Capítulo 2

Uma Visão sobre Sistemas de Detecção de Intrusão Baseados em Anomalias

Kelson Carvalho Santos, Rodrigo Sanches Miani, Flávio de Oliveira Silva

Abstract

This short course covers Intrusion Detection Systems (IDS) in information security, particularly Anomaly-based IDS, and their enhancement through Machine Learning. In addition, it highlights the types of systems and the utilization of Machine Learning in creating more advanced IDS. In the same way, it discusses the relevance of datasets for improving the adaptability of IDS models to detect unknown attacks. Thus, by elucidating anomaly-based intrusion detection strategies, the short course assists in comprehending these concepts while unveiling the challenges faced in the practical implementation of IDS in an environment of constantly evolving cyber threats.

Resumo

Este minicurso aborda os Sistemas de Detecção de Intrusão (Intrusion Detection Systems - IDS) na segurança da informação, sobretudo os IDS baseados em Anomalias e o seu impulsionamento com o Aprendizado de Máquina. Além disso, são destacados os tipos de sistemas e a aplicação do Aprendizado de Máquina na criação de IDS mais avançados. Da mesma forma, é discutido a relevância dos conjuntos de dados para aprimorar a capacidade de adaptação dos modelos de IDS para detecção de ataques desconhecidos. Assim, ao elucidar estratégias de detecção de intrusão baseados em anomalias, o minicurso auxilia na compreensão desses conceitos, enquanto revela os desafios enfrentados na aplicação prática dos IDS em um ambiente de constante evolução de ameaças cibernéticas.

2.1. Introdução

A crescente dependência da tecnologia digital tem ampliado as fronteiras cibernéticas, tornando as redes e os sistemas potenciais alvos para ataques maliciosos [9]. Assim, a proliferação de ameaças cibernéticas exige abordagens avançadas de proteção. Nesse

cenário, os Sistemas de Detecção de Intrusão *(Intrusion Detection Systems - IDS)* desempenham um papel importante na identificação e prevenção de atividades maliciosas.

Os IDS são mecanismos de segurança cibernética projetados para monitorar, analisar e alertar sobre atividades suspeitas ou maliciosas em redes de computadores ou sistemas de informação. Essas atividades podem incluir tentativas de acessar dados não autorizados, explorações de vulnerabilidades, ataques de *malwares* e outras ameaças à segurança.

No geral, existem dois tipos principais de Sistemas de Detecção de Intrusão [23]: IDS baseados em Rede *(Network-based IDS - NIDS)* e IDS baseados em Host *(Host-based IDS - HIDS)*, conforme ilustrado na Figura 2.1.

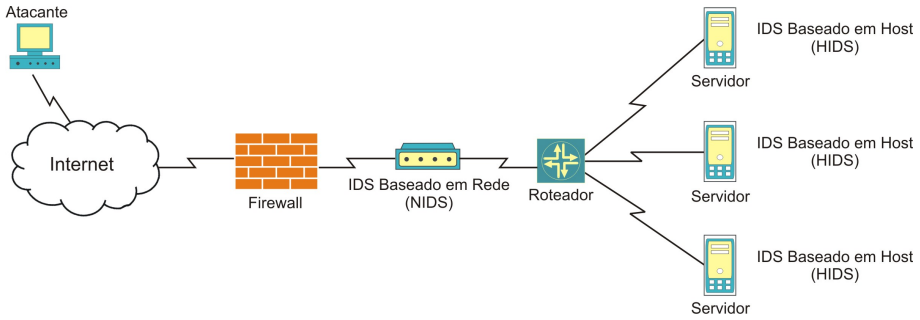

Figura 2.1. Posicionamento dos NIDS e HIDS na estrutura da rede.

Os NIDS são sistemas que monitoram o tráfego de rede em busca de padrões de atividades que possam indicar uma intrusão. Esses sistemas são implantados em pontos estratégicos da rede e examinam os pacotes de dados à medida que trafegam pela rede. Eles não exigem a instalação de *softwares* em *hosts* individuais, tornando-os mais escaláveis.

Por sua vez, os HIDS são sistemas que operam em nível de *host*, monitorando as atividades que ocorrem em um sistema específico, como um servidor ou uma estação de trabalho. Esses sistemas analisam registros de eventos, *logs* do sistema operacional e outras fontes de dados para identificar atividades potencialmente maliciosas. Os HIDS podem detectar ataques direcionados a um único *host*, como tentativas de escalonamento de privilégios, modificações indevidas de arquivos e atividades suspeitas de processos.

Além dessas categorias básicas, os IDS também são classificados com base em suas abordagens de detecção. As duas principais são: Detecção baseada em Assinaturas *(Signature-based Detection)* [11] e Detecção baseada em Anomalias *(Anomaly-based Detection)* [6] .

A Detecção baseada em Assinaturas é uma abordagem que envolve comparar as atividades em tempo real com um banco de dados de padrões conhecidos de ataques e comportamentos maliciosos. Quando uma correspondência *(match)* é encontrada, o IDS emite um alerta. No entanto, essa abordagem é limitada a detectar ameaças previamente identificadas e não é eficaz contra ataques desconhecidos [20].

Por outro lado, a Detecção baseada em Anomalias é uma abordagem que envolve a criação de um perfil de comportamento normal do sistema ou da rede. Assim, o IDS

monitora as atividades em busca de desvios significativos desse comportamento padrão. Isso possibilita a detecção de ataques não identificados anteriormente, mas pode gerar um número maior de falsos positivos, já que em alguns casos, as variações legítimas também podem ser consideradas anômalas.

A Detecção baseada em Anomalias pode ser realizada usando métodos estatísticos, algoritmos de aprendizado de máquina ou uma combinação de ambos [11]. Essa abordagem de detecção é mais difundida com o aprendizado de máquina, que usa algoritmos para identificar padrões complexos que podem indicar atividades maliciosas.

A utilização do aprendizado de máquina para detecção de intrusão pode aumentar a capacidade de identificar ataques anteriormente desconhecidos ou variações de padrões de ataque conhecidos [13].

No aprendizado de máquina existem vários algoritmos de classificação que podem ser aplicados no desenvolvimento de modelos para detecção de intrusão, como *Decision Tree, Random Forest, SVM (Support Vector Machine), kNN (k-Nearest Neighbor), ANN (Artificial Neural Network), XGBoost (eXtreme Gradient-Boosting), LightGBM (Light Gradient-Boosting Machine),* entre outros.

Um modelo de aprendizado de máquina para detecção de intrusão deve ser treinado, validado e testado antes da implementação. Portanto, é nesse contexto que o aprendizado de máquina está sendo bastante utilizado na detecção de intrusão, que baseado em dados, aprende a diferenciar o tráfego de rede entre normal e malicioso [16].

Para o bom desempenho no treinamento dos modelos de aprendizado de máquina para detecção de intrusão, são necessários conjuntos de dados que contenham informações relevantes. Assim é possível estudar os padrões de ataques e as atividades anormais de um sistema de rede.

Diante do exposto, este minicurso discute os Sistemas de Detecção de Intrusão baseados em Anomalias com uso de Aprendizado de Máquina, ou simplesmente, IDS baseados em Aprendizado de Máquina *(ML-based IDS)*. Esses sistemas analisam os dados do tráfego de rede para diferenciar o que é considerado normal e malicioso, de acordo com um modelo de classificação que é treinado, validado e testado.

