Aplicação de Aprendizado de Máquina na Detecção de Intrusão em Redes de Computadores

Carla Cursino[1] , **José Marcos Gomes**[1] , **Ana Carolina Lorena**[1] , **Filipe Alves Neto Verri**[1] , **Luiz Alberto Vieira Dias**[1]

[1]Instituto Tecnológico de Aeronáutica - ITA
Divisão da Ciência da Computação - IEC
São José dos Campos/SP - Brasil

cursino@ita.br, gomesjm@ita.br, aclorena@ita.br, verri@ita.br, vdias@ita.br

Resumo. *Sistemas de detecção de intrusão, os chamados IDS, trabalham com um sofisticado conjunto de regras e analisam o tráfego de rede em busca de padrões de ataques conhecidos. Os sistemas mais modernos utilizam técnicas de análise de tráfego de redes e procuram analisar o comportamento de usuários e agentes de rede.*
Uma das limitações de tais sistemas é que as regras para detectar intrusão devem ser descritas de antemão para então, analisando o tráfego de rede, o sistema detectar anomalias.
Propomos aplicar métodos de aprendizado de máquina para que o sistema possa antecipar comportamentos anômalos sem a necessidade de existirem regras previamente definidas.

1. Introdução

Sistema de Detecção de Intrusos - do inglês: "*Intrusion Detection System*" - é um dispositivo ou *software* monitora uma rede ou sistemas buscando e alertando os operadores na ocorrência de atividades maliciosas ou violações de políticas de segurança. Estes "sistemas de vigilância e monitoração de ameaças computadorizados" podem detectar comportamentos "Suspeitos" tais como: "acesso fora do horário usual", "frequência de uso além do costumeiro", "acesso não usual à dados e programas", e "accesso excessivo no volume de dados" [4]. Dada a importância vital de sistemas computadorizados em nossas vidas, cada vez mais atenção tem sido dada à ferramentas IDS, tornando este um componente de grande importância em sistemas de segurança [11].

Já as técnicas de invasão se apresentam cada vez mais avançadas, e como consequência os métodos tradicionais baseados em assinaturas ou regras de especialistas não são suficientes [16] para detectar invasões. Com o passar dos anos, ferramentas IDSs baseados em técnicas de aprendizado de máquina tem sido desenvolvidas [11].

Pretendemos com este trabalho avaliar abordