Aprendizado não supervisionado para detecção de ataques em Internet das Coisas

Luis Henrique Barbosa Vieira[1], **Bruno Bogaz Zarpelão**[1]

[1]Departamento de Computação – Universidade Estadual de Londrina (UEL)
Caixa Postal 10.011 – CEP 86057-970 – Londrina – PR – Brasil

vieiralhb@gmail.com, brunozarpelao@uel.br

Abstract. *The increasing amount of IoT devices challenges the availability of an effective security. The presence of these devices in the urban and home environment creates a scenario to be exploited by malicious users who decide to perform attacks on IoT systems. With the increase of these devices on the market, more ways of getting around its defenses are developed, thus creating a demand for research and analysis in the area in question, as well as a solution that involves reduced human intervention on the detection of these attacks. Considering this scenario, this project has the objective of studying the use of unsupervised algorithms such as DBSCAN to detect attacks based on header analysis of network packet headers. For such, a system for analyzing network packets extracted from an multi-device IoT environment and obtained through a public dataset will be developed, extracting the most significant fields from each protocol's packet headers.*

Resumo. *A crescente quantidade de dispositivos IoT desafia a disponibilidade de uma segurança eficaz. A presença destes dispositivos em ambiente domiciliar e urbano cria um cenário a ser explorado por usuários maliciosos que decidam por realizar ataques a sistemas de Internet das Coisas. Com o aumento destes dispositivos no mercado, mais formas de contornar suas defesas são desenvolvidas, criando assim uma demanda por investigação e análise na área em questão, bem como uma solução que envolva reduzida intervenção humana na detecção destes ataques. Considerando esse cenário, o projeto tem como objetivo estudar a aplicação de algoritmos não supervisionados como o DBSCAN na detecção de ataques realizada a partir da análise dos cabeçalhos dos pacotes coletados na rede. Para tal, será criado um sistema para a análise dos pacotes de rede extraídos de um ambiente IoT com vários dispositivos e obtidos através de um conjunto de dados público, extraindo os campos mais significativos dos cabeçalhos dos pacotes de cada protocolo.*

1. Introdução

Dispositivos inteligentes de Internet das Coisas estão tornando-se cada vez mais difundidos nas diversas áreas do espaço urbano e rural. Seja no controle da agricultura e pecuária até a infraestrutura, logística e produção de uma grande corporação ou segurança de um espaço domiciliar, a ideia de integrar o acesso à Internet em dispositivos capazes de oferecer controle remoto ou enviar dados úteis para a observação e criação de relatórios tem se tornado cada vez mais atrativa [17]. Porém, ainda que apresentem um potencial

de melhoria aos vários setores nos quais podem ser implementados, estes dispositivos ainda contam, até então, com uma segurança que necessita ser estudada e desenvolvida, o que certamente é um convite para usuários que queiram explorar brechas para atividades maliciosas [5].

Analisando de forma profunda a ideia de controlar remotamente ou enviar dados sobre algumas áreas como a de infraestrutura ou domiciliar, não é difícil encontrar um ponto no qual seja crítica a interrupção do serviço ou captura das informações. Por exemplo, um dispositivo que controle a fechadura de uma casa e registre dados sobre a entrada dos moradores cadastrados, se tiver seu funcionamento interrompido ou o envio de dados interceptado pode fornecer um acesso ao interior que não seja avisado ou informações sobre os horários de entrada e saída dos moradores, respectivamente [5]. Tendo isso em vista, fica claro o fato de que a pesquisa por segurança e detecção de ataques em Internet das Coisas torna-se importante, observada a gradativa introdução desta tecnologia no mercado.

Existe a possibilidade de detecção de ataques com o uso de aprendizado de máquina supervisionado, como visto, por exemplo, no trabalho de Belavagi e Muniyal [4], em que a técnica de *Random Forest* é usada obtendo-se uma precisão de 99% na detecção de intrusões. Esta técnica mostra-se eficaz também no trabalho de Stevanovic e Pedersen [16], que a classifica como promissora, precisa e oportuna. Porém, o processo de treinamento nos trabalhos citados foi realizado em computadores com *hardware* superior ao encontrado em dispositivos IoT, tornando muito difícil o aprendizado supervisionado neste caso, visto o grande volume de pacotes de rede. Além disso, temos o problema de que essas técnicas exigem que façamos um conjunto de dados de treinamento, no qual classificamos cada pacote como legítimo ou malicioso, tornando o processo de construção deste conjunto de dados trabalhoso e suscetível a erro, sendo difícil também colocar exemplos de todos os tipos de ataques nestes conjuntos de treinamento, o que pode levar o sistema a não detectar alguns deles depois.