Considerando esses pontos, o minicurso tem o objetivo de apresentar uma introdução aos IDS baseados em Aprendizado de Máquina, por meio de abordagens teóricas e enriquecidas com demonstrações práticas. Assim, o minicurso auxilia a compreender, projetar e aplicar Sistemas de Detecção de Intrusão baseados em Anomalias.

Além deste material que faz parte do minicurso, também serão disponibilizados os conjuntos de dados e códigos que serão trabalhados nas demonstrações práticas.

A continuidade deste material está organizado da seguinte forma: A Seção 2 apresenta um resumo sobre os métodos do aprendizado de máquina na criação de modelos de classificação para detecção de intrusão; A Seção 3 discute a relevância dos conjuntos de dados na criação de modelos para detecção de intrusão; A Seção 4 descreve uma visão geral sobre as etapas básicas do desenvolvimento de um IDS baseado em Aprendizado de Máquina; A Seção 5 destaca os problemas e desafios encontrados nos IDS baseados em Anomalias; A Seção 6 descreve a metodologia utilizada na aplicação do minicurso,

detalhando as etapas: teórica e prática; A Seção 7 apresenta uma breve discussão sobre a relevância do minicurso na divulgação da pesquisa.

2.2. Aprendizado de Máquina aplicado em IDS

O aprendizado de máquina é uma abordagem essencial para aprimorar os Sistemas de Detecção de Intrusão *(Intrusion Detection Systems - IDS)*, afim de fortalecer a capacidade de identificar ameaças cibernéticas desconhecidas [8].

Ao analisar padrões de comportamento e anomalias em dados de rede, os algoritmos de aprendizado de máquina podem detectar atividades maliciosas que escapam das regras tradicionais de detecção. Isso resulta em uma detecção mais precisa de ataques desconhecidos e uma redução significativa de falsos positivos.

Vários algoritmos de aprendizado de máquina, como *Decision Tree, Random Forest, SVM (Support Vector Machine), kNN (k-Nearest Neighbor), Logical Regression, XGBoost (eXtreme Gradient-Boosting), ANN (Artificial Neural Network)*, entre outros, podem ser empregados na construção de modelos de classificação para detecção de intrusão (consulte a Tabela 2.1).

Os modelos construídos permitem que os IDS se adaptem dinamicamente a novos vetores de ataque, sendo evoluídos para acompanhar as táticas em constante mudança dos invasores [33]. Esses modelos ao serem treinados com conjuntos de dados representativos e diversificados, permitem que os IDS aprendam a reconhecer padrões e comportamentos anômalos, aprimorando a sua eficácia na proteção de redes e sistemas [25].

Os modelos de aprendizado de máquina são identificados de acordo com os métodos de aprendizado, a seguir: Aprendizado Supervisionado, Aprendizado Não Supervisionado e Aprendizado Semi-supervisionado (ver Figura 2.2).

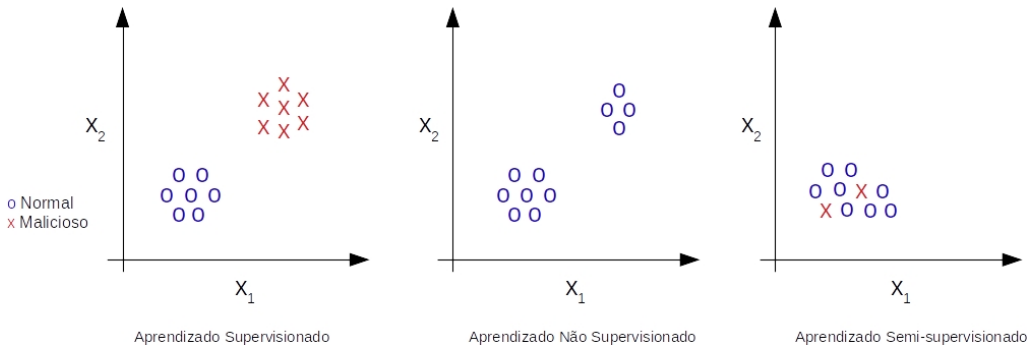

Figura 2.2. Tipos de aprendizado de máquina.

O Aprendizado Supervisionado é utilizado com mais frequência nos modelos de IDS, onde requer um conjunto de dados totalmente rotulado para encontrar a relação entre os dados e suas classes, durante as etapas de treinamento, validação e teste [27]. A rotulagem manual dos dados é cara e demorada, consequentemente, a falta de dados rotulados constitui um problema nesse tipo de aprendizado [14].

Por outro lado, o Aprendizado Não Supervisionado pode ser usado para identificar anomalias ou valores discrepantes no conjunto de dados. Esse tipo de aprendizado é

Tabela 2.1. Algoritmos de aprendizado de máquina.

Algoritmos	Descrições
Decision Tree	Este algoritmo cria uma árvore de decisão que pode ser interpretável. Nela é possível excluir automaticamente recursos irrelevantes e redundantes. O processo de aprendizado inclui a seleção de atributos, a geração de árvores e a poda de árvores. O algoritmo seleciona os atributos mais adequados e gera nós filhos a partir do nó raiz [14].
Random Forest	Este algoritmo consiste em múltiplas árvores de decisão, que funciona criando *n* árvores diferentes com a seleção aleatória de várias amostras do conjunto de dados. Em seguida é realizado a classificação (predição) para cada árvore de decisão, havendo uma votação e selecionado o resultado que obtiver a maioria dos votos [24].
SVM (Support Vector Machine)	A estratégia deste algoritmo é encontrar um hiperplano de separação de margem máxima no espaço de atributos de *n* dimensões. O algoritmo pode alcançar resultados gratificantes mesmo com conjuntos de treinamento em pequena escala, porque o hiperplano de separação é determinado apenas por um pequeno número de vetores de suporte. No entanto, é sensível ao ruído próximo ao hiperplano. As funções de *kernel*, geralmente, são utilizadas para dados não lineares [14].
kNN (k-Nearest Neighbors)	Este algorimto é baseado na hipótese múltipla, onde a maioria dos vizinhos de uma amostra pertencer a mesma classe, a amostra tem uma alta probabilidade de pertencer aquela classe. Assim, o resultado da classificação está relacionado apenas aos *k*-vizinhos mais próximos. O parâmetro *k* influência muito no desempenho do modelo. Quanto menor for *k*, mais complexo é o modelo e maior o risco de sobreajuste do modelo. Por outro lado, quanto maior for *k*, mais simples é o modelo e mais fraca é a capacidade de ajuste [14].
Logical Regression	Este é um tipo de algoritmo linear logarítmico, que calcula as probabilidades de diferentes classes por meio de distribuição logística paramétrica. O algoritmo não consegue lidar bem com dados não lineares, o que limita a sua aplicação [14].
XGBoost (eXtreme Gradient-Boosting)	Este algoritmo foi projetado, principalmente, para ganho de velocidade e desempenho usando árvores de decisão. Tem a vantagem do processamento paralelo, que utiliza todos os núcleos da máquina. É altamente escalável e eficaz para lidar com questões de classificação e pré-processamento de dados em alto nível [4].
ANN (Artificial Neural Network)	São algoritmos bastante utilizados para tarefas de classificação em múltiplos domínios [3]. Em relação a detecção de intrusão, este algoritmos são capazes de capturar relações altamente complexas e não lineares entre variáveis dependentes e independentes, sem o conhecimento prévio de ambas [31].

