados não estáticos, ou seja os fluxos contínuos, exigindo adaptações e inovações nas técnicas existentes.

Normalmente são utilizados algoritmos de aprendizado em lote, ou *Batch Learning Model* (BLM), que refere-se a uma abordagem de treinamento em que os dados são estáticos, coletados ao longo do tempo e periodicamente utilizados em lotes para treinar o modelo. O processo de treinamento em lote não permite uma aprendizagem gradual, pois o sistema é treinado offline e, portanto, em caso de novos dados, é necessário retreiná-lo. Essa técnica demanda uma quantidade significativa de tempo e recursos, incluindo CPU, RAM, espaço em disco, devido ao volume acumulado de dados, sendo menos adequado para sistemas que precisam responder a dados em rápida mudança, como em detecção de intrusões [24].

Já os fluxos contínuos de dados representam uma classe especial de dados caracterizada por sua natureza dinâmica e características únicas, portanto os algoritmos devem ser capazes de lidar com dados que estão em constante evolução, chegando em grande quantidade, em alta velocidade e de forma contínua [25, 26, 27].

Diferentemente de abordagens tradicionais, a mineração de fluxos contínuos de dados opera de maneira eficiente, adaptando-se à dinâmica rápida e contínua dos dados. Esses fluxos, provenientes de diversas fontes heterogêneas, desafiam as técnicas convencionais, requerendo algoritmos sofisticados para lidar com a diversidade de tipos de dados e também com a análise, processamento e extração de informações úteis em tempo real, permitindo a compreensão de padrões subjacentes e a tomada de decisões rápidas e eficazes [24].

Nesse cenário, os modelos e algoritmos são projetados e possuem algumas características mais específicas como [28]:

- Aprendizado Incremental: cada nova amostra é processada de forma incremental e, em seguida, é realizado o aprendizado e a atualização de suas estatísticas e parâmetros;
- Descarte ou Retenção Limitada: considerando o alto e potencialmente infinito volume de dados, após o processamento de uma amostra, dependendo da aplicação, a amostra pode ser descartada para economizar recursos ou pode ser armazenada em um histórico limitado para análises futuras ou para a atualização de modelos;
- Adaptabilidade e Mudança de Conceito: deve ser capaz de se adaptar a mudanças nas características dos dados ao longo do tempo, conhecidas como mudanças de conceito.

Existem três principais técnicas relacionadas ao aprendizado de máquina: o supervisionado, o não-supervisionado e o semi-supervisionado.

No aprendizado supervisionado, os modelos são treinados utilizando um conjunto de dados rotulados, onde as entradas e suas correspondentes saídas desejadas são fornecidas. Essa técnica é frequentemente empregada em tarefas de classificação, na qual é possível categorizar novos dados em classes predefinidas, e regressão, sendo útil para prever respostas contínuas e valores numéricos [29, 30].

Por outro lado, o aprendizado não supervisionado é uma técnica em que o modelo é treinado em um conjunto de dados desprovido de rótulos associados [29]. O modelo identifica padrões e estruturas nos dados por conta própria, sem orientação externa, buscando agrupar instâncias semelhantes ou reduzir a dimensionalidade dos dados [30]. Sendo assim, é valioso quando se utiliza abordagens de agrupamento (ou *Clustering*), no qual os dados são agrupados em conjuntos distintos, e os membros de cada grupo compartilham características comuns.

Por fim, no aprendizado semi-supervisionado, os modelos são treinados em um conjunto de dados que combina exemplos rotulados e não rotulados. Esta abordagem visa aproveitar ao máximo a informação disponível, especialmente quando a obtenção de rótulos para todos os dados é custosa ou impraticável [31].

2.3.1 One-Class Support Vector Machine

Uma abordagem muito comum para a detecção de anomalias é o algoritmo *One-class Support Vector Machine* (OCSVM), proposto pela primeira vez em 1999 por Scholkopf [32]. O OCSVM é uma extensão do algoritmo SVM, de aprendizado supervisionado, proposto em 1995 por Vapnik [33], no qual é realizado o uso de funções discriminantes que serão responsáveis por separar os dados em duas classses distintas, fazendo o uso de um hiperplano e levando em conta o conjunto de dados de treinamento fornecido [34].

No entanto, o SVM apresenta algumas limitações quando aplicado a situações que representam o mundo real. O algoritmo é notavelmente eficaz em cenários de classificação binária convencional, mas sua aplicação em conjuntos de dados não convencionais, onde as classes são desproporcionalmente representadas, pode resultar em modelos direcionados à classe majoritária [35]. Além disso, por ser um algoritmo de classificação binária, o mesmo assume que durante a fase de treinamento estarão disponíveis exemplos de ambas as classes, quando na realidade, isso nem sempre irá ocorrer.