Logo, o aprendizado de máquina não supervisionado foi escolhido como uma alternativa para contornar estes problemas, já que esses algoritmos não exigem um treinamento prévio. Nesses algoritmos, os dados são agrupados de acordo com suas similaridades, o que permite identificar diferentes padrões de comportamento e detectar situações de interesse com mais facilidade.

Considerando este contexto, neste projeto pretende-se a construção de um sistema de detecção de ataques utilizando o algoritmo de aprendizado de máquina não supervisionado DBSCAN para a clusterização dos dados juntamente a um algoritmo de detecção de anomalias Page-Hinkley para detectar ataques em redes IoT a partir da análise dos cabeçalhos de pacotes capturados na rede. Preocupa-se também em trabalhar o DBSCAN com janelas de período fixo, de forma que a clusterização seja renovada conforme a entrada de novos dados, visto que o comportamento dos elementos monitorados muda ao longo do tempo. Ao final, com os resultados obtidos através dos testes, espera-se poder verificar a eficácia do aprendizado de máquina não supervisionado para a detecção de ataques a redes de IoT.

2. Fundamentação Teórico-Metodológica e Estado da Arte

2.1. Internet of Things

O termo Internet of Things foi primeiramente usado por Kevin Ashton em 1999 enquanto trabalhava na Procter & Gamble (P&G) em uma tentativa de atrair a atenção da gerência para uma nova tecnologia de identificação automática por radiofrequência conhecida como RFID. Segundo o autor em [3], como o tópico Internet estava em alta naquele período, criou o termo em questão junto à palavra "coisas" e, visto que trabalhava no processo de optimização do *supply chain* da empresa, acreditou que isso certamente geraria uma boa impressão nos espectadores de sua apresentação.

Por mais que o termo tenha sido criado em 1999, é apenas recentemente que a popularidade no seu uso tem aumentado de forma massiva [7]. O progresso na aplicação e na tentativa de estabelecer uma definição sólida para o IoT acompanhou de perto os avanços na tecnologia de transmissão sem fio e o surgimento cada vez maior dos dispositivos inteligentes no mercado [7], o que é uma possível explicação para a diferença de tempo entre o surgimento e a disseminação em trabalhos e discussões.

Ainda que o termo seja recente, a ideia de interligar aparelhos é algo que acompanha a área de tecnologia e inovação há muito tempo [13]. Mais especificamente, podemos dizer que a raiz para este desígnio é a própria disseminação da Internet ao público em geral, bem como quando começaram a surgir as primeiras redes de acesso local que interligavam os computadores de uma empresa ou universidade, por exemplo [7].

Assim que o termo se popularizou, várias definições foram surgindo através da interpretação de cada autor. Em meio a sensores que reportam continuamente estatísticas, possibilidade de controle remoto e interconectividade entre aparelhos, sejam domésticos ou não, um leitor pode acabar adquirindo um sentimento de confusão ao inicialmente tentar compreender o conceito de IoT [13]. Mesmo que o problema com a definição seja real, o que se pode afirmar que há em comum entre todas é que se deseja integrar o mundo real ao virtual [8] [17], explorando cada vez mais a capacidade de criar uma rede de objetos que antes nem mesmo imaginávamos que futuramente poderiam fornecer tal tipo de interação [7].