útil em cenários onde a maioria dos dados pertence a uma classe e a principal tarefa é detectar instâncias que se desviam do padrão normal. [29]. Para isso, é comum a aplicação de técnicas de agrupamentos, onde os algoritmos só reconhecem os dados considerados normais e todo o resto é classificado como malicioso [11].

Já o Aprendizado Semi-supervisionado pode ser empregado quando os rótulos estão disponíveis somente nas instâncias de dados normais [7]. Sendo as anomalias detectadas pelas instâncias de dados dos atributos que se desviam, significativamente, do modelo construído. Esse tipo de aprendizado também pode utilizar uma pequena quantidade de dados rotulados, juntamente, com um conjunto maior de dados não rotulados. Um exemplo, pode ser o algoritmo *One-class Support Vector Machine (OSVM)*, que pode ser aplicado em cenários de detecção de intrusão onde os dados de anomalias rotulados são escassos [2].

Embora os métodos de aprendizado de máquina existam há bastante tempo, ainda não foi estabelecido quais métodos são mais eficientes para a detecção de intrusão [35]. Portanto, é necessária a realização de uma avaliação de desempenho dos dados de referência para comparar os diferentes métodos.

Com isso, a Tabela Verdade, também conhecida como Matriz de Confusão e ilustrada na Figura 2.3, serve de base para avaliação de desempenho dos modelos de aprendizado de máquina. A partir desta tabela, um conjunto de métricas (consulte a Tabela 2.2) podem ser empregadas para avaliar de forma abrangente o desempenho dos modelos.

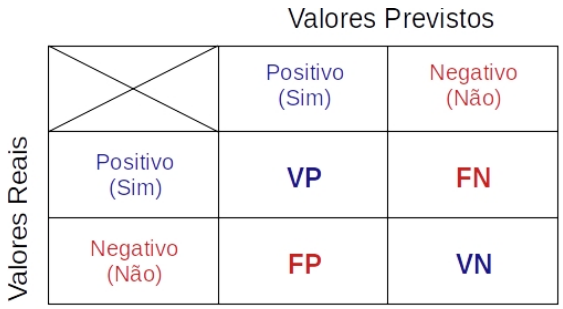

Figura 2.3. Tabela verdade.

A Tabela Verdade (Matriz de Confusão) é composto por:

- VP (Verdadeiro Positivo): significa a proporção de eventos maliciosos classificados corretamente;
- VN (Verdadeiro Negativo): significa a proporção de eventos normais classificados corretamente;
- FP (Falso Positivo): significa a proporção de eventos normais classificados erroneamente como maliciosos;
- FN (Falso Negativo): significa a proporção de eventos maliciosos classificados erroneamente como normais.

Tabela 2.2. Métricas de avaliação de desempenho dos modelos.

Métricas	Descrições	Fórmulas
Precision	É a porcentagem de eventos verdadeiros positivos sobre o número total de positivos identificados.	$\frac{TP}{TP+FP}$
Recall	É a porcentagem de eventos verdadeiros positivos identificados como maliciosos sobre o total de eventos.	$\frac{TP}{TP+FN}$
Accuracy	É a porcentagem de previsões corretas (verdadeiras e falsas).	$\frac{TP}{TP+FN}$
False Alarm Rate	É a porcentagem de eventos maliciosos classificadas incorretamente sobre o número total de eventos normais.	$\frac{FP}{TN+FP}$
Miss Rate	É a porcentagem de eventos maliciosos classificados incorretamente sobre o número total de eventos maliciosos.	$\frac{FN}{FN+TP}$
Error Rate	É a porcentagem de eventos classificadas incorretamente sobre o total de eventos.	$\frac{FP+FN}{TP+TN+FP+FN}$
False Positive Ratio	É a porcentagem de falsos positivos sobre o número total de positivos identificados.	$\frac{FP}{TP+FP}$
False Negative Ratio	É a porcentagem de falsos negativos sobre o número total de negativos identificados.	$\frac{FN}{TN+FN}$
F-Measure	Utiliza a porcentagem do *Precision* e do *Recall* para medir a exatidão do método.	$2 * \frac{Precision*Recall}{Precision+Recall}$
Total Cost Ratio (TCR)	É a porcentagem do custo de eventos classificados incorretamente, onde λ é o custo relativo de ambos os erros.	$\frac{FN+TP}{\lambda(FP+FN)}$
Weighted Error (W Err)	É a porcentagem do erro ponderado que é calculado usando um peso específico λ.	$\frac{\lambda TN+TP}{\lambda FP}$
ROC Curve	É a porcentagem da taxa de verdadeiros positivos, plotada em relação à taxa de falsos positivos.	não existe

Fonte: Adaptado de [17].

Em resumo, o aprendizado de máquina fomenta os Sistemas de Detecção de Intrusões (IDS) a um mecanismo de defesa proativo e inteligente, aumentando a sua capacidade de oferecer uma defesa mais robusta contra ameaças cibernéticas emergentes. Com o aprendizado de máquina os IDS proporcionam uma abordagem flexível e adaptável para detecção de intrusão. Dessa forma, os IDS podem discernir padrões e anomalias no tráfego de rede com maior precisão, minimizando falsos positivos e melhorando a acurácia da detecção de ameaças.

2.3. Conjuntos de Dados para Detecção de Intrusão

Os conjuntos de dados para detecção de intrusão contêm informações que são necessárias para estudar os padrões de ataques e as atividades anormais de um sistema de rede [8]. Essas informações são os dados de coleta de entrada e saıda do tráfego de rede.

Ao fornecer os dados do tráfego de rede, os conjuntos de dados possibilitam o treinamento, validação e teste dos modelos de aprendizado de máquina para detecção de intrusão, com o intuito de identificar atividades suspeitas ou maliciosas.

A variedade de cenários e ataques existentes nos conjuntos de dados proporcionam a criação de IDS robustos, capazes de lidar com ameaças desconhecidas [13]. No entanto, a escolha do conjunto de dados adequado para o treinamento, validação e teste do modelo é considerado crucial, pois deve refletir com precisão as condições mais próximas do mundo real. Assim é possível assegurar resultados confiáveis na detecção de intrusão.

De acordo com [25], os conjuntos de dados podem ser criados com base em eventos reais, emulados ou sintéticos. Os criados com base em eventos reais são registrados sobre uma topologia com configuração de rede completa, contendo por exemplo, modens, *firewalls*, *switches*, roteadores, servidores e *hosts* com diferentes sistemas operacionais [8]. Já os conjuntos de dados de eventos emulados são registrados em um ambiente de teste ou de emulação de rede [25].

Por sua vez, os baseados em eventos sintéticos são registrados sobre um tráfego de rede injetado por um gerador de tráfego. A injeção de ataque sintético, também, pode ser usada para introduzir ataques a um conjunto de dados existente, por exemplo, para equilibrar as classes de ataques [8].

Na criação de um conjunto de dados para detecção de intrusão, normalmente, o tráfego de rede é capturado no formato de pacotes ou no formato de fluxos [25].

Os dados no formato de pacotes abrangem informações completas sobre a carga útil do tráfego de rede e são disponibilizados em arquivos PCAP *(Packet CAPture)*. A captura, geralmente, é realizada com o espelhamento das portas nos dispositivos de rede.

Os dados no formato de fluxos possuem informações mais compactas, contendo apenas os metadados das conexões, como endereço IP de origem e destino, porta de origem e destino, protocolos de transporte e aplicação, tempo de duração do fluxo, etc. Dessa forma, os fluxos podem aparecer nas formas unidirecionais ou bidirecionais, contendo os pacotes que compartilham os metadados dentro da janela de tempo do fluxo de rede (ver a Figura 2.4).

