Predição de Ataques DDoS pela Correlação de Séries Temporais via Padrões Ordinais

Lucas Albano[1], **Ligia F. Borges**[1], **Anderson B. de Neira**[2], **Michele Nogueira**[1,2]

[1]Departamento de Ciência da Computação - Universidade Federal de Minas Gerais

[2]Departamento de Informática - Universidade Federal do Paraná

{lucasalbano,ligia.borges,michele}@dcc.ufmg.br, abneira@inf.ufpr.br

Resumo. *Os dispositivos infectados na Internet das Coisas (IoT) representam um dos principais desafios no combate aos ataques de negação de serviço distribuído (DDoS). Os atacantes camuflam suas ações e atrasam a predição do ataque, impulsionando a criação de novas soluções resilientes a ruídos e variações anormais no tráfego de rede. Este artigo apresenta uma técnica para predição de ataques DDoS fundamentada em uma metodologia inovadora para extração de características de rede. A proposta beneficia-se da tolerância ao ruído da transformação ordinal para a predição dos ataques DDoS. A predição aplica o algoritmo One-Class SVM que independe de dados rotulados. A técnica prediz um ataque até 44 minutos antes do seu início com acurácia de 89%.*

Abstract. *Infected devices in Internet of Things (IoT) represent one of the main challenges in fighting distributed denial of service (DDoS) attacks. Attackers camouflage their actions and delay attack prediction, requiring new solutions resilient to noise and variations in network traffic. This article presents a technique for predicting DDoS attacks using a novel methodology for extracting network features. The proposal benefits from the noise tolerance of the ordinal transformation to the DDoS attack prediction. The prediction applies the One-Class SVM algorithm, independent of labeled data. The technique predicts a DDoS attack up to 44 minutes before its start with an accuracy of 89%.*

1. Introdução

A Internet das Coisas (IoT) é uma realidade com múltiplas aplicações na academia, na sociedade e na indústria. Entretanto, suas limitações computacionais e a falta de padronização em relação à segurança dos dispositivos IoT os tornam um alvo fácil para os invasores. Um ataque recorrente envolvendo dispositivos IoT é o ataque de negação de serviço distribuído (DDoS), uma das ameaças cibernéticas mais nocivas e comuns [Jyoti and Behal 2021]. Apenas no segundo semestre de 2022 ocorreram 6.797.959 ataques DDoS originados em mais 230 países. Esses ataques têm atingido grandes volumes de dados em uma velocidade sem precedentes. Em novembro de 2022, um ataque DDoS contra um único alvo nos Estados Unidos atingiu 978,5 Gbps [Netscout 2023].

O perigo potencial dos ataques DDoS é tal que não é suficiente apenas detectá-los. Isso ocorre porque no momento em que um ataque é detectado grande parte do dano já foi ocasionado [Abaid et al. 2016]. Predizer os ataques DDoS antes que eles sejam lançados reduz significativamente os gastos necessários para mitigar um ataque em

andamento e reparar um ataque bem-sucedido. Contudo detectar os sinais da preparação de ataques não é uma tarefa trivial. A falta de informações relacionadas ao ataque causa um desequilíbrio significativo de dados, dificultando a diferenciação da preparação do ataque do tráfego de rede regular. Por outro lado, os dados de séries temporais do tráfego da rede (*i.e.*, conjuntos de observações coletadas sequencialmente) são uma fonte valiosa de informação sobre um sistema a ser avaliado. Esses dados podem ser analisados e utilizados em soluções de cibersegurança [Borges et al. 2022]. Porém a análise desses dados pode ser prejudicada pela presença de ruídos e contaminação caso esse processo envolva a utilização direta dos dados observados [Brockwell and Davis 2009].

A caracterização da rede com base na transformação de dados de séries temporais é investigada na literatura de cibersegurança. A técnica Casual Graph Process serviu para calcular o grau de correlação dos nós infectados da rede por meio de cálculos autorregressivos sobre dados de séries temporais em [Ferreira and Nogueira 2018]. Para isso, os cálculos são realizados em tempo real, exigindo recursos computacionais consideráveis. A solução é baseada em modelo autorregressivo, conhecidos por serem sensíveis a valores atípicos, levando a previsões imprecisas [Box et al. 2015]. Em contrapartida, o uso de padrões ordinais foi proposto em [Chagas et al. 2022] para identificar *botnets* (*i.e.*, redes de dispositivos infectados) em IoT. Essa abordagem não paramétrica torna a análise de séries temporais robusta aos ruídos, além de ser adequada para ambientes IoT [Borges et al. 2022]. Essas soluções focam apenas na detecção de ataques. Os trabalhos que realizam a predição seguem abordagens que requerem ou dados rotulados, ou redes neurais com longo tempo de treinamento, demandando maior poder computacional [Silva et al. 2022, Rahal et al. 2020, Brito et al. 2023].

Esse trabalho apresenta uma técnica de predição de ataques DDoS baseado na análise de séries temporais e padrões ordinais extraídos através da simbolização de Bandt-Pompe [Bandt and Pompe 2002]. A combinação de padrões ordinais e descritores da teoria da informação tem se mostrado altamente eficaz na caracterização da dinâmica dos dados [Chagas et al. 2022]. Assim, a técnica emprega os padrões ordinais em conjunto com sua transformação em grafos de transição para representar dados de séries temporais de rede em um novo domínio. Com essas ferramentas é possível caracterizar séries temporais de acordo com seu comportamento. A representação dos dados correlacionados (*i.e.*, características de rede extraídas da transformação) é então utilizada para treinar o modelo de aprendizado de máquina para predizer ataques DDoS. A técnica utiliza o *One Class Support Vector Machine* (SVM) para automatizar a predição. O modelo SVM é treinado em uma única classe positiva (*i.e.*, dados benignos). Posteriormente, esse modelo é usado para identificar instâncias diferentes a essa classe para predizer ataques DDoS. O *One Class* SVM não requer dados rotulados para realizar a predição [Amer et al. 2013], funciona bem em cenários desbalanceados e já foi testado em IoT [Bezerra et al. 2019].

A técnica proposta foi avaliada seguindo quatro experimentos. Os dois primeiros experimentos usam capturas de tráfego de rede disponibilizadas pelo *Czech Technical University dataset* (CTU-13) referentes à rede local da universidade [Garcia et al. 2014]. O terceiro experimento usa a base de dados *DDoS Evaluation Dataset* (CIC-DDoS2019) [Sharafaldin et al. 2019] e o quarto utiliza a captura IoT-23 [Garcia et al. 2020]. No primeiro experimento, a técnica proposta predisse o ataque 44 minutos e 41 segundos antes do seu lançamento com acurácia de 89,89%. No se-

gundo experimento, a predição do ataque ocorreu com 1 minuto e 40 segundos com acurácia de 88,77%. No terceiro experimento a predição ocorreu 13 minutos e 24 segundos antes do seu início. No quarto experimento, a técnica previu os sinais da preparação do ataque DDoS 35 minutos após a execução do *malware* em uma rede IoT com uma acurácia de 72,31%. Os resultados superam o tempo de predição de ataques de [Rahal et al. 2020, de Neira et al. 2023, Silva et al. 2022].

Este artigo prossegue como segue. A Seção 2 apresenta os trabalhos relacionados. A Seção 3 detalha a técnica para predição de ataques DDoS. A Seção 4 apresenta os resultados deste trabalho. Por fim, na Seção 5 as considerações finais são postuladas.

2. Trabalhos Relacionados

A literatura aponta que dispositivos maliciosos costumam gerar dados na rede antes do início do ataque [Jyoti and Behal 2021], sendo possível usá-los para identificar sinais da preparação do ataque. Neste sentido, alguns trabalhos empregam técnicas que visam à detecção de ataques com base na identificação das mensagens de Comando & Controle — C&C (*i.e.*, mensagens trocadas na fase de preparação de um ataque DDoS). Contudo esse procedimento não é eficaz caso as *botnets* mudem suas arquiteturas ou protocolos.Como os invasores procuram ocultar suas atividades, é vital pesquisar maneiras de identificar os sinais de preparação para o ataque [de Neira et al. 2023]. Quando aplicados à predição de ataques DDoS, a literatura é limitada. As pesquisas buscam formas de prever estágios comportamentais relacionados aos ataques na rede. Por exemplo, em [Silva et al. 2022] os autores descrevem um sistema baseado no aprendizado profundo para identificar sinais da orquestração de ataques DDoS. O sistema extrai as características *Skewness* e *Kurtosis* para analisar o tráfego da rede. A proposta consegue predizer ataques, mas demanda alto consumo de tempo e recursos. Em [Brito et al. 2023], os autores apresentam um sistema que define automaticamente a melhor configuração da rede neural para distinguir o tráfego normal do tráfego do ataque. O sistema utiliza o *Long short-term memory Autoencoder* para identificar os sinais da preparação do ataque através das características *Skewness*, *Kurtosis* e coeficiente de variação e consegue predizer um ataque DDoS em 29 minutos.

Os poucos trabalhos que realizam a predição de ataque DDoS usam abordagens que requerem dados rotulados [de Neira et al. 2023, Rahal et al. 2020] ou propõem soluções baseadas em redes neurais complexas (*e.g.*, *autoencoders*) [Brito et al. 2023, Silva et al. 2022] que apresentam longo tempo de treinamento e exigem maior poder computacional, sendo inviáveis para ambientes IoT. Além disso, as soluções utilizam poucas características oferecendo uma visão menos robusta da variabilidade dos dados. Esse trabalho avança a literatura com uma técnica de predição de ataques DDoS baseada na análise de séries temporais e padrões ordinais extraídos através da simbolização de Bandt-Pompe, uma metodologia robusta a *outliers* [Borges et al. 2022]. A técnica extrai da transformação oito características para identificar mudanças no comportamento da rede e antecipar ataques DDoS. A solução dispensa dados rotulados, o que impede a vinculação do modelo a tipos específicos de ataques DDoS e a torna adequada para ambientes onde a eficiência computacional é um problema.

3. Predição de Ataques DDoS por Correlação de Séries Temporais

Esta seção descreve a técnica de predição de ataques DDoS, composta por quatro etapas: (i) coleta do tráfego da rede; (ii) extração de características; (iii) treinamento do modelo;

(iv) predição de ataque DDoS. Os dados coletados na Etapa 1 são transformados na Etapa 2 e características são extraídas para treinar o modelo de aprendizado profundo na Etapa 3. O modelo treinado é aplicado na Etapa 4 para a predição do ataque DDoS.

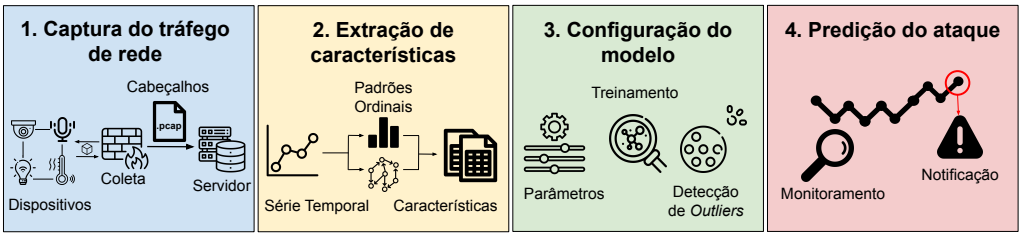

Figura 1. Visão geral da técnica proposta

3.1. Coleta de Dados

A captura de tráfego de rede ocorre por uma ferramenta integrada ao *firewall* que intercepta a entrada e saída da rede e captura amostras dos cabeçalhos dos pacotes transmitidos. A técnica foi projetada para trabalhar de forma centralizada ou distribuída. Na forma centralizada, a ferramenta integrada ao *firewall* encaminha uma cópia do cabeçalho dos pacotes para um dispositivo que irá executar as próximas etapas da técnica proposta. Dependendo do volume de dados trafegados, esse processo requer muitos recursos computacionais. No processamento distribuído, os pacotes são coletados simultaneamente de várias fontes distribuídas geograficamente. Esse processo diminui a ocorrência de gargalos no processamento dos dados, no entanto aumenta a complexidade do sistema. A escolha do modo de captura deve ser feita conforme a estrutura e recursos computacionais disponíveis. As amostras dos cabeçalhos coletados são armazenadas em um servidor de arquivos dedicado. Os dados coletados são exportados para arquivos do tipo Captura de Pacotes (*Packet Capture* — PCAP) para possibilitar a execução da Etapa 2 (Subsec. 3.2).