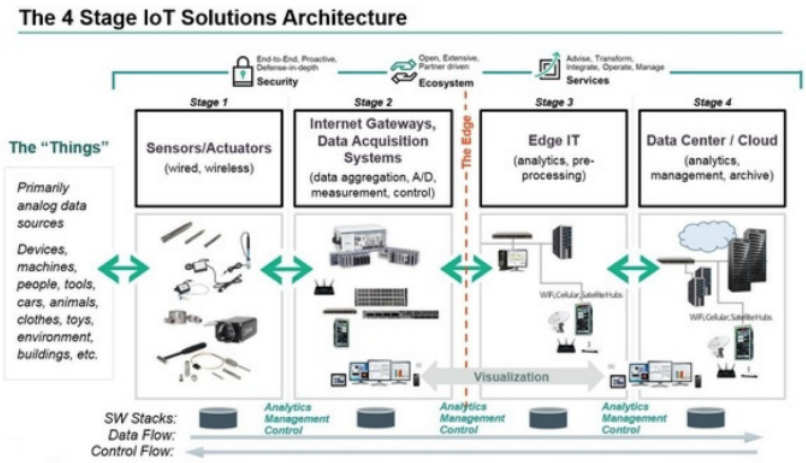

Figura 1. Estágios da arquitetura de uma solução IoT

Assumindo isso, acabamos elevando o espectro do IoT a uma ampla gama de aparelhos e dispositivos que poderiam ser contemplados com esta solução. Para termos uma melhor compreensão de onde tudo isso culminaria, a Figura 1 oferece uma perspectiva sobre a arquitetura separada em estágios de uma solução IoT. Nela, é possível perceber um processo que inicia no que classificamos como as "coisas", que podem ser todos os dispositivos, máquinas, animais, construções entre outras fontes das quais podemos extrair alguma informação. Para tal, seria necessária a fixação de sensores ou atuadores que transmitam as informações com ou sem fio, dependendo do que se deseja conseguir como dados. Seguindo a sequência, percebe-se que a medida em que avançamos entre os estágios temos um fluxo de dados, e que quando retrocedemos temos um fluxo de controle, nos oferecendo uma clara visão de que o resultado de uma aplicação IoT, no geral, seria dispor estatísticas sobre as informações coletadas e o controle do que é observado.

2.2. DBSCAN

O DBSCAN (*Density-based spatial clustering of applications with noise*) foi inicialmente proposto no trabalho de Ester et al. [6] como um algoritmo de clusterização desenvolvido para ser capaz de encontrar agrupamentos e ruídos em um conjunto de dados espacial.

Em síntese, os algoritmos de clusterização tem por objetivo primário detectar a similaridade entre elementos de um conjunto de dados e então agrupá-los de uma forma na qual não seja necessário informar previamente quais são estes grupos ou as características que determinam se um elemento deve ser incluído ou não. Segundo Stein e Busch [15] e tendo em vista este objetivo inicial dos algoritmos de clusterização, a diferença entre os diversos algoritmos desta categoria dá-se pela estratégia adotada para a detecção e formação dos *clusters* com os elementos do conjunto informado, aspirando sempre maximizar a similaridade entre membros de um mesmo *cluster* e a diferença com os de fora.

Dentre as classificações de algoritmos de clusterização, temos os conhecidos por *Density-Based Algorithms*. O conceito principal destes algoritmos é, tendo recebido um valor k e outro ε referente ao raio de uma vizinhança, encontrar agrupamentos que possuam no mínimo k elementos alcançáveis diretamente, indiretamente ou por conexão no raio ε através da distância euclidiana, podendo estes elementos serem classificados entre *directly density-recheable*, *density-recheable* ou *density-connected* entre si.

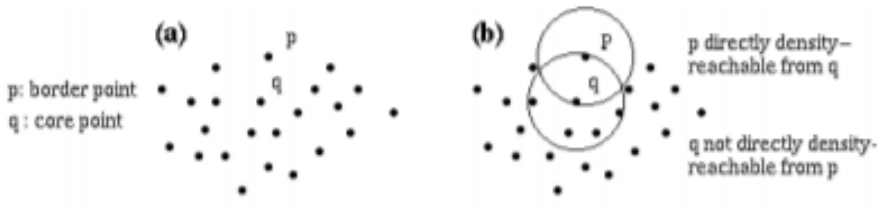

Figura 2. Exemplo sobre *directly-density reachable* Fonte: [6]

É possível observar na Figura 2 um exemplo para a definição de um elemento como diretamente alcançável. Tendo em vista que a circunferência maior ao redor dos pontos representa o raio ε informado e que os pontos representam os dados de um conjunto de dados, percebe-se que o ponto p é classificado como diretamente alcançável devido ao fato de estar no interior da vizinhança do ponto *core* q, porém o inverso não é

verdadeiro, visto que isso causaria assimetria em decorrência da forma na qual os pontos foram marcados (*border* e *core*).