Na forma unidirecional são reunidos os pacotes do *host* de origem ao *host* de destino, que compartilham os metadados do fluxo. Na direção contrária, ou seja, do *host* destino ao *host* de origem, os pacotes são reunidos em outro fluxo unidirecional. Enquanto isso, na forma bidirecional são contidos, nos mesmos fluxos, todos os pacotes entre os *hosts* de origem e destino e entre os *hosts* de destino e origem, independente da direção.

Além disso, no tráfego de rede os pacotes ou fluxos de dados são capturados e

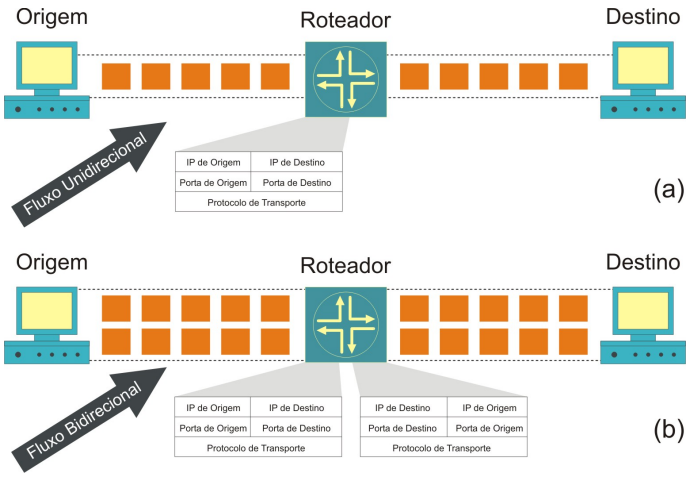

Figura 2.4. Fluxos de rede unidirecional (a) e bidirecional (b)

registrados de forma desordenada e precisam ser pré-processados para a seleção[1] e extração[2] de dados relevantes que possam caracterizar os eventos da rede [18]. Isso permite que os modelos de aprendizado de máquina possam classificar e detectar as anomalias.

Em geral, os conjuntos de dados para detecção de intrusão devem conter os seguintes critérios [34]: (i) tráfego de rede que permita a classificação dos eventos em normais e maliciosos; (ii) tráfego de rede que representa características de cenários reais de redes; (iii) rótulos para classificação dos eventos como normal ou malicioso; (iv) tráfego de rede de cenários distintos; (v) estrutura flexível para fácil atualização ou reprodução, afim de comparação com outros conjuntos de dados; (vi) ausência de dados confidenciais que impossibilite o compartilhamento; (vii) documentação disponível, informando as limitações e os métodos utilizados durante a sua construção.

Os critérios descritos acima são importantes para analisar a qualidade dos conjuntos de dados. Ainda assim, é possível encontrar irregularidades que podem influenciar na qualidade e na confiabilidade dos modelos de classificação para detecção de intrusão, como valores ausentes, ruídos, inconsistências e redundâncias de dados.

As correções para tais irregularidades podem ser realizadas com métodos de pré-processamento de dados [1], que ajudam a mitigar a influência desses problemas nos conjuntos de dados, durante as etapas treinamento, validação e teste.

O pré-processamento de dados é importante na transformação dos dados, tornando os dados mais fáceis de manipular e reduzindo o tempo de execução dos modelos. São empregadas entre as técnicas mais comuns de transformação dos dados: a normalização e a codificação [21].

[1] A seleção de dados é responsável por criar um subconjunto relevante de atributos contidos no conjunto de dados, para uma tarefa específica, com base em determinados critérios ou condições. Isso pode ser feito para melhorar a robustez do modelo, reduzir os custos computacionais ou focar em aspectos específicos dos dados [26].

[2] A extração de dados envolve o processo de recuperação ou extração de informações ou atributos específicos do conjunto de dados, com o objetivo de identificar e isolar atributos ou características relevantes que são cruciais para a tarefa do modelo de classificação [26].

A normalização de dados permite que atributos que contenham instâncias de valores muito baixos, não sejam superados por aqueles que possuem instâncias de valores muito altos. Assim, é realizado o dimensionamento de ambos para um intervalo especificado, como 0.0 e 1.0. A principal vantagem da normalização é a redução de tempo no processamento do modelo, já que as instâncias ficam numa mesma escala numérica reduzida [1]. Algumas técnicas de normalização de dados incluem o *decimal scale*, o *z-score* e o *min-max scale* [22].

Por a sua vez, a codificação é o processo de conversão de dados categóricos em um formato que os modelos podem usar para melhorar a classificação [16]. Dessa forma, se o conjunto de dados possuir dados categóricos, os mesmos são codificados para valores numéricos antes de serem utilizados pelos modelos de classificação.

Para fins de pesquisas existem vários conjuntos de dados para detecção de intrusão disponíveis publicamente para o desenvolvimento de trabalhos que possam contribuir com a evolução dos IDS, como NSL-KDD [32], UNSW-NB15 [19], UGR'16 [15], CIC-IDS2017 [30], CSE-CIC-IDS2018 [30], entre outros.

O NSL-KDD *(National Science Laboratory - Knowledge Discovery in Databases)* foi desenvolvido em 2009, na Universidade de *New Brunswick*, Canadá, como um aprimoramento do conjunto de dados original KDD Cup'99. O NSL-KDD foi criado para abordar os problemas e limitações encontrados KDD Cup'99, visando uma representação mais realista de cenários de intrusão de rede. Entre as novidades foi eliminado a redundância, introduzido novas instâncias de ataques e garantido a distribuição equilibrada dos tipos de ataques. O NSL-KDD compreende quatro tipos principais de ataques: *Denial of Service (DoS), Probe, User to Root (U2R) e Remote to Local (R2L)*. Esses ataques simulam diversas ameaças à segurança, fornecendo um conjunto abrangente e diversificado de cenários para avaliar IDS. O NSL-KDD tornou-se uma referência amplamente utilizada para avaliar a robustez dos modelos de detecção de intrusão.

O UNSW-NB15 foi desenvolvido em 2015, na Universidade de *New South Wales (UNSW)*, Austrália, como um conjunto abrangente de dados de tráfego de rede para pesquisa de detecção de intrusão. O conjunto de dados foi criado capturando o tráfego do mundo real em um ambiente controlado, incluindo atividades normais e maliciosas. O UNSW-NB15 consiste em nove categorias principais de ataque, abrangendo vários tipos de intrusão, como *Fuzzers, Analysis, Backdoors, Denial of Service (DoS, Exploits, Generic, Reconnaissance, Shellcode e Worms*. O seu desenvolvimento teve como consequência fornecer uma representação mais realista dos desafios modernos de segurança de redes, tornando-o um recurso valioso para avaliar e melhorar os IDS.

O UGR'16 foi desenvolvido em 2016, na Universidade de Granada, Espanha, como referência para avaliação de IDS. Foi gerado capturando o tráfego de rede em um ambiente controlado, combinando atividades normais e maliciosas para simular ameaças cibernéticas do mundo real. O UGR'16 inclui uma variedade de ataques, categorizados em diferentes classes, como *Denial of Service (DoS), Distributed Denial of Service (DDoS), Reconnaissance, User to Root (U2R), Remote to Local (R2L)* e vazamentos de dados. O seu desenvolvimento teve como objetivo responder à necessidade de dados diversos e relevantes para avaliar a robustez dos modelos de detecção de intrusão em uma vasta gama de ameaças cibernéticas.