3.2. Extração de Características

Esta subseção define os atributos do tráfego que serão extraídos dos pacotes e como a técnica os processa para que eles possam ser usados para a predição dos ataques DDoS. Alguns atributos de rede sofrem variações quando o atacante realiza testes ou sincroniza ações antes de iniciar o ataque. 40 atributos representativos para a detecção de ações realizadas antes do início do ataque foram identificados em [Feng et al. 2018]. Exemplos de atributos do tráfego de rede bastante influenciados pela preparação dos ataques são: a quantidade de endereços de IP na origem e destino dos pacotes, a quantidade de pacotes e o tamanho do pacote. Esses atributos foram definidos para serem a base do processo de extração de métricas (detalhado na Subsec. 4.1). Porém usuários da técnica podem selecionar outros atributos. Para realizar a extração de características (Etapa 2) baseado nos atributos do tráfego de rede, a técnica agrega as capturas por unidade de tempo. Inicialmente, a técnica monitora o servidor de arquivos em busca de novas capturas. Quando novos dados estiverem disponíveis, a técnica agrupa os pacotes usando o valor padrão de um segundo, podendo ser alterado pelo usuário. A agregação mantém as propriedades temporais, *i.e.,* os intervalos são ordenados sequencialmente ao longo do tempo. Após a agregação dos pacotes, a técnica processa os pacotes para obter os atributos definidos pelo usuário. Após o processamento, obtém-se uma série temporal para cada atributo.

A extração de características utiliza as séries temporais processadas anteriormente e é baseada na teoria dos padrões ordinais. A primeira ação é transformar a série temporal

capturada em uma sequência ordinal através da metodologia de Bandt-Pompe, que caracteriza a dinâmica das séries temporais e permite a análise de conjuntos de dados complexos. Esta transformação consiste em criar um conjunto de padrões simbólicos baseados na relação ordinal entre pontos de dados sucessivos da série temporal (Subsubseção 3.2.1). A segunda ação consiste em representar os dados transformados em padrões ordinais em um novo domínio através da simbolização em grafo de transição e distribuição de probabilidade (Subsubseção 3.2.2). Esse processo é importante pois a distribuição resultante torna-se menos sensível a ruídos e apresenta bons resultados na predição dos ataques [Chagas et al. 2022]. O ruído pode interferir na distribuição de diferentes formas, em geral causando um desequilíbrio e resultados diferentes do esperado, o que pode levar ao não reconhecimento do ataque pelo modelo. Assim, ao final da Etapa 2, a técnica obtém as características baseadas na metodologia de simbolização ordinal de Bandt-Pompe combinada com descritores da teoria da informação que serão utilizadas na predição dos ataques DDoS. A Subsubseção 3.2.3 detalha as características.

3.2.1. Transformação Ordinal de Bandt-Pompe

A transformação de Bandt-Pompe [Bandt and Pompe 2002] consiste em criar padrões das relações ordinais de pontos consecutivos de uma série temporal. Para obter o conjunto de padrões ordinais da série temporal capturada (etapa 1) são necessários dois processos: (i) a divisão da série temporal em subconjuntos de dimensão D e (ii) a obtenção dos padrões ordinais através da permutação dos índices de cada subconjunto ordenados de forma crescente. Assim, dada uma série temporal $x = (x_1, ..., x_n)$ de comprimento n, dimensão $D \in \mathbb{N}$ e atraso $\tau \in \mathbb{N}$ que define um intervalo entre os pontos considerados para D, a cada instante $t = \{1, ..., n - (D-1)\tau\}$ é definida uma janela deslizante $\mathbf{w}_t$:

$$\mathbf{w}_t = (x_t, x_{t+\tau}, \ldots, x_{t+(D-2)\tau}, x_{t+(D-1)\tau}), \tag{1}$$

em que o padrão ordinal para cada janela representa a permutação necessária nos elementos de $\mathbf{w}_t$ para os elementos serem ordenados. Para cada instante t a relação ordinal consiste na permutação $\pi = \{r_1, r_2, \ldots, r_D\}$ de $(1, 2, \ldots, D)$ tal que:

$$x_{t+r_1-1} \leq x_{t+r_2-1} \leq \cdots \leq x_{t+r_{D-1}-1} \leq x_{t+r_D-1}. \tag{2}$$

Após a transformação, a série temporal é convertida em um conjunto de padrões ordinais $\Pi = \{\pi_1, \ldots, \pi_m\}$ onde $m = n - 1(D-1)\tau$, e cada π_i representa uma permutação do conjunto das $D!$ permutações possíveis. A escolha de D depende do comprimento n da série temporal e deve satisfazer $n \gg D!$ para alcançar estatísticas confiáveis [Borges et al. 2022]. A Figura 2 ilustra o processo de transformação ordinal de uma série temporal. No exemplo, a primeira janela deslizante obtida no tempo $t = 1$ é dada por $\mathbf{w}_1 = (x_1, x_2, x_3)$, a permutação necessária para ordená-la é mover o primeiro elemento para o fim, o segundo para o início e o último para o meio. A nova ordem segue $\mathbf{w'}_t$=($x_{1;3}$;$x_{1;1}$;$x_{1;2}$) correspondendo ao padrão $\pi = \{3, 1, 2\}$.

3.2.2. Representação dos Padrões Ordinais

A análise do conjunto de padrões ordinais extraído de uma sequência de séries temporais é realizada por histogramas de frequências e suas transições. Esse processo envolve

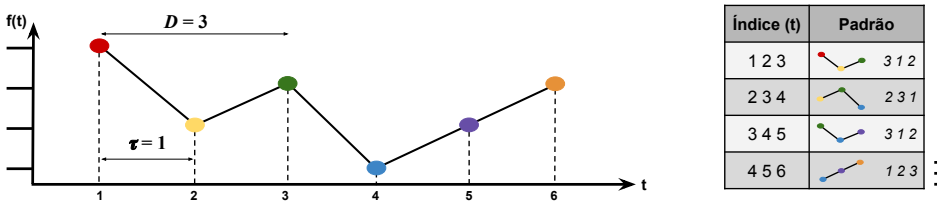

Figura 2. Processo de construção do padrão ordinal de $D = 3$ e atraso τ=1

examinar a frequência de ocorrência de cada padrão individualmente. Isso permite identificar padrões dominantes, padrões menos comuns e anomalias. Para visualizar essa distribuição utilizam-se grafos de transição e distribuição de probabilidade (histogramas) que representam graficamente a probabilidade de transição entre cada padrão.

Distribuição de probabilidade: O cálculo da distribuição de probabilidade de padrões ordinais pode ser realizada através do histograma de frequência. Para isso deve-se extrair a distribuição de probabilidade das permutações $p_\pi = \{p(\pi_t) : \forall t \in 1, \ldots, D!\}$ de um conjunto de padrões ordinais Π obtidos a partir da transformação da série temporal (*i.e.*, tráfego da rede):

$$p(\pi_t) = \frac{|s_{\pi_t}|}{n - (D-1)\tau} \tag{3}$$

em que $\pi_t \in \Pi$ representa cada possível permutação onde $t \in \{1, \ldots, D!\}$, $|s_{\pi_t}| \in \{0, \ldots, m\}$ é o número de padrões observados de tipo π_t e satisfaz as condições $p(\pi_t) \geq 0$ e $\sum_{\pi_t} p(\pi_t) = 1$. A ordem das séries temporais são preservadas, fazendo com que a presença do ruído multiplicativo não afete os padrões produzidos [Chagas et al. 2022].

Grafos de transição: A técnica utiliza a análise do conjunto de padrões ordinais transformados em grafos de transição. Com essa ferramenta é possível caracterizar séries temporais de acordo com seu comportamento para suportar a predição de ataques DDoS. Para caracterizar um conjunto Π de padrões ordinais como um grafo orientado $G = (V, E)$ que represente as transições entre dois padrões ordinais consecutivos ao longo do tempo t, a nova representação assume cada vértice como um padrão e as arestas representam as transições entre eles. A transformação segue [Borges et al. 2022]:

$$V = \{v_{\pi_i} : i = 1, \ldots, D!\}, \text{ e } E = \{(v_{\pi_t}, v_{\pi_{t+1}}) : v_{\pi_t}, v_{\pi_{t+1}} \in V\}. \tag{4}$$

em que cada aresta recebe um peso $W = \{w_{v_{\pi_i}, v_{\pi_j}} : v_{\pi_i}, v_{\pi_j} \in V\}$ que representa a probabilidade de transição entre cada padrão conforme a Equação 5.

$$w_{v_{\pi_i}, v_{\pi_j}} = \frac{|\Pi_{\pi_i, \pi_j}|}{n - (D-1)\tau - 1} \tag{5}$$

em que o termo $|\Pi_{\pi_i, \pi_j}|$ representa o número de transições entre os padrões (*i.e.*, de π_i para π_j) respeitando a restrição $\sum_{v_{\pi_i}, v_{\pi_j}} w_{v_{\pi_i}, v_{\pi_j}} = 1$ e o denominador é o número de transições na série de comprimento $n - (D-1)\tau$.

3.2.3. Extração de Características Obtidas das Transformações

A partir da distribuição de probabilidade e do grafo de transição obtidos da transformação em padrões ordinais são extraídas novas características. Para isso, são aplicados três quantificadores da Teoria da Informação: entropia de permutação, complexidade estatística e

informação de Fisher, os quais já foram validados na literatura para caracterizar o comportamento dinâmico presente em séries temporais [Ribeiro et al. 2017]. Dessa forma, obtêm-se as seguintes características representativas sobre os atributos do tráfego da rede: (i) entropia de permutação normalizada; (ii) complexidade estatística; (iii) medida da informação de Fisher; (iv) número de vértices do grafo de transição; (v) entropia de permutação normalizada da distribuição de pesos das arestas; (vi) complexidade estatística da distribuição de pesos das arestas; (vii) medida da informação de Fisher da distribuição de pesos das arestas e $(viii)$ probabilidade de autotransição entre os vértices.

A **entropia de permutação** é aplicada para medir a complexidade e a imprevisibilidade de uma sequência de padrões ordinais. Quanto maior a entropia de permutação, maior é a diversidade e a imprevisibilidade dos padrões na sequência temporal. Em contrapartida, valores mais baixos indicam que a série temporal original é determinística [Bandt and Pompe 2002]. O cálculo segue:

$$H = -\sum_{t=1}^{D!} p(\pi_t) \log p(\pi_t) \qquad (6) \qquad\qquad H_{norm} = \frac{H}{\log D!}. \qquad (7)$$

onde H é a entropia clásica de Shannon e H_{norm} é a entropia de Shannon normalizada, $0 \leq H \leq \log D!$ e $0 \leq H_{norm} \leq 1$.

A **complexidade estatística** fornece percepções sobre a complexidade e a imprevisibilidade da estrutura (distribuição) dos padrões ordinais na sequência original. Essa medida é útil no reconhecimento de padrões e detecção de anomalias nos dados das séries temporais capturadas. A complexidade estatística dos padrões ordinais pode ser calculada considerando a Entropia de Permutação como uma medida de desequilíbrio que quantifica a diferença em relação a uma distribuição esperada e medida. Assim, com base na divergência de Jensen-Shannon-JS [Lamberti et al. 2004] entre a distribuição de probabilidade associada p_π e a distribuição uniforme p_u, a complexidade estatística segue:

$$C_{JS}[p_\pi, p_u] = Q_{JS}[p_\pi, p_u] H_{norm}, \qquad (8)$$

onde $p_\pi = \{p(\pi)\}$ é a distribuição de probabilidade dos padrões ordinais e p_u é a distribuição uniforme de $\{1, 2, \ldots, D!\}$. A medida de desequilíbrio $Q_{JS}[p_\pi, p_u]$ segue a Equação 9 [Rosso et al. 2007]:

$$Q_{JS}[p_\pi, p_u] = Q_0 \left\{ H\left[\frac{p_\pi + p_u}{2}\right] - \frac{H[p_\pi] + H[p_u]}{2} \right\}, \qquad (9)$$

$$Q_0 = -2 \left\{ \left(\frac{D! + 1}{D!}\right) \ln(D! + 1) - 2\ln(2D!) + \ln(D!) \right\}^{-1}, \qquad (10)$$

onde Q_0 é a constante de normalização conforme $0 \leq Q_{JS} \leq 1$.

A **informação de Fisher** é utilizada para caracterizar a complexidade estatística dos padrões encontrados na série temporal. Assim, fornece percepções sobre a complexidade e desequilíbrio dos padrões. Essa medida estatística apresenta uma propriedade de localidade (*i.e.*, diferente da entropia de Shannon que mede o espalhamento global para indicar a incerteza de um sistema) pois considera diferenças entre as probabilidades consecutivas na distribuição. O quantificador de informações de Fisher aumenta à medida que a densidade da distribuição de probabilidade é mais concentrada (*i.e.*, as probabilidades

estão mais próximas). O cálculo da informação de Fisher para o caso discreto segue:

$$F[p_\pi] = F_0 \sum_{t=1}^{D!-1} (\sqrt{p_{t+1}} - \sqrt{p_t})^2, \tag{11}$$

onde F_0 é uma constante de normalização:

$$F_0 = \begin{cases} 1 \text{ se } p_{i^*} = 1 \text{ para } i^* = 1 \text{ ou } i^* = N \text{ e } p_i = 0, \forall i \neq i^*, \\ 1/2 \text{ caso contrário.} \end{cases} \tag{12}$$

A **probabilidade de autotransição** no grafo (*i.e.*, padrões ordinais consecutivos) é uma característica importante na dinâmica de séries temporais por estar relacionada à sua correlação temporal. A probabilidade de autotransição p_{st} segue a Equação 13.

$$p_{st} = p(\pi_i, \pi_i) = \sum_{i \in \{1,\ldots,D!\}} w(v_{\pi_i}, v_{\pi_i}). \tag{13}$$

dada a matriz de adjacência de G_π, $A_\pi = \{a_{\pi_i,\pi_j} : \pi_i\pi_j \in \Pi\}$, nota-se que p_{st} é a soma da diagonal principal de A_π. Logo, a probabilidade de autotransição é $p_{st} = \sum_{\pi_i \in \Pi} a_{\pi_i,\pi_i}$.

3.3. Treinamento do Modelo para Predição de Ataques DDoS

Na Etapa 3, é utilizado o algoritmo de aprendizado de máquina One-Class SVM para detecção de *outliers*. O algoritmo é utilizado de forma semi-supervisionada, requerendo apenas dados normais para o treinamento [Amer et al. 2013]. O One-Class SVM encontra um limite de decisão que separa os dados normais dos atípicos. O limite de decisão é definido por um hiperplano que maximiza a margem entre os dados. Este limite define novos pontos de dados (normais/atípicos) conforme o lado em que se encontram. A capacidade do One-Class SVM de lidar com dados de alta dimensionalidade depende de alguns parâmetros. O parâmetro *nu* atua como um limite para a taxa de erros no treinamento e como um limite inferior para a proporção de vetores de suporte. Ajustando o valor do parâmetro *nu*, é possível controlar a quantidade de pontos de dados considerados *outliers* durante o treinamento. Esse valor é um número entre [0, 1] que representa a porcentagem de pontos classificados como *outliers*. O parâmetro *Kernel* define como os dados são separados no espaço. Existem várias opções de *Kernel*, *(e.g.*, *linear*, *poly* e *sigmoid*). Os parâmetros se adaptam aos dados para uma solução conforme a necessidade.

3.4. Predição de Ataques

Os dados coletados e processados são usados na Etapa 4 para a predição de ataques DDoS. Para isso, a técnica utiliza o algoritmo de aprendizado de máquina One-Class SVM para identificar *outliers* no conjunto de dados. O One-Class SVM consegue identificar observações incomuns em um conjunto de dados independentemente de terem sido rotulados como *outliers* ou não. Essa característica viabiliza a adoção da técnica em ambientes reais, uma vez que o processo de rotulação é custoso. Os *outliers* representam mudanças no tráfego de rede que podem não ser perceptíveis. Dessa forma, essas alterações são realçadas na Etapa 2. A transformação ordinal para extração de características (*i.e.*, oito características processadas sobre cada um dos três atributos de rede) permite correlacionar os diferentes padrões de dados encontrados nas séries temporais e identificar se dados do tráfego analisados são benignos ou não. A técnica pode ser aplicada ao nível da rede para notificar a equipe responsável sobre a ocorrência de ataques DDoS ou ainda promover a desconexão de dispositivos infectados por *malware*.

4. Avaliação

Essa seção descreve os experimentos e resultados obtidos. Avaliar a predição dos ataques DDoS não é trivial, pois os *datasets* utilizados devem indicar o início dos ataques DDoS e alguma ação de preparação do ataque. O início do ataque indica até qual momento a análise pode ocorrer, uma vez que a predição deve acontecer antes do início. Assim, as avaliações apresentadas neste trabalho usam apenas dados antes do início do ataque DDoS. Essa abordagem faz com que o modelo não seja dependente do tipo de ataque. Além disso, a avaliação utiliza o indicativo do início do ataque para verificar quanto tempo antes a predição ocorreu. O dado de preparação do ataque é importante pois justifica a predição do ataque DDoS. A infecção dos dispositivos ou algum teste de ataque são exemplos de ações de preparação que os atacantes podem realizar [Griffioen et al. 2021].

4.1. Definição dos Experimentos

O **Experimento 1** usa o tráfego de uma rede local disponível na captura 51 da CTU-13. A captura tem 8803 segundos, 41 GB, 46 milhões de pacotes, ataques do tipo *flood Internet Control Message Protocol* (ICMP) e *User Datagram Protocol* (UDP) e dez *bots*. Os pesquisadores lançam os ataques no segundo 5632 e combinam os ataques com dados reais. O *One Class* SVM possui uma etapa de treinamento que não usa dados rotulados. Para treiná-lo, usou-se apenas um terço do *dataset*. Assim, o treinamento vai do início do *dataset* até o segundo 2934. Como o objetivo é predizer os ataques, este experimento analisa somente o tráfego de rede anterior ao início do ataque. Assim, o teste foi até o segundo 5632. O *One Class* SVM aplicado neste experimento utiliza todos os parâmetros padrões da biblioteca Scikit-learn[1], exceto pelo *kernel* alterado para o polinomial (*poly*) e o *nu* que recebeu o valor de 0,1 pois durante os testes maximizou o tempo de predição.

O **Experimento 2** utiliza tráfego de rede local coletado na captura 52 de [Garcia et al. 2014]. A captura tem 972 segundos, 555 MB, 6 milhões de pacotes, um ataque do tipo *flood* do ICMP e três *bots*. Os pesquisadores conduziram o ataque ao segundo 778 da captura e o combinaram com dados reais. Assim como no experimento anterior, este usa um terço do *dataset* para o treinamento do *One Class* SVM (*i.e.*, até o segundo 324) e o período de teste termina no início do ataque (*i.e.*, até o segundo 778). Por fim, o *One Class* SVM aplicado no Experimento 2 foi configurado exatamente como no Experimento 1 ($kernel = poly$ e $nu = 0,1$).