Para que um ponto q seja classificado como *core*, é necessário que a condição k de tamanho mínimo da vizinhança seja satisfeita, com o próprio q fazendo parte desta contagem. Já para a classificação do ponto como *border*, esta condição não precisa ser seguida, visto que existe uma grande possibilidade de que os pontos da fronteira de um *cluster* não tenham uma vizinhança tão numerosa quantos os que estão mais ao interior. Por exemplo, se tivéssemos uma cadeia de pontos que fosse de um ponto *core* q até um ponto fronteiriço p poderíamos afirmar que todos os pontos que levassem até p seriam também *core* por satisfazerem a condição de vizinhança mínima, ou seja, conectariam o ponto q ao p por densidade. Mas o inverso não pode ser afirmado, visto que, como dito anteriormente, os pontos na fronteira são os primeiros a não conseguir satisfazer esta condição de vizinhança. Portanto se tentássemos começar uma cadeia que fosse de p até q, ou seja, o inverso, existe uma grande possibilidade de que já no começo dela em p não conseguiríamos satisfazer a condição de densidade mínima, e assim nem mesmo teríamos uma cadeia. Esta, então, seria a noção de assimetria.

De acordo com Ester et al. [6], esta diferenciação entre *core* e *border points* como também a definição de assimetria são necessárias visto a menor quantidade de vizinhos encontrados na fronteira de um *cluster*. Logo, se todos os pontos fossem iguais para o algoritmo com apenas o número de vizinhos internos a um raio sendo considerados, haveria casos nos quais os pontos fronteiriços de um *cluster* sempre seriam excluídos.

Na Figura 3, observa-se dois exemplos para *density-reachable* e *density-connected*. No caso (a), nota-se novamente a existência de um ponto q *core* e outro p *border*, desta vez sem possuir qualquer alcance direto entre ambos. Porém, é possível identificar a presença de um alcance indireto entre os pontos citados se levarmos em consideração a direct density-reachability entre q e outro ponto intermediário. O caso (a) ainda apresenta uma situação de assimetria, alerta Ester et al. [6], o que é comparável ao exemplo dado anteriormente, no qual o inverso não é verdadeiro, isto é, de p até q.

Logo, no caso (b) é apresentado um exemplo no qual os pontos p e q são agora considerados *density-connected* em um outro ponto intermediário o pelo qual são ambos *density-reachable*. Percebe-se agora que a relação entre os pontos é simétrica, uma vez que não é vista como uma cadeia de pontos alcançáveis por densidade (o que levaria ao problema discutido anteriormente) mas pela presença de um ponto comum pelo qual são ambos alcançáveis, e isso garante que o inverso de uma relação seja verdadeiro (por exemplo, a relação p e q é *density-connected* e a q e p também).

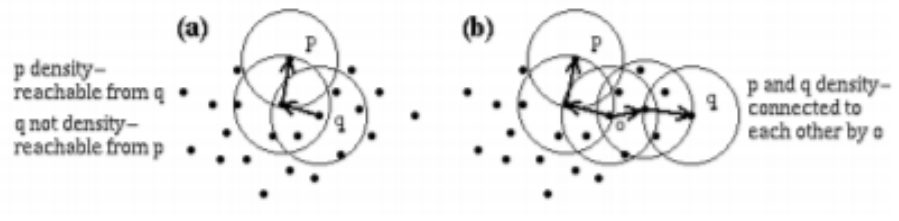

Figura 3. Exemplo sobre *density-reachable* e *density-connected* Fonte: [6]

Tendo em mente estas definições, na Figura 4 é possível verificar através do *clus-*

ter model do DBSCAN como os exemplos anteriores se aplicam no caso deste algoritmo. Com as circunferências representando o raio ε e as setas representando a *density-reachability*, constata-se que os pontos *B* e *C* são *density-connected* devido ao fato de serem ambos *density-reachable* por *A*. Identifica-se também o ponto *N* que por não ser *density-reachable* foi classificado como *noise*.