Já o OCSVM também faz a utilização de hiperplano, porém sua função é encontrar uma fronteira de decisão que envolva a maior parte dos dados [36]. Sendo assim, durante o treinamento, é utilizada apenas a classe com o comportamento esperado, ou seja, o algoritmo aprende e define situações normais, e tudo aquilo que não estiver dentro da aréa delimitada pela fronteira, é considerado um comportamento anômalo.

O emprego do OCSVM apresenta flexibilidade ao se adaptar a diferentes modos de processamento de dados: em lote (*batch*) e em fluxo (*stream*). Estes dois cenários refletem distintas abordagens no treinamento e na aplicação do modelo e a escolha da melhor opção depende da natureza temporal dos dados disponíveis e dos requisitos de atualização do modelo, respondendo assim, às necessidades específicas de ambientes estáticos e dinâmicos.

O processamento em lote é empregado quando o conjunto de dados está completamente disponível no início do treinamento e avaliação do modelo. Nesse contexto, todo esse conjunto de dados é utilizado para treinar o modelo, permitindo que o OCSVM identifique padrões representativos da classe normal. O treinamento em lote é adequado para cenários em que os dados são estáticos, não sofrendo alterações frequentes, e onde a atualização do modelo pode ser realizada de forma assíncrona ao processamento dos dados.

Em contrapartida, a aplicação do OCSVM em processamento em fluxo é apropriada para cenários dinâmicos nos quais os dados são recebidos sequencialmente ao longo do tempo, exigindo que o modelo seja adaptável a mudanças graduais ou abruptas nos padrões dos dados. Essa abordagem é valiosa em situações em que a atualização contínua do modelo é necessária para manter a relevância em um ambiente em constante evolução.

Sendo assim, o OCSVM emerge como uma alternativa promissora para a detecção de intrusões, principalmente devido à sua capacidade de modelar padrões em conjuntos de dados desbalanceados, sua eficácia na identificação de instâncias anômalas e sua capacidade de criar um limite de decisão em torno da classe normal, tornando-o resiliente a ataques adversários e a variações nas características das instâncias anômalas.

2.4 Trabalhos Correlatos

A detecção de ataques em ambientes computacionais é uma área de pesquisa crucial à medida que a complexidade e a sofisticação das ameaças cibernéticas continuam a evoluir. Por isso, na literatura, são apresentadas diversas técnicas para realizar a detecção de ataques e anomalias em redes.

Vu et al. [37], propuseram o uso de Deep Transfer Learning (DTL) em um modelo baseado em Autoencoders (AE) para transferir conhecimento entre domínios de dados, permitindo a detecção eficaz de ataques em domínios sem informações de rótulo. Os experimentos realizados demonstraram que o modelo proposto, denominado MMD-AE, superou outros métodos de detecção de ataques, melhorando significativamente a Área Sob a Curva (AUC) na detecção de ataques em redes IoT.

Abordando a crescente complexidade dos modelos de IoT, Hasan et al. [38] utilizaram técnicas de aprendizado de máquina, como Regressão Logística, Máquina de Vetores de Suporte e Árvore de Decisão, para prever com precisão ataques e anomalias em sistemas de IoT. Eles compararam o desempenho desses algoritmos e demonstraram que modelos como Árvore de Decisão e Floresta Aleatória apresentaram maior precisão na detecção de anomalias.

Utilizando o aprendizado federado e redes neurais recorrentes (LSTM e GRU) para modelar o comportamento normal dos dispositivos IoT e detectar anomalias em tempo real, Mothukuri et al. [39] visaram também preservar a privacidade dos dados dos dispositivos IoT. Assim, os modelos foram treinados localmente em cada dispositivo e, em seguida, agregados em um modelo global, sem compartilhar os dados brutos. Os resultados experimentais mostraram que a abordagem proposta superou os métodos de detecção de anomalias centralizadas em termos de precisão e minimização de alarmes falsos. Além disso, a integração do aprendizado federado com um *ensemble* de modelos melhorou ainda mais a precisão da detecção de anomalias.

Já Bhunia et al. [40] apresentam o framework *SoftThings*, que utiliza técnicas de SDN para detectar e mitigar ameaças de segurança em dispositivos IoT de forma dinâmica e adaptativa. O método envolve o uso de algoritmos de aprendizado de máquina para monitorar e aprender o comportamento dos dispositivos IoT ao longo do tempo, permitindo a detecção precoce de tráfego anômalo e a mitigação de ataques. Os resultados preliminares dos experimentos realizados no emulador Mininet mostram que o *SoftThings* é capaz de detectar ataques com cerca de 98% de precisão e mitigar os fluxos subsequentes bloqueando ou limitando a taxa em poucos segundos.