O **Experimento 3** avalia o tráfego de rede do conjunto de dados CIC-DDoS2019, com 19 ataques de diferentes tipos (Portmap, UDP, SYN Flood, entre outros) lançados por pesquisadores em dois dias. O conjunto de dados possui 27 GB de dados referentes a ataques e dados reais, 61 milhões de pacotes. Os *bots* se conectaram à vítima pela Internet. A avaliação concentrou-se no primeiro ataque DDoS do primeiro dia. O ataque começou no 1484º segundo da captura. Para manter o padrão estabelecido nos experimentos anteriores, este experimento também usa um terço do *dataset* como treinamento e dados anteriores ao ataque (*i.e.*, até o segundo 674) e avalia até o início do ataque (*i.e.*, até o segundo 1484). Por fim, o *One Class* SVM aplicado no Experimento 3 foi configurado com $kernel = poly$ e $nu = 0,2$. (*i.e.*, definido empiricamente para melhores resultados).

O **Experimento 4** utiliza o conjunto de dados IoT-23, com 23 cenários de ataques DDoS em ambientes IoT. O cenário 17 contém mais de um bot infectado e ativo.

[1]https://scikit-learn.org/stable/modules/generated/sklearn.svm.OneClassSVM.html

O cenário possui 8,3 GB e 109 milhões de pacotes enviados em 24 horas. Os pesquisadores iniciaram a captura às 06:43:20, e a execução do *malware* foi às 11:43:43 do mesmo dia. Assim, a captura de tráfego pré-infecção possui tráfego legítimo e o tráfego pós-infecção contém vestígios da preparação do ataque. A documentação apresentou um problema elétrico na universidade, portanto este trabalho não identificou a iniciação efetiva de ataques DDoS. Este trabalho levanta a hipótese de que a falta de energia elétrica comprometeu o lançamento do ataque. O *One Class* SVM aplicado no Experimento 4 foi configurado assim como no Experimento 3 ($kernel = poly$ e $nu = 0,2$).

Para realizar os experimentos utilizou-se o intervalo de um segundo para agrupar os pacotes nos experimentos 1, 2 e 3, visando obter predições mais precisas [de Neira et al. 2023, Rafiee et al. 2022]. Como o *dataset* do Experimento 4 possui 24 horas de tráfego de rede, optou-se por usar um minuto para o intervalo. Este trabalho rotulou cada intervalo como normal e como malicioso. A rotulagem de dados é usada apenas para quantificar os resultados, visto que, a detecção de *outliers* realizada pelo *One Class* SVM não usa rótulos para treinar e predizer os ataques DDoS. O intervalo normal compreende todo o tráfego de rede em um segundo ou um minuto (Experimento 4), onde nenhum pacote é originado ou destinado a *bots*. Já o intervalo malicioso compreende todo o tráfego de rede onde ao menos um pacote tem como origem ou destino um *bot*.

Os quatro experimentos utilizaram a quantidade de endereços de IP na origem e no destino dos pacotes. Para a quantidade de IPs de origem, contou-se quantos IPs únicos enviaram pacotes. A quantidade de IPs de destino baseia-se na contagem de IPs únicos existentes no campo de destino do pacote IP. Esses atributos foram selecionados, pois a falsificação de IPs é uma prática comum em ataques DDoS [Jyoti and Behal 2021]. A quantidade de IPs que enviam pacotes antes do ataque apresenta potencial como atributo, pois a preparação do ataque causa variações nesse atributo (*i.e.*, pico na distribuição dos dados) [Griffioen et al. 2021]. Nos experimentos 1, 2 e 3 escolheu-se a quantidade de pacotes, por ser um dos atributos mais relevantes identificados em [Feng et al. 2018]. Quando os atacantes testam os ataques, a quantidade de pacotes trafegados varia [Feng et al. 2018]. No Experimento 4, o atributo maior pacote foi selecionado, pois o dataset IoT-23 possui aspectos distintos. Esse atributo evidência quando pacotes com tamanho muito grande são trafegados.

Aplicando a metodologia de transformação dos padrões ordinais (Sec. 3) sob os atributos do tráfego de rede, são extraídas as características usadas para a avaliação da predição: entropia de permutação normalizada, complexidade e complexidade estatística dos pesos das arestas, informação de Fisher dos pesos das arestas e probabilidade de autotransição (Subsec. 3.2). Assim, cada atributo é transformado em oito características. Os resultados da técnica estão disponíveis em[2]. A combinação dos padrões ordinais com os atributos foi responsável por maximizar os resultados da predição de ataques DDoS dentre vários testes. Para eliminar tendências errôneas e avaliar a solução ao longo do tempo, a técnica utiliza o conceito de janela deslizante [Bury et al. 2020]. Definiu-se 5% para o tamanho da janela com o intuito de maximizar o tempo de predição dos ataques.

A avaliação dos resultados utiliza a acurácia, precisão e o *recall*. A acurácia representa a proporção de amostras classificadas corretamente em relação ao total de amostras.

[2]`https://github.com/ligiafb/sbseg-data-2023`

A precisão e o *recall* são usados para complementar a análise dos resultados. A precisão indica a relação entre as observações rotuladas pela técnica para um tipo específico e quantas são do tipo assumido. O *recall* apresenta a relação entre todas as observações esperadas do tipo específico e quantas observações a técnica classificou corretamente. Devido ao desbalanceamento inerente ao ataque DDoS e à variação na quantidade de amostras nas classes, é necessário considerar a precisão e o *recall* de forma ponderada. Nesse trabalho, utiliza-se a média ponderada da precisão e do *recall* considerando a quantidade de amostras em cada classe. Isso permite uma avaliação mais equilibrada e representativa.

4.2. Resultados

A Tabela 1 apresenta o resultado considerando a captura 51 da CTU-13. A tabela indica que três intervalos maliciosos, onde os *bots* trafegam dados, foram corretamente identificados. Assim, a técnica proposta emitiria corretamente três alertas indicando a possibilidade de que, no futuro, um ataque DDoS pudesse acontecer. O primeiro alerta correto aconteceu 44 minutos e 41 segundos antes do início do ataque. O resultado indica que 2422 intervalos normais foram corretamente identificados. 60 intervalos normais foram identificados como *outliers* gerando falsos positivos. 213 intervalos maliciosos foram erroneamente identificados como intervalos normais (falsos negativos). Os resultados indicam uma acurácia de 89,88%, precisão média de 84,94% e *recall* médio de 89,88%.