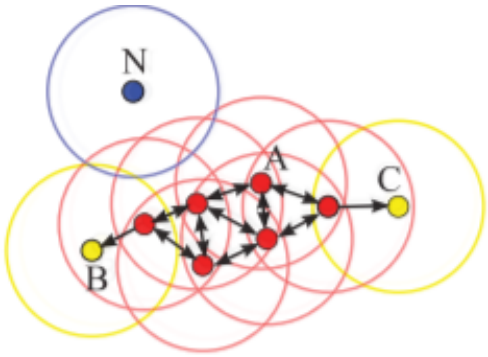

Figura 4. *Cluster model* do DBSCAN Fonte: [14]

De uma forma mais ilustrativa, é possível compreender melhor como ocorreria o processo do DBSCAN e o seu resultado através da Figura 5, na qual dois *clusters* são gerados e dois dados do conjunto marcados como *noise*.

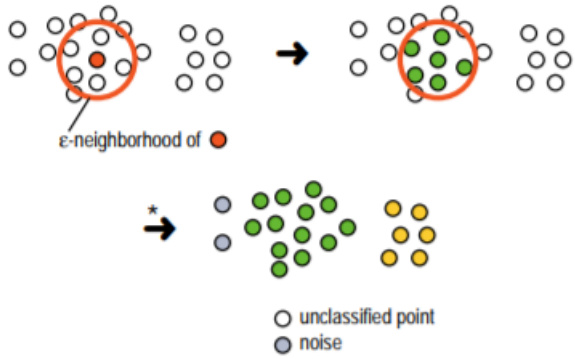

Figura 5. Exemplo ilustrativo do processo do DBSCAN Fonte: [15]

2.3. Page-Hinkley Test

O CUSUM, também citado como *Page-Hinkley Test* [11], é um teste estatístico utilizado para detectar uma mudança abrupta em uma série de dados. No geral, o problema de detectar um ponto de mudança já é um tópico muito discutido e aprofundado, visto sua utilidade em várias áreas [9], tendo o *Page-Hinkley Test* sido proposto como uma técnica para a análise sequencial, na qual o tamanho da amostra não é previamente fixada.

Para o caso do *Page-Hinkley*, vamos considerar uma sequência de valores que serão informados ao teste individualmente, iniciando pelo primeiro elemento. Consideremos também uma variável $\bar{m}$ que representará a média dos valores observados até momento, outra variável s que armazenará a diferença com a média anterior e um valor t

que será o número de amostras que foram incluídas no teste até então. Ao início do teste, as variáveis citadas iniciam com os seguintes valores:

$$s = 0, t = 1, \bar{m} = 0 \tag{1}$$

Considerando que um novo valor v seja inserido no teste, calcula-se uma nova média através da divisão entre $v - \bar{m}$ pela contagem de amostras t e adicionando o resultado ao valor de $\bar{m}$. A média é calculada desta forma para que não sejam necessárias as amostras antigas do teste, o que nos levaria a ter que realizar o cálculo através da divisão da soma de todos os valores anteriores.

Tendo calculado a nova média, adquire-se o valor de s através de uma verificação pelo máximo entre 0 e a soma entre $v - \bar{m} - \delta$ e uma multiplicação de um fator de esquecimento α com o valor atual de s, como é possível observar em (2). Percebe-se que, ao multiplicar α com o valor anterior de s, α atua como um redutor do impacto das observações anteriores quando assume valores menores que 1, assim como o fator δ reduz o impacto do novo valor v no teste.

$$\begin{gathered} \bar{m} = \bar{m} + \frac{v - \bar{m}}{t} \\ s = max(0, \alpha * s + (v - \bar{m} - \delta)) \\ t = t + 1 \end{gathered} \tag{2}$$

Após os cálculos, o teste chega a duas hipóteses, como pode ser visto em (3). Se o resultado s for maior que um limiar λ informado como o mínimo de diferença para a geração do alerta, uma mudança é detectada, como é o caso da hipótese H_1. Nesta situação, as variáveis têm seus valores restaurados iguais aos vistos em (1) e o teste é reiniciado para o primeiro valor que venha a ser inserido no teste.

$$\begin{gathered} H_0 : s < \lambda \\ H_1 : s \geq \lambda \end{gathered} \tag{3}$$

3. Objetivos

O projeto possui a finalidade de criar uma solução que seja capaz de detectar possíveis ataques maliciosos a sistemas de Internet das Coisas (IoT). Logo, será objetivo deste projeto pesquisar a utilização do aprendizado de máquina não supervisionado (*unsupervised machine learning*) para a detecção.