Outro framework, trazido por Rathore et al. [41], envolve a detecção de ataques distribuídos baseada em aprendizado semi-supervisionado para IoT. O método proposto utiliza um algoritmo ESFCM (Semi-Supervised Fuzzy C-Means) que combina o algoritmo

ELM (Extreme Learning Machine) com o algoritmo SFCM (Fuzzy C-Means) para detecção em tempo real. Os resultados da avaliação experimental no conjunto de dados NSL KDD demonstram que o framework alcançou um ótimo desempenho, com um tempo de detecção de ataque de 11 ms e uma taxa de precisão de 86,53%.

Em um contexto mais geral, isto é, não apenas voltado para a IoT, o trabalho apresentado por Jha e Ragha [42] aborda a utilização de Support Vector Machine em Sistemas de Detecção de Intrusão. O estudo propõe um modelo de Aprendizado de Máquina que combina os benefícios de aprendizado supervisionado e não supervisionado, utilizando uma versão modificada do SVM. Além disso, é fornecido um processo preliminar de seleção de características usando Algoritmos Genéticos (GA) para selecionar campos de pacotes mais apropriados. O método proposto foi testado no conjunto de dados KDD 99, demonstrando a eficácia na detecção de anomalias de rede.

Parveen et al. [43] apresentam um algoritmo de mineração de fluxo baseado em conjunto para a detecção de ameaças internas. O algoritmo aborda o desafio de identificar anomalias raras em contextos nos quais comportamentos em evolução tendem a mascara-las. Para isso, são utilizados métodos de aprendizado supervisionado, incluindo o One-Class Support Vector Machine e a abordagem de *ensemble*. Os resultados obtidos demonstram que o classificador desenvolvido exibe uma precisão de classificação substancialmente aumentada para fluxos reais de ameaças internas em comparação com abordagens tradicionais de aprendizado supervisionado e outros métodos de modelo único, destacando-se pela sua capacidade de lidar com comportamentos evolutivos e anomalias raras.

Ainda nesse contexto, com estratégias inovadoras de aprendizado ativo para lidar com a detecção de mudanças de conceito em fluxos de dados, Krawczyk et al. [44], abordam a necessidade de adaptação rápida à natureza evolutiva desses fluxos. O estudo apresentou um framework para aprendizado ativo em cenários de aprendizado online a partir de fluxos de dados com mudanças de conceito, utilizando o detector de mudanças ADWIN2 devido à sua eficiência com baixa complexidade computacional. Além disso, foram propostas três estratégias de aprendizado ativo, baseadas na incerteza do modelo, na alocação dinâmica de recursos ao longo do tempo e na randomização do espaço de busca. Os resultados obtidos demonstraram a eficácia dessas estratégias, mostrando que elas foram capazes de obter rótulos para objetos provenientes de distribuições evoluídas, resultando em uma classificação precisa do fluxo de dados.

Por fim, Winter et al. [45] propuseram um sistema de detecção de intrusão em redes que opera com fluxos de rede e utiliza Máquinas de Vetores de Suporte de Uma Classe. Em contraste com os sistemas tradicionais de detecção de anomalias, o sistema é treinado apenas com dados maliciosos, visando reconhecer ataques previamente aprendidos, incluindo variações de ataques, em vez de detectar anomalias. O estudo realizou

uma avaliação que resultou em uma taxa de alarmes falsos de 0% e uma taxa de erro de detecção de 2%. Esses resultados indicam que a abordagem proposta é promissora para pesquisas futuras e complementa os sistemas tradicionais de detecção de intrusões baseados em assinaturas.

3 MATERIAIS E MÉTODOS

4 RESULTADOS

5 CONCLUSÃO

REFERÊNCIAS

[1] BORGIA, E. The internet of things vision: Key features, applications and open issues. *Computer Communications*, v. 54, p. 1–31, 2014. ISSN 0140-3664. Disponível em: <https://www.sciencedirect.com/science/article/pii/S0140366414003168>.

[2] ABOMHARA, M.; KøIEN, G. M. Security and privacy in the internet of things: Current status and open issues. In: *2014 International Conference on Privacy and Security in Mobile Systems (PRISMS)*. [S.l.: s.n.], 2014. p. 1–8.