Tabela 1. Resultados no Experimento 1

Matriz de confusão		Classe Real			
		Intervalo malicioso	Intervalo normal	Acurácia	89,88%
Classe Hipotética	Intervalo malicioso	3	60	Precisão	84,94%
	Intervalo normal	213	2422	*Recall*	89,88%

A Tabela 2 exibe os resultados da execução da proposta para a captura 52 do CTU-13. A tabela indica que dois intervalos maliciosos (*i.e.*, onde os *bots* trafegam dados), foram corretamente identificados. Assim, a técnica proposta emitiria dois alertas indicando que, no futuro, um ataque DDoS poderia acontecer. A predição do ataque (alerta) aconteceu 1 minuto e 50 segundos antes do lançamento do ataque. A Tabela 2 mostra que 29 intervalos normais foram erroneamente identificados como *outliers*. Isso gerou 29 falsos positivos. Em relação aos falsos negativos, 22 intervalos maliciosos foram erroneamente identificados como intervalos normais. O resultado indica que 401 intervalos normais foram corretamente identificados. Os resultados indicam acurácia de 88,77%, precisão média de 90,13% e *recall* médio de 88,77%.

Tabela 2. Resultados no Experimento 2

Matriz de confusão		Classe Real			
		Intervalo malicioso	Intervalo normal	Acurácia	88,77%
Classe Hipotética	Intervalo malicioso	2	29	Precisão	90,13%
	Intervalo normal	22	401	*Recall*	88,77%

A Tabela 3 reporta os resultados do Experimento 3 (CIC-DDoS2019). A tabela indica que 16 intervalos maliciosos, nos quais os *bots* trafegam dados foram corretamente identificados. Assim, a técnica emitiria corretamente 16 alertas, indicando um ataque futuro. A análise indica que a predição ocorreu 13 minutos e 24 segundos antes do lançamento do ataque. A tabela mostra a ocorrência de 17 falsos positivos, 234 falsos

negativos e 543 intervalos normais que foram corretamente identificados. Os resultados indicam uma acurácia de 69,01%, precisão média de 63,28% e *recall* médio de 69,01%.

Tabela 3. Resultados no Experimento 3

Matriz de confusão		Classe Real			
		Intervalo malicioso	Intervalo normal	Acurácia	69,01%
Classe Hipotética	Intervalo malicioso	16	17	Precisão	63,28%
	Intervalo normal	234	543	*Recall*	69,01%

O Experimento 4 analisou até o minuto 351 da captura IoT-23. O resultado da Tabela 4 é referente aos 80 minutos antes e 50 minutos depois da execução do *malware*. A análise considerou 50 minutos para reforçar que a abordagem proposta pode identificar rapidamente os sinais de preparação para o ataque, mesmo considerando um cenário com mais dados de tráfego normais (80 minutos antes da execução do *malware*). A análise indicou corretamente a existência de 15 intervalos maliciosos, 1 falso positivo, 35 falsos negativos e 79 verdadeiros negativos. Isso resulta em uma acurácia de 72,31%, uma precisão ponderada de 78,70% e um *recall* ponderado de 72%. A técnica identificou a preparação do ataque DDoS 35 minutos após a execução do *malware* em uma rede IoT.

Tabela 4. Resultados no Experimento 4

Matriz de confusão		Classe Real			
		Intervalo malicioso	Intervalo normal	Acurácia	72,31%
Classe Hipotética	Intervalo malicioso	15	1	Precisão	78,70%
	Intervalo normal	35	79	*Recall*	72,31%

4.3. Discussão

Os resultados apresentados indicam a relevância da técnica apresentada neste trabalho. No Experimento 1, a predição do ataque ocorreu 44 minutos e 41 segundos antes do início do ataque (*i.e.*, 19 segundos após o início da infecção dos *bots*). Esses resultados superam o tempo de predição de [Brito et al. 2023, de Neira et al. 2023, Rahal et al. 2020] (Tab. 5). A precisão excede a obtida em [de Neira et al. 2023] em 5,29%, mas é menor que a obtida em [Brito et al. 2023, Rahal et al. 2020]. Para [Brito et al. 2023] alcançar esses resultados, os autores usaram uma rede neural *Autoencoder* que demanda mais tempo e poder de processamento. Por exemplo, o sistema proposto em [Brito et al. 2023] utilizou 42 e 28 minutos para configurar e treinar as redes neurais LSTM *Autoencoder* nas capturas 51 e 52 da CTU-13. Já o SVM One-Class utilizou apenas 0.056566 e 0.002813 segundos para esse processo no mesmo dataset.

O *dataset* utilizado no Experimento 2 é muito menor (972 segundos), tornando a predição do ataque DDoS mais desafiadora. Assim, a predição ocorreu 1 minuto e 50 segundos antes do lançamento do ataque (343 segundo após o início da infecção). Mesmo com uma acurácia menor, os resultados dos Experimento 3 e 4 são muito relevantes. O *dataset* utilizado no Experimento 3 apresenta uma topologia diferente dos experimentos 1 e 2. Neste caso, a Internet conecta a vítima com os atacantes dificultando a predição dos ataques DDoS e mesmo assim, a proposta predisse o primeiro ataque com 13 minutos e 24 segundos superando [de Neira et al. 2023]. A acurácia é menor no experimento 3 quando comparado com os Experimentos 1 e 2. Como o *dataset* não indica o início da infecção, é possível que o processo de rotulação tenha indicado intervalos maliciosos erroneamente.

Tabela 5. Comparação dos resultados

Experimento	Acurária	Precisão	Recall	Predição	Dataset
Experimento 1	89,88%	84,94%	89,88%	44m e 41s	CTU-13 (Captura 51)
Experimento 2	88,77%	90,13%	88,17%	1m e 50s	CTU-13 (Captura 52)
Experimento 3	69,01%	63,28%	69,01%	13m e 24s	CIC-DDoS2019
Experimento 4	72,31%	78,70%	72,31%	35m	IoT-23
[Brito et al. 2023]	97,89%	97,4%	97,9%	29m e 51s	CTU-13 (Captura 51)
[Rahal et al. 2020]	N/A	N/A	N/A	5m e 41s	CTU-13 (Captura 51)
[de Neira et al. 2023]	98,87%	N/A	N/A	3m e 49s	CTU-13 (Captura 52)
[de Neira et al. 2023]	99,60%	N/A	N/A	3m e 55s	CIC-DDoS2019

Os resultados do Experimento 4 também são relevantes, mesmo com a acurácia em 72,31%. Isso porque o *dataset* utilizado neste cenário possui tráfego IoT que introduz aspectos não consideradas nos experimentos anteriores e que são extremamente relevantes no atual cenário. Embora não tenha sido possível medir quanto tempo antes do início do ataque ocorreu a previsão, o Experimento 4 foi um excelente exemplo de quão importante é a previsão do ataque. A previsão no conjunto de dados IoT ocorreria apenas 35 minutos após o início do *malware*. Portanto a proposta maximizou o tempo para lidar com ataques DDoS. A quantidade de falsos negativos foi o principal fator responsável por diminuir os resultados apresentados. Esse trabalho hipotetiza que nem sempre o tráfego gerada pelos *bots* é suficiente para impactar no tráfego de rede. Apesar desse tipo de erro impactar negativamente a acurácia, precisão e *recall*, o impacto desse tipo de erro é menor do que o falso positivo. Esse erro alerta erroneamente que há a possibilidade de um ataque DDoS. Este não é um grande problema, considerando que o propósito da técnica é predizer os ataques. A quantidade de falsos positivos em relação ao total de amostras é de 2%, 6%, 2% e 1% nos Experimento 1, 2, 3 e 4 respectivamente. Esses resultados são baixos visto que a solução não usa rótulos para realizar a predição e os dados são desbalanceados.

5. Conclusão

Este trabalho apresenta uma técnica de predição de ataques DDoS. A proposta combina a transformação de séries temporais em padrões ordinais e sua simbolização em grafos de transição para a extração de características representativas do tráfego da rede. As novas características são usadas para treinar o modelo de aprendizado de máquina *One-Class SVM* que não requer dados rotulados. No melhor resultado, a técnica predisse um ataque DDoS com 44 minutos e 41 segundos de antecedência, superando resultados da literatura.

Agradecimentos

Este trabalho foi financiado pela FAPESP, bolsas #2018/23098-0 e #2022/06840-0 CNPq, bolsas #309129/2017-6 e #432204/2018-0, CAPES, bolsas #88887.501287/2020-00.

Referências

Abaid, Z., Sarkar, D., Kaafar, M. A., and Jha, S. (2016). The early bird gets the botnet:a markov chain based early warning system for botnet attack. In *IEEE LCN*, pages 1–8.

Amer, M., Goldstein, M., and Abdennadher, S. (2013). Enhancing one-class support vector machines for unsupervised anomaly detection. *ACM SIGKDD*, pages 8–15.

Bandt, C. and Pompe, B. (2002). Permutation entropy: a natural complexity measure for time series. *Physical review letters*, 88(17):174102.

Bezerra, V. H., da Costa, V. G. T., Barbon Junior, S., Miani, R. S., and Zarpelão, B. B. (2019). Iotds: A one-class classification approach to detect botnets in internet of things devices. *Sensors*, 19(14):3188.

Borges, J. B., Medeiros, J. P., Barbosa, L. P., Ramos, H. S., and Loureiro, A. A. (2022). Iot botnet detection based on anomalies of multiscale time series dynamics. *IEEE TKDE*.

Box, G. E., Jenkins, G. M., Reinsel, G. C., and Ljung, G. M. (2015). *Time series analysis: forecasting and control*. John Wiley & Sons.

Brito, D., Neira, A., Borges, L., Araújo, A., and Nogueira, M. (2023). Um sistema autônomo para a predição de ataques de ddos em redes locais e internet. In *WGRS*, pages 29–42, Porto Alegre, RS, Brasil. SBC.

Brockwell, P. J. and Davis, R. A. (2009). *Time series: theory and methods*. Springer science & business media.

Bury, T. M., Bauch, C. T., and Anand, M. (2020). Detecting and distinguishing tipping points using spectral early warning signals. *J. R. Soc.*, 17(170).

Chagas, E. T., Borges, J. B., and Ramos, H. S. (2022). Uso de padrões ordinais na caracterização e análise de ataques de botnets em internet das coisas (IoT). In *WebMedia*, pages 133–137. SBC.

de Neira, A. B., de Araujo, A. M., and Nogueira, M. (2023). An intelligent system for DDoS attack prediction based on early warning signals. *IEEE TNSM*, 20(2):1–13.

Feng, Y., Akiyama, H., Lu, L., and Sakurai, K. (2018). Feature selection for machine learning-based early detection of distributed cyber attacks. In *DASC*, pages 173–180, Greece. IEEE.

Ferreira, A. E. and Nogueira, M. (2018). Identificando botnets geradoras de ataques ddos volumétricos por processamento de sinais em grafos. In *WGRS*. SBC.

Garcia, S., Grill, M., Stiborek, J., and Zunino, A. (2014). An empirical comparison of botnet detection methods. *C&S*, 45:100–123.

Garcia, S., Parmisano, A., and Erquiaga, M. J. (2020). IoT-23: A labeled dataset with malicious and benign IoT network traffic.

Griffioen, H., Oosthoek, K., van der Knaap, P., and Doerr, C. (2021). Scan, test, execute: Adversarial tactics in amplification ddos attacks. In *ACM SIGSAC*, pages 940–954.

Jyoti, N. and Behal, S. (2021). A meta-evaluation of machine learning techniques for detection of DDoS attacks. In *INDIACom*, pages 522–526, India. IEEE.

Lamberti, P. W., Martin, M., Plastino, A., and Rosso, O. (2004). Intensive entropic non-triviality measure. *PHYSA*, 334(1-2):119–131.

Netscout (2023). Findings from 2nd half 2022. [(Acessado em: Abril de 2023)]. www.netscout.com/threatreport/global-highlights.

Rafiee, M. et al. (2022). Self-organization map (SOM) algorithm for ddos attack detection in distributed software defined network (D-SDN). *JIST*, 2(38):120.

Rahal, B. M., Santos, A., and Nogueira, M. (2020). A distributed architecture for DDoS prediction and bot detection. *IEEE Access*, 8:159756–159772.

Ribeiro, H. V., Jauregui, M., Zunino, L., and Lenzi, E. K. (2017). Characterizing time series via complexity-entropy curves. *Physical Review E*, 95(6):062106.

Rosso, O. A., Larrondo, H., Martin, M. T., Plastino, A., and Fuentes, M. A. (2007). Distinguishing noise from chaos. *Physical review letters*, 99(15):154102.

Sharafaldin, I., Lashkari, A. H., Hakak, S., and Ghorbani, A. A. (2019). Developing realistic distributed denial of service (DDoS) attack dataset and taxonomy. In *ICCST*.

Silva, G. L. F. E., de Neira, A. B., and Nogueira, M. (2022). A deep learning-based system for ddos attack anticipation. In *LATINCOM*, pages 1–6. IEEE.