O CIC-IDS2017 foi desenvolvido em 2017, no Instituto Canadense de Segurança Cibernética *(Canadian Institute for Cybersecurity - CIC)*, da Universidade de *New Brunswick*, Canadá. O conjunto de dados foi criado capturando o tráfego de rede em um ambiente controlado para servir como referência para avaliação de IDS. O CIC-IDS2017 foi gerado usando uma variedade de cenários realistas para emular o comportamento normal da rede e diversas ameaças cibernéticas. O conjunto de dados inclui uma grande diversidade de ataques, abrangendo categorias como *Denial of Service (DoS), Distributed Denial of Service (DDoS), Brute Force Attacks, Network Scans* e infecções por *Malwares*. O desenvolvimento do CIC-IDS2017 concentrou-se em fornecer uma amostra abrangente e representativa das ameaças cibernéticas contemporâneas, tornando-o um recurso valioso para pesquisas na área da segurança cibernética.

O CSE-CIC-IDS2018 foi desenvolvido em 2018, no Instituto Canadense de Segurança Cibernética *(Canadian Institute for Cybersecurity - CIC)*, da Universidade de *New Brunswick*, Canadá. Foi criado como parte do projeto Avaliação de Segurança Cibernética *(Cybersecurity Evaluation - CSE)*, com o objetivo de fornecer um conjunto de dados realista e diversificado para avaliar IDS, podendo ser considerado uma atualização do CIC-IDS2017. O CSE-CIC-IDS2018 foi gerado capturando o tráfego de rede em um ambiente controlado, combinando atividades normais e maliciosas. O CSE-CIC-IDS2018 abrange uma ampla gama de ataques, incluindo *Denial of Service (DoS), Distributed Denial of Service (DDoS), Brute Force Attacks*, tentativas de infiltração e várias infecções por *Malwares*. O desenvolvimento do conjunto de dados objetivou refletir as ameaças contemporâneas à cibersegurança, garantindo a sua relevância como referência para avaliar a robustez dos métodos de detecção de intrusão.

2.4. Processo de Desenvolvimento de um IDS baseado em Aprendizado de Máquina

O desenvolvimento de um Sistema de Detecção de Intrusão *(Intrusion Detection System - IDS)* baseado em Aprendizado de Máquina é um processo dinâmico que exige uma compreensão abrangente dos dados, seleção criteriosa de algoritmos e avaliação rigorosa do modelo. A visão geral das etapas básicas é ilustrada na Figura 2.5.

2.4.1. Etapa 1: Seleção do Conjunto de Dados

A seleção do conjunto de dados adequado é fundamental para construção do modelo de detecção de intrusão. Os dados contidos no conjunto irão servir de base para a criação dos padrões desejados para a classificação das instâncias.

O conjunto de dados deve abranger uma gama diversificada de atividades normais e maliciosas para garantir a robustez do modelo na identificação dos padrões. Uma compreensão abrangente dos dados contidos no conjunto, incluindo a documentação com informações sobre o formato (binário ou multiclasse) e as características dos dados, são fundamentais para adaptar os estágios seguintes de desenvolvimento do modelo de detecção de intrusão.

A Seção 2.3 aborda com mais detalhes os conjuntos de dados, inclusive descrevendo alguns exemplos de conjunto de dados públicos encontrados na literatura.

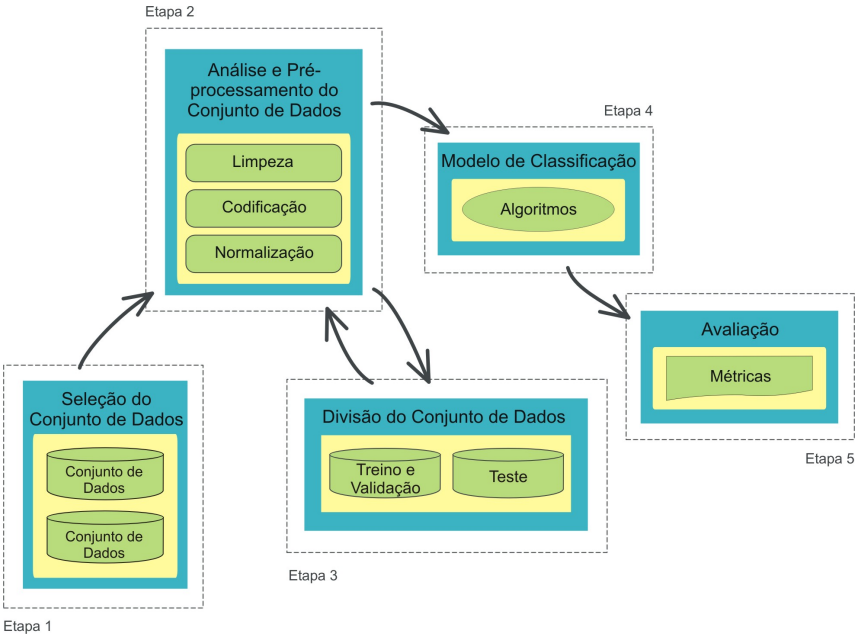

Figura 2.5. Etapas básicas para o desenvolvimento de um IDS baseado em Aprendizado de Máquina.

2.4.2. Etapa 2: Análise e Pré-processamento do Conjunto de Dados

Geralmente, os conjuntos de dados possuem irregularidades que devem ser tratadas. Em sua maioria é possível encontrar valores ausentes, ruídos, inconsistências e redundâncias de dados. Esses problemas podem influenciar na qualidade e na confiabilidade dos modelos de classificação para detecção de intrusão.

Assim a etapa de análise e pré-processamento dos dados é fundamental para a correção desses problemas. Isso envolve a aplicação de métodos consideradas comuns: (i) a limpeza de dados para corrigir problemas como valores ausentes, valores discrepantes e inconsistências; (ii) a codificação que serve para transformar os valores categóricos dos atributos em valores numéricos, visto que dados categóricos não atendem as exigências dos modelos; e (iii) a normalização que é empregada para trazer uniformidade aos valores existentes no conjunto de dados.

Uma boa realização da análise e do pré-processamento de dados pode estabelecer as bases para um modelo robusto, capaz de discernir nuances sutis nos dados, de forma a melhorar a precisão na detecção de intrusão. Alguns métodos de pré-processamento de dados podem ser empregados antes ou após a divisão do conjunto de dados.

A Seção 2.3 aborda, resumidamente, os métodos de pré-processamento de dados empregados no tratamento dos conjuntos de dados, inclusive detalhando os métodos mais comuns.

2.4.3. Etapa 3: Divisão do Conjunto de Dados

Para treinar um modelo de aprendizado de máquina de maneira eficaz, é necessário particionar o conjunto de dados em subconjuntos distintos para treinamento, validação e teste.

Dessa forma, o conjunto de treinamento serve para o modelo aprender os padrões existentes dentro dos dados. Já o conjunto de validação auxilia no ajuste de hiperparâmetros para um bom treinamento do modelo. Por sua vez, o conjunto de teste avalia a generalização e o desempenho do modelo para dados ainda não vistos.

É importante encontrar o equilíbrio certo na divisão dos conjuntos de dados. Assim é possível evitar *overfitting*[3] ou *underfitting*[4], garantindo a robustez do modelo em diferentes cenários.

2.4.4. Etapa 4: Modelos de Classificação

A etapa de escolha do algoritmo de classificação é uma decisão crítica que molda as capacidades de aprendizado do IDS. Existem diferentes algoritmos, como os detalhados na Tabela 2.1, que oferecem pontos fortes e vantagens únicas.

A seleção do algoritmo adequado para o treinamento do modelo de classificação, depende da natureza dos dados e dos requisitos específicos da tarefa de detecção de intrusão. Essa etapa exige um equilíbrio criterioso entre a complexidade do algoritmo, a interpretabilidade e o custo computacional.

A Seção 2.2 aborda com mais detalhes os algoritmos de aprendizado de máquina para a construção de modelos de detecção de intrusão, inclusive descrevendo alguns exemplos.

2.4.5. Etapa 5: Avaliação de desempenho dos modelos

A etapa final é o teste do modelo construído para a detecção de intrusão, que reside na avaliação de desempenho. As métricas mais comuns na avaliação de um IDS são: *Accuracy, Precision, Recall e F1-score*. Essas métricas oferecem *insights* sobre a robustez do modelo na tarefa de classificação das instâncias entre normais e maliciosas.

Além disso, a Matriz de Confusão e a Curva ROC *(Receiver Operating Characteristic)* mostram ainda mais a capacidade do modelo na tarefa de classificação. Essa etapa não apenas valida a confiabilidade do IDS, mas também informa possíveis ajustes e refinamentos para melhorar o seu desempenho futuro.

A Seção 2.2 aborda com mais detalhes as métricas de avaliação, inclusive descrevendo alguns exemplos.

2.5. Problemas e Desafios dos IDS baseados em Anomalias

A literatura mostra que embora existam muitas pesquisas voltadas para a melhoria dos IDS, várias questões ainda precisam ser resolvidas [11, 33]. Os IDS devem ser mais precisos, com a capacidade de detectar uma variedade distinta de intrusões, gerar e atualizar

[3] O *overfitting* é um problema comum no aprendizado de máquina, onde o modelo aprende bem com os dados de treinamento, mas captura os ruídos presentes nos dados. Como resultado, o modelo sobreajustado tende a ter um desempenho ruim em novos dados. Isso ocorre porque o modelo memorizou o conjunto de treinamento, em vez de aprender os padrões gerais que podem ser aplicados as novas instâncias [12].

[4] O *underfitting* ocorre quando o modelo não é complexo o suficiente para aprender as nuances e complexidades do conjunto de dados. Com isso, o resultado é uma falha no ajuste adequado do modelo em relação aos dados de treinamento. Esse problema pode levar o modelo a altas taxas de falsos positivos e falsos negativos, comprometendo a capacidade de detectar e classificar com precisão intrusões na rede [12].

informações sobre novos ataques e emitir menos alarmes falsos.

Um valor de falso positivo (alarme falso) representa o estado em que um evento não é intrusão, mas o modelo classifica erroneamente como intrusão [7]. Dessa forma, é esperado que um IDS robusto tenha a taxa de detecção de intrusão muito alta e a taxa de falsos positivos muito baixa.

Em relação aos conjuntos de dados, alguns estudos apontam que a maioria dos conjuntos de dados públicos não possuem dados suficientes para permitir uma cobertura elevada dos modelos de aprendizado máquina na detecção de diferentes tipos de ataques [25, 18, 10].

Em razão disso, a criação de novos conjuntos de dados para detecção de intrusão que caracterizam ambientes diversificados no tráfego de rede, com relação ao reconhecimento de padrões que permitam ampliar a detecção de ataques e identificar ataques desconhecidos, podem refletir na melhoria dos modelos de aprendizado de máquina. Assim, pode haver um impacto na melhoria dos IDS em relação a detecção de ataques diversificados e a redução de alarmes falsos.

Além disso, é possível encontrar estudos destacando o pré-processamento de dados como uma etapa essencial para melhorar a qualidade dos conjuntos de dados [22, 1, 21]. Isso mostra que é possível explorar os métodos de pré-processamento de dados, visando a criação de novas técnicas que possam diversificar os atributos do tráfego de rede nos conjuntos de dados e, consequentemente, impactar no desempenho dos modelos de aprendizado de máquina na detecção de intrusão.

A seguir, são apresentados algumas limitações e desafios encontrados nos conjuntos de dados para detecção de intrusão, que podem ser explorados em novas pesquisas, para ajudar a melhorar e difundir os IDS baseados em Anomalias:

- A cobertura de ataques é considerado um dos maiores desafios, que impedem os IDS de serem implementados em ambientes de redes reais. Apenas 33%, aproximadamente, dos ataques conhecidos são cobertos pelos conjuntos de dados [8];
- Em relação a simulações que caracterizam estruturas de redes reais, apenas 11% dos IDS usam conjuntos de dados gerados em ambientes reais ou simulados, ou seja, gerados em ambientes que caracterizam tráfegos de rede próximos dos reais [8];
- Os ataques estão evoluindo em um ritmo com o qual os conjuntos de dados não estão conseguindo acompanhar. Assim, novas técnicas de geração de conjuntos de dados são necessárias para mitigar os ataques desconhecidos e ataque de dia zero[5] [29]; e
- A ausência de documentação e informações detalhadas sobre a geração dos conjuntos de dados, contendo as formas de coleta do tráfego de rede, a extração de dados,

[5]Um ataque de dia zero é difícil de prevenir, refere-se à uma vulnerabilidade ainda não explorada. O ataque aproveita a falha de segurança do sistema para o qual nenhuma solução ou defesa foi desenvolvida. Os invasores exploram essas vulnerabilidades antes que os desenvolvedores possam lançar *patches* ou atualizações. Assim é o termo *dia zero*, que indica zero dias de proteção [28].

o pré-processamento de dados e a carga bruta *(Packet Capture - PCAP)* do tráfego, dificultam a comparação de diferentes métodos de detecção de intrusão [5].

Além das limitações e desafios apresentados acima, a Figura 2.6 mostra outras lacunas existentes nos conjuntos de dados para detecção de intrusão, que impactam diretamente nos IDS baseados em Anomalias. Dessa forma, a realização de novas pesquisas para as lacunas nos IDS são cruciais para melhorar as defesas da cibersegurança.

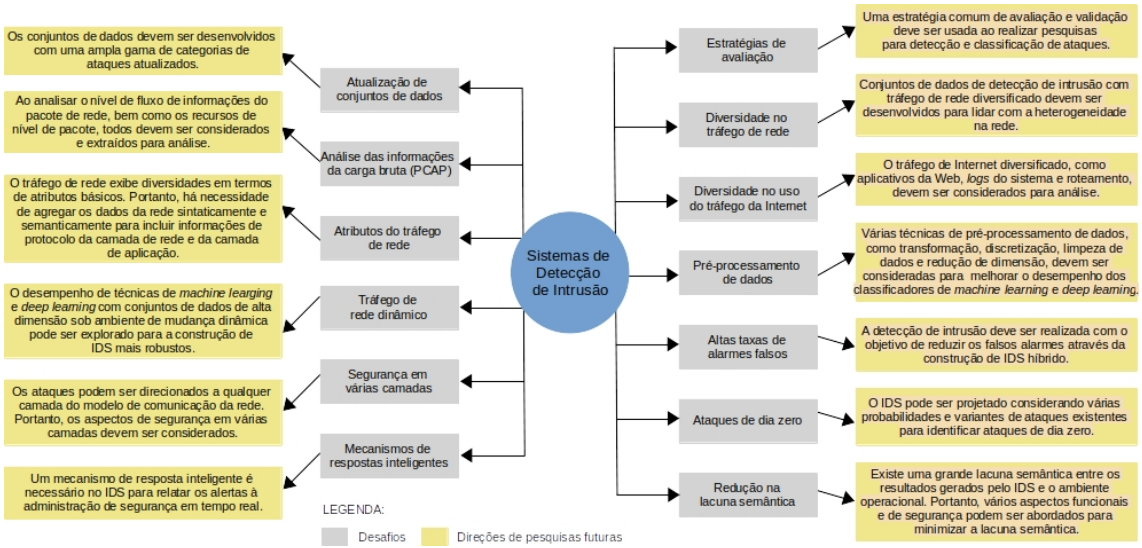

Figura 2.6. Esquema de desafios e direções de pesquisas futuras.