Especificamente, o projeto também busca:

• Verificar a eficácia desses algoritmos, analisando a taxa de detecção de ataques e a taxa de falsos positivos que eles produzem;

• Verificar se há tipos de ataque que não são detectáveis por esses algoritmos ou que oferecem maiores dificuldades.

4. Procedimentos metodológicos/Métodos e técnicas

Tendo em vista o objetivo desta proposta, livros como [10] e artigos como [2] e [5] serão fundamentais para o entendimento das técnicas a serem utilizadas.

Definido o escopo do projeto em questão, o mesmo compreenderá o desenvolvimento de uma solução que utilize do aprendizado de máquina não supervisionado para a detecção de ataques a sistemas de IoT. Feito isso, será analisada sua efetividade em previnir danos a tais redes de dispositivos inteligentes, colhendo os resultados derivados desta de modo a serem posteriormente estudados.

Para os testes e análises a serem realizados sobre o produto final do projeto, considera-se a construção de um ambiente virtual que possibilite uma capacidade de aplicação e observância relativa à que uma rede física e privada concederia. Este difere-se no fato de prover uma alternativa econômica para estudo, visto que a virtualidade do processo descartaria a necessidade de possuir os dispositivos fisicamente.

Seguem as etapas previstas a serem seguidas durante o decorrer do projeto:

Etapa 1: Realizar um levantamento bibliográfico sobre aprendizado de máquina não supervisionado e detecção de intrusão em sistemas IoT;

Etapa 2: Verificar os algoritmos já existentes a fim de inspirar-se na implementação e nas características benéficas de cada um;

Etapa 3: Estudar os datasets a serem utilizados no projeto, retratados nas publicações a seguir:

- Anthi et al. [2], que alegando a dificuldade de detectar atividades maliciosas sem categorizar os acessos normais, classifica o comportamento de oito dispositivos IoT (Belkin NetCam camera, TP-Link NC200 Camera, TP-Link Smart Plug, Samsung Smart Things hub, Amazon Echo Dot, British Gas Hive), a fim de identificar os que diferem-se do habitual. Cita também o desafio de distinguir os tipos de ataques automaticamente, isto é, sem observação humana;
- Parmisano et al. [12], que financiado pela "Avast Software s.r.o." tem como objetivo oferecer um conjunto de dados de larga escala com infecções por malware em dispositivos IoT reais, além de tráfego benigno. Os dados foram coletados dos dispositivos IoT "Philips HUE smart LED lamp", "Amazon Echo home intelligent personal assistant" e "Somfy smart doorlock", todos possuindo hardware real sem qualquer simulação;

Etapa 4: Construção de cenários mais diversos a partir da manipulação dos datasets originais;

Etapa 5: Desenvolver uma solução com aprendizado não supervisionado e aplicá-la aos datasets selecionados para o projeto, utilizando a linguagem de programação "Python", a fim de detectar ataques ao ambiente IoT;

Etapa 6: Observar a eficácia da solução desenvolvida na detecção e os vários detalhes de seu funcionamento nas situações que vierem a ocorrer;

Etapa 7: Redação do TCC.

5. Cronograma de Execução

Atividades:

1. Realizar um levantamento bibliográfico sobre aprendizado de máquina não supervisionado e detecção de intrusão em sistemas IoT;
2. Verificar os algoritmos já existentes a fim de inspirar-se na implementação e nas características benéficas de cada um;
3. Estudar os datasets a serem utilizados no projeto;

4. Construção de cenários mais diversos a partir da manipulação dos datasets originais;
5. Desenvolver uma solução com aprendizado não supervisionado e aplicá-la aos datasets selecionados para o projeto, utilizando a linguagem de programação "Python", a fim de detectar ataques ao ambiente IoT;
6. Observar a eficácia da solução desenvolvida na detecção e os vários detalhes de seu funcionamento nas situações que vierem a ocorrer;
7. Redação do TCC.

Tabela 1. Cronograma de Execução

	set	out	nov	dez	jan	fev	mar	abr	mai	jun
Atividade 1	—	—								
Atividade 2		—								
Atividade 3		—								
Atividade 4			—	—						
Atividade 5				—	—	—	—			
Atividade 6				—	—	—	—			
Atividade 7				—	—	—	—	—	—	—

6. Contribuições e/ou Resultados esperados

O projeto pretende promover uma discussão sobre a viabilidade de utilizar esse tipo de algoritmo para detectar ataques a sistemas IoT, trazendo informações sobre as vantagens e possíveis dificuldades na aplicação. Mais especificamente, pretende-se avaliar a eficácia, trazendo resultados no que diz respeito a taxa de detecção de ataques e falsos positivos, bem como dos tipos de ataques e se são detectáveis por esse tipo de técnica.