[3] FRUSTACI, M. et al. Evaluating critical security issues of the iot world: Present and future challenges. *IEEE Internet of Things Journal*, v. 5, n. 4, p. 2483–2495, 2018.

[4] MAHMOUD, R. et al. Internet of things (iot) security: Current status, challenges and prospective measures. In: *2015 10th International Conference for Internet Technology and Secured Transactions (ICITST)*. [S.l.: s.n.], 2015. p. 336–341.

[5] ZHANG, Z.-K. et al. Iot security: Ongoing challenges and research opportunities. In: *2014 IEEE 7th International Conference on Service-Oriented Computing and Applications*. [S.l.: s.n.], 2014. p. 230–234.

[6] LIU, Z. et al. Anomaly detection on iot network intrusion using machine learning. In: *2020 International Conference on Artificial Intelligence, Big Data, Computing and Data Communication Systems (icABCD)*. [S.l.: s.n.], 2020. p. 1–5.

[7] MANIRIHO, P. et al. Anomaly-based intrusion detection approach for iot networks using machine learning. In: *2020 International Conference on Computer Engineering, Network, and Intelligent Multimedia (CENIM)*. [S.l.: s.n.], 2020. p. 303–308.

[8] SUSILO, B.; SARI, R. F. Intrusion detection in software defined network using deep learning approach. In: *2021 IEEE 11th Annual Computing and Communication Workshop and Conference (CCWC)*. [S.l.: s.n.], 2021. p. 0807–0812.

[9] DITZLER, G. et al. Learning in nonstationary environments: A survey. *IEEE Computational Intelligence Magazine*, v. 10, n. 4, p. 12–25, 2015.

[10] L'HEUREUX, A. et al. Machine learning with big data: Challenges and approaches. *IEEE Access*, v. 5, p. 7776–7797, 2017.

[11] ARBEX, G. V. et al. Iot ddos detection based on stream learning. In: *2021 12th International Conference on Network of the Future (NoF)*. [S.l.: s.n.], 2021. p. 1–8.

[12] AZUMAH, S. W. et al. A deep lstm based approach for intrusion detection iot devices network in smart home. In: *2021 IEEE 7th World Forum on Internet of Things (WF-IoT)*. [S.l.: s.n.], 2021. p. 836–841.

[13] NAKAGAWA, F. H. Y.; JUNIOR, S. B.; ZARPELãO, B. B. Attack detection in smart home iot networks using clustream and page-hinkley test. In: *2021 IEEE Latin-American Conference on Communications (LATINCOM)*. [S.l.: s.n.], 2021. p. 1–6.

[14] ASHTON, K. et al. That 'internet of things' thing. *RFID journal*, Hauppauge, New York, v. 22, n. 7, p. 97–114, 2009.

[15] ATZORI, L.; IERA, A.; MORABITO, G. The internet of things: A survey. *Computer Networks*, v. 54, n. 15, p. 2787–2805, 2010. ISSN 1389-1286. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1389128610001568>.

[16] ROMAN, R. et al. Key management systems for sensor networks in the context of the internet of things. *Computers & Electrical Engineering*, Elsevier, v. 37, n. 2, p. 147–159, 2011.

[17] VERMESAN, O.; FRIESS, P. *Internet of things applications-from research and innovation to market deployment*. [S.l.]: Taylor & Francis, 2014.

[18] FERNÁNDEZ-CARAMÉS, T. M.; FRAGA-LAMAS, P. A review on the use of blockchain for the internet of things. *Ieee Access*, IEEE, v. 6, p. 32979–33001, 2018.

[19] LAZAREVIC, A.; KUMAR, V.; SRIVASTAVA, J. Intrusion detection: A survey. In: _____. [S.l.: s.n.], 2005. v. 5, p. 19–78. ISBN 0-387-24226-0.

[20] ZARPELãO, B. B. et al. A survey of intrusion detection in internet of things. *Journal of Network and Computer Applications*, v. 84, p. 25–37, 2017. ISSN 1084-8045. Disponível em: <https://www.sciencedirect.com/science/article/pii/S1084804517300802>.

[21] AXELSSON, S. Intrusion detection systems: A survey and taxonomy. Citeseer, 2000.

[22] HAN, J.; KAMBER, M.; PEI, J. Data mining concepts and techniques third edition. *University of Illinois at Urbana-Champaign Micheline Kamber Jian Pei Simon Fraser University*, 2012.

[23] FAYYAD, U.; PIATETSKY-SHAPIRO, G.; SMYTH, P. From data mining to knowledge discovery in databases. *AI magazine*, v. 17, n. 3, p. 37–37, 1996.