Fonte: Adaptado de [33].

2.6. Materiais e Métodos

Este minicurso visa despertar o interesse dos participantes nos Sistema de Detecção de Intrusão *(Intrusion Detection Systems - IDS)* baseados em anomalias com uso de aprendizado de máquina, ou simplesmente, IDS baseados em Aprendizado de Máquina *(ML-based IDS)*. Para isso, o minicurso foi elaborado para oferecer uma abordagem com *insights* teóricos e aplicações práticas.

A abordagem teórica será interativa, por meio de uma palestra que apresentará material derivado de uma revisão da literatura. Com isso, o principal objetivo é transmitir os fundamentos básicos sobre os seguintes temas apresentados neste material (Figura 2.7):

- Sistemas de Detecção de Intrusão (IDS): descrito na Seção 2.1;
- Aprendizado de Máquina aplicado em IDS: descrito na Seção 2.2;
- Conjuntos de Dados para Detecção de Intrusão: descrito na Seção 2.3;
- Processo de Desenvolvimento de um IDS baseado em Aprendizado de Máquina: descrito na Seção 2.4 ; e
- Problemas e Desafios dos IDS baseados em Anomalias: descrito na Seção 2.5.

Figura 2.7. Temas abordados na apresentação teórica do minicurso.

A abordagem prática também será interativa, por meio de implementação prática com a linguagem de programação Python e bibliotecas de aprendizado de máquina. Com isso, os participantes terão a oportunidade de visualizar a aplicação de algoritmos de aprendizado de máquina em conjuntos de dados para detecção de intrusão, adquirindo experiência prática na construção de IDS baseado em anomalias.

Por se tratar de fundamentos básicos, durante a exploração dos conteúdos práticos, será empregado somente o método de aprendizado supervisionado, que é utilizado com mais frequência nos modelos de IDS para detecção de intrusão. Nesse tipo de método, os conjuntos de dados são rotulados para encontrar a relação entre os dados e suas classes, durante as etapas de treinamento, validação e teste.

Assim, o conjunto de dados CSE-CIC-IDS2018 [30] será empregado na abordagem prática para a construção do modelo de aprendizado de máquina para detecção de intrusão. Trata-se de um conjunto de dados composto pelas seguintes categorias de ataques: *DoS, DDoS, Brute Force, BotNet, Infiltration e Webattack*. Entre essas categorias existem diversas variações.

Além disso, serão aplicadas as técnicas de pré-processamentos mais comuns nos conjuntos de dados para detecção de intrusão, como limpeza de dados, codificação e normalização.

A limpeza de dados é aplicada para lidar com valores ausentes, inconsistentes e redundantes que são comuns em conjuntos de dados. Se esses dados não forem tratados, podem causar ruídos e gerar *outliers* que atrapalham o treinamento e a tomada de decisão do modelo.

Já a codificação é aplicada em atributos com valores categóricos que alguns modelos não podem processar. Com isso, os dados são transformados em valores numéricos para atender aos requisitos do modelo.

Por sua vez, a normalização é aplicada em atributos para reduzir e padronizar es-

calas de dados. Essa técnica elimina ruídos que podem causar *outliers* e ajusta os valores para facilitar a interpretação do modelo.

Em relação ao algoritmos de classificação, serão construídos modelos com o *Random Forest*, que apresenta bons resultados na avaliação de desempenho de conjuntos de dados com método supervisionado. Portanto, para fins de comparação também serão aplicados os algoritmos *kNN (k-Nearest Neighbors)* e *SVM (Support Vector Machine)*.

Por fim, os modelos construídos serão avaliados com as métricas Matriz de Confusão, *Accuracy, Precision, Recall e F1-Score*. Essas métricas são bastante utilizadas na literatura para avaliar modelos de detecção de ataques. A Figura 2.8 ilustra uma visão geral dos métodos utilizados na abordagem prática, já descritos nessa seção.

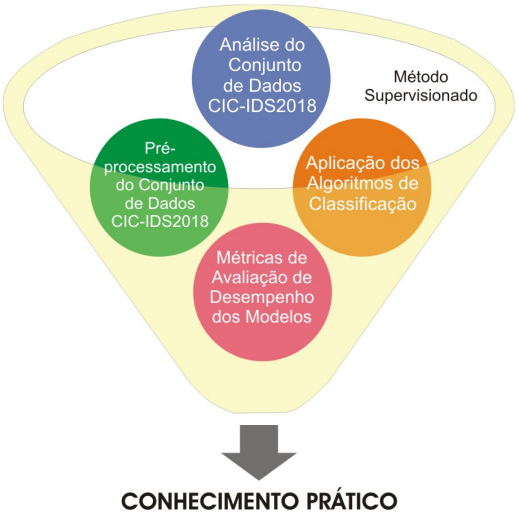

Figura 2.8. Conteúdos aplicados na prática do minicurso.

Em resumo, a abordagens teórica permitirá que os participantes mergulhem nos fundamentos básicos teóricos e compreendam as dificuldades e desafios que esses sistemas enfrentam para serem aplicados no mundo real. Enquanto a abordagem prática promoverá o enriquecimento e o melhor entendimento dos tópicos teóricos abordados.

2.7. Considerações Finais

Ao final do minicurso, os participantes terão uma sólida compreensão das estratégias de detecção de intrusão baseadas em anomalias e sua aplicabilidade na segurança cibernética. Sobretudo, será reforçado, entre os participantes, a importância da atualização contínua das habilidades para enfrentar ameaças em constante evolução e incentivado a exploração mais aprofundada dos tópicos abordados.

Assim, este minicurso ajuda a divulgar conteúdos sobre pesquisas focadas em Sistemas de Detecção de Intrusão *(Intrusion Detection Systems – IDS)* baseados em Anomalias. Dessa forma, melhora a compreensão da evolução das ameaças cibernéticas e da necessidade de mecanismos avançados para detecção de intrusão.

A disseminação da pesquisa promove a colaboração entre profissionais de segurança cibernética, fomentando o desenvolvimento de técnicas de detecção de intrusão

mais robustas e adaptativas. Além disso, a divulgação das pesquisas garantem que as organizações se mantenham informadas sobre as inovações mais recentes.

Portanto, ao partilhar resultados de investigação sobre IDS baseados em Anomalias, a comunidade de segurança cibernética pode contribuir colectivamente para a melhoria das capacidades de detecção de intrusão.

Referências

[1] Ahmad, T. and Aziz, M. N. (2019). Data preprocessing and feature selection for machine learning intrusion detection systems. ICIC Express Lett, 13(2):93–101.