Há uma possibilidade de termos 35,8 bilhões de dispositivos de Internet das Coisas no mundo nos próximos anos [1], e o desenvolvimento de uma técnica desse tipo permitiria o aumento da segurança destes dispositivos e da nossa, ao usá-los.

7. Espaço para assinaturas

Londrina, *13 de Setembro de 2021*.

_____ _____

Aluno Orientador

Referências

[1] Tanweer Alam. A reliable communication framework and its use in internet of things (iot). *International Journal of Scientific Research in Computer Science, Engineering and Information Technology (IJSRCSEIT)*, 3(5):450–456, 2018.

[2] Eirini Anthi, Lowri Williams, Małgorzata Słowińska, George Theodorakopoulos, and Pete Burnap. A supervised intrusion detection system for smart home iot devices. *IEEE Internet of Things Journal*, 6(5):9042–9053, 2019.

[3] Kevin Ashton et al. That 'internet of things' thing. *RFID journal*, 22(7):97–114, 2009.

[4] Manjula C Belavagi and Balachandra Muniyal. Performance evaluation of supervised machine learning algorithms for intrusion detection. *Procedia Computer Science*, 89(2016):117–123, 2016.

[5] Vitor Hugo Bezerra, Victor Guilherme Turrisi da Costa, Sylvio Barbon Junior, Rodrigo Sanches Miani, and Bruno Bogaz Zarpelão. Iotds: A one-class classification approach to detect botnets in internet of things devices. *Sensors*, 19(14):3188, 2019.

[6] Martin Ester, Hans-Peter Kriegel, Jörg Sander, Xiaowei Xu, et al. A density-based algorithm for discovering clusters in large spatial databases with noise. In *kdd*, volume 96, pages 226–231, 1996.

[7] Samuel Greengard. *The internet of things*. MIT press, 2021.

[8] Stephan Haller. The things in the internet of things. *Poster at the (IoT 2010). Tokyo, Japan, November*, 5(8):26–30, 2010.

[9] Cédric Hartland, Nicolas Baskiotis, Sylvain Gelly, Michèle Sebag, and Olivier Teytaud. Change point detection and meta-bandits for online learning in dynamic environments. In *CAp 2007: 9è Conférence francophone sur l'apprentissage automatique*, pages 237–250, 2007.

[10] Alboukadel Kassambara. *Practical guide to cluster analysis in R: Unsupervised machine learning*, volume 1. STHDA, 2017.

[11] Ewan S Page. Continuous inspection schemes. *Biometrika*, 41(1/2):100–115, 1954.

[12] A Parmisano, S Garcia, and MJ Erquiaga. A labeled dataset with malicious and benign iot network traffic. *Stratosphere Laboratory: Praha, Czech Republic*, 2020.

[13] Karen Rose, Scott Eldridge, and Lyman Chapin. The internet of things: An overview. *The internet society (ISOC)*, 80:1–50, 2015.

[14] Erich Schubert, Jörg Sander, Martin Ester, Hans Peter Kriegel, and Xiaowei Xu. Dbscan revisited, revisited: why and how you should (still) use dbscan. *ACM Transactions on Database Systems (TODS)*, 42(3):1–21, 2017.

[15] Benno Stein and Michael Busch. Density-based cluster algorithms in low-dimensional and high-dimensional applications. In *International Workshop on Text-Based Information Retrieval*, page 45. Citeseer, 2005.

[16] Matija Stevanovic and Jens Myrup Pedersen. An efficient flow-based botnet detection using supervised machine learning. In *2014 international conference on computing, networking and communications (ICNC)*, pages 797–801. IEEE, 2014.

[17] Bruno Bogaz Zarpelão, Rodrigo Sanches Miani, Cláudio Toshio Kawakani, and Sean Carlisto de Alvarenga. A survey of intrusion detection in internet of things. *Journal of Network and Computer Applications*, 84:25–37, 2017.