[24] ADEWOLE, K. S. et al. Empirical analysis of data streaming and batch learning models for network intrusion detection. *Electronics*, MDPI, v. 11, n. 19, p. 3109, 2022.

[25] HULTEN, G.; SPENCER, L.; DOMINGOS, P. Mining time-changing data streams. In: . New York, NY, USA: Association for Computing Machinery, 2001. ISBN 158113391X. Disponível em: <https://doi.org/10.1145/502512.502529>.

[26] GABER, M. M.; ZASLAVSKY, A.; KRISHNASWAMY, S. Mining data streams: A review. Association for Computing Machinery, New York, NY, USA, v. 34, n. 2, 2005. ISSN 0163-5808. Disponível em: <https://doi.org/10.1145/1083784.1083789>.

[27] LESKOVEC ANAND RAJARAMAN, J. U. J. *Mining of Massive Datasets*. [S.l.]: Cambridge University Press, 2021.

[28] DOMINGOS, P.; HULTEN, G. Mining high-speed data streams. In: *Proceedings of the sixth ACM SIGKDD international conference on Knowledge discovery and data mining*. [S.l.: s.n.], 2000. p. 71–80.

[29] BISHOP, C. M.; NASRABADI, N. M. *Pattern recognition and machine learning*. [S.l.]: Springer, 2006. v. 4.

[30] HASTIE, T. et al. *The elements of statistical learning: data mining, inference, and prediction*. [S.l.]: Springer, 2009. v. 2.

[31] ENGELEN, J. E. V.; HOOS, H. H. A survey on semi-supervised learning. *Machine learning*, Springer, v. 109, n. 2, p. 373–440, 2020.

[32] SCHöLKOPF, B. et al. Support vector method for novelty detection. In: SOLLA, S.; LEEN, T.; MüLLER, K. (Ed.). *Advances in Neural Information Processing Systems*. MIT Press, 1999. v. 12. Disponível em: <https://proceedings.neurips.cc/paper_files/paper/1999/file/8725fb777f25776ffa9076e44fcfd776-Paper.pdf>.

[33] VAPNIK, V. *The nature of statistical learning theory*. [S.l.]: Springer science & business media, 1999.

[34] MAMMONE, A.; TURCHI, M.; CRISTIANINI, N. Support vector machines. *WIREs Computational Statistics*, v. 1, n. 3, p. 283–289, 2009. Disponível em: <https://wires.onlinelibrary.wiley.com/doi/abs/10.1002/wics.49>.

[35] HE, H.; GARCIA, E. A. Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering*, v. 21, n. 9, p. 1263–1284, 2009.

[36] BEZERRA, V. H. Botnet detection in internet of things devices using one-class classification. 2019.

[37] VU, L. et al. Deep transfer learning for iot attack detection. *IEEE Access*, IEEE, v. 8, p. 107335–107344, 2020.

[38] HASAN, M. et al. Attack and anomaly detection in iot sensors in iot sites using machine learning approaches. *Internet of Things*, Elsevier, v. 7, p. 100059, 2019.

[39] MOTHUKURI, V. et al. Federated-learning-based anomaly detection for iot security attacks. *IEEE Internet of Things Journal*, IEEE, v. 9, n. 4, p. 2545–2554, 2021.

[40] BHUNIA, S. S.; GURUSAMY, M. Dynamic attack detection and mitigation in iot using sdn. In: IEEE. *2017 27th International telecommunication networks and applications conference (ITNAC)*. [S.l.], 2017. p. 1–6.

[41] RATHORE, S.; PARK, J. H. Semi-supervised learning based distributed attack detection framework for iot. *Applied Soft Computing*, Elsevier, v. 72, p. 79–89, 2018.

[42] JHA, J.; RAGHA, L. Intrusion detection system using support vector machine. *International Journal of Applied Information Systems (IJAIS)*, v. 3, p. 25–30, 2013.

[43] PARVEEN, P. et al. Supervised learning for insider threat detection using stream mining. In: IEEE. *2011 IEEE 23rd international conference on tools with artificial intelligence*. [S.l.], 2011. p. 1032–1039.

[44] KRAWCZYK, B.; PFAHRINGER, B.; WOŹNIAK, M. Combining active learning with concept drift detection for data stream mining. In: IEEE. *2018 IEEE International Conference on Big Data (Big Data)*. [S.l.], 2018. p. 2239–2244.

[45] WINTER, P.; HERMANN, E.; ZEILINGER, M. Inductive intrusion detection in flow-based network data using one-class support vector machines. In: IEEE. *2011 4th IFIP international conference on new technologies, mobility and security*. [S.l.], 2011. p. 1–5.