[2] Bezerra, V. H., da Costa, V. G. T., Junior, S. B., Miani, R. S., and Zarpelao, B. B. (2018). One-class classification to detect botnets in iot devices. Anais do XVIII Simpósio Brasileiro de Segurança da Informação e de Sistemas Computacionais (SBSeg), Natal, RN, 22-25 Outubro 2018.

[3] Choraś, M. and Pawlicki, M. (2021). Intrusion detection approach based on optimised artificial neural network. Neurocomputing, 452:705–715.

[4] Dhaliwal, S. S., Nahid, A., and Abbas, R. (2018). Effective intrusion detection system using xgboost. Information, 9(7):1–24.

[5] Engelen, G., Rimmer, V., and Joosen, W. (2021). Troubleshooting an intrusion detection dataset: the cicids2017 case study. 2021 IEEE Security and Privacy Workshops (SPW), San Francisco, CA, USA, 27 May 2011.

[6] García-Teodoro, P., Díaz-Verdejo, J., Maciá-Fernández, G., and Vázquez, E. (2009). Anomaly-based network intrusion detection: Techniques, systems and challenges. Computers Security, 28(1):18–28.

[7] Hajj, S., El Sibai, R., Bou Abdo, J., Demerjian, J., Makhoul, A., and Guyeux, C. (2021). Anomaly-based intrusion detection systems: The requirements, methods, measurements, and datasets. Transactions on Emerging Telecommunications Technologies, 32(4):1–36.

[8] Hindy, H., Brosset, D., Bayne, E., Seeam, A. K., Tachtatzis, C., Atkinson, R., and Bellekens, X. (2020). A taxonomy of network threats and the effect of current datasets on intrusion detection systems. IEEE Access, 8:104650–104675.

[9] International Telecommunication Union (2021). Global Cybersecurity Index 2020: Measuring commitment to cybersecurity. ITUPublications, Geneva, Switerland, 1 edition.

[10] Kenyon, A., Deka, L., and D., E. (2020). Are public intrusion datasets fit for purpose characterising the state of the art in intrusion event datasets. Computers Security, 99:1–26.

[11] Khraisat, A., Gondal, I., Vamplew, P., and Kamruzzaman, J. (2019). Survey of intrusion detection systems: techniques, datasets and challenges. Cybersecurity, 2(20):1–22.

[12] Koehrsen, W. (2018). Overfitting vs. underfitting: A complete example. Towards Data Science, 405.

[13] Layeghy, S. and Portmann, M. (2022). On generalisability of machine learning-based network intrusion detection systems. arXiv preprint arXiv:2205.04112, [s.n.]:1–12.

[14] Liu, H. and Lang, B. (2019). Machine learning and deep learning methods for intrusion detection systems: A survey. Applied Sciences, 9(20):1–28.

[15] Maciá-Fernández, G., Camacho, J., Magán-Carrión, R., García-Teodoro, P., and Theró, R. (2018). Ugr'16: A new dataset for the evaluation of cyclostationarity-based network idss. Computers Security, 73:411–424.

[16] Mahfouz, A., Abuhussein, A., Venugopal, D., and Shiva, S. (2020). Ensemble classifiers for network intrusion detection using a novel network attack dataset. Future Internet, 12(11):1–19.

[17] Martínez Torres, J., Iglesias Comesaña, C., and García-Nieto, P. J. (2019). Machine learning techniques applied to cybersecurity. International Journal of Machine Learning and Cybernetics, 10:2823–2836.

[18] Molina-Coronado, B., Mori, U., Mendiburu, A., and Miguel-Alonso, J. (2020). Survey of network intrusion detection methods from the perspective of the knowledge discovery in databases process. IEEE Transactions on Network and Service Management, 17(4):2451–2479.

[19] Moustafa, N. and Slay, J. (2015). Unsw-nb15: a comprehensive data set for network intrusion detection systems (unsw-nb15 network data set). Paper presented at the 2015 Military Communications and Information Systems Conference (MilCIS), Canberra, ACT, Australia, 10-12 November 2015.

[20] Naseer, M., Rusdi, J. F., Shanono, N. M., Salam, S., Muslim, Z. B., Abu, N. A., and Abadi, I. (2021). Malware detection: issues and challenges. Cybersecurity, 1807(1):1–6.

[21] Obaid, H. S., Dheyab, S. A., and Sabry, S. S. (2019). The impact of data pre-processing techniques and dimensionality reduction on the accuracy of machine learning. Paper presented at the 9th Annual Information Technology, Electromechanical Engineering and Microelectronics Conference (IEMECON), Jaipur, India, 13–15 March 2019.

[22] Paulauskas, N. and Auskalnis, J. (2017). Analysis of data pre-processing influence on intrusion detection using nsl-kdd dataset.

[23] Putra, W. and Huang, J. J. (2019). A survey of intrusion detection system. International Journal of Informatics and Computation, 1(1):1–19.

[24] Resende, P. A. A. and Drummond, A. C. (2018). A survey of random forest based methods for intrusion detection systems. ACM Computing Surveys (CSUR), 51(3):1–36.

[25] Ring, M., Wunderlich, S., Scheuring, D., Landes, D., and Hotho, A. (2019). A survey of network-based intrusion detection data sets. Computers Security, 86:147–167.

[26] Salih, A. A. and Abdulrazaq, M. B. (2019). Combining best features selection using three classifiers in intrusion detection system. 2019 International Conference on Advanced Science and Engineering (ICOASE), Zakho - Duhok, Iraq, 02-04 April 2019.

[27] Saranya, T., Sridevi, S., Deisy, C., Chung, T. D., and Khan, M. A. (2020). Performance analysis of machine learning algorithms in intrusion detection system: A review. Procedia Computer Science, 171:1251–1260.

[28] Sarhan, M., Layeghy, S., Gallagher, M., and Portmann, M. (2023). From zero-shot machine learning to zero-day attack detection. International Journal of Information Security, 22:947–959.

[29] Sarker, I. H., Kayes, A., Badsha, S., Alqahtani, H., Watters, P., and Ng, A. (2020). Cybersecurity data science: an overview from machine learning perspective. Journal of Big data, 7(1):1–29.

[30] Sharafaldin, I., Lashkari, A. H., and Ghorbani, A. A. (2018). Toward generating a new intrusion detection dataset and intrusion traffic characterization. Paper presented at the 4th International Conference on Information Systems Security and Privacy (ICISSp), Funchal, Madeira, Portugal, 22–24 January 2018.

[31] Shenfield, A., Day, D., and Ayesh, A. (2018). Intelligent intrusion detection systems using artificial neural networks. Ict Express, 4(2):95–99.

[32] Tavallaee, M., Bagheri, E., Lu, W., and Ghorbani, A. A. (2009). A detailed analysis of the kdd cup 99 data set. Paper presented at the 2009 IEEE Symposium on Computational Intelligence for Security and Defense Applications, Ottawa, ON, Canada, 08-10 July 2009.

[33] Thakkar, A. and Lohiya, R. (2022). A survey on intrusion detection system: feature selection, model, performance measures, application perspective, challenges, and future research directions. Artificial Intelligence Review, 55:453–563.

[34] Viegas, E. K., Santin, A. O., and Oliveira, L. S. (2017). Toward a reliable anomaly-based intrusion detection in real-world environments. Computer Networks, 127:200–216.

[35] Zaman, M. and Lung, C. (2018). Evaluation of machine learning techniques for network intrusion detection. NOMS 2018-2018 IEEE/IFIP Network Operations and Management Symposium, Taipei, Taiwan, 23-27 April 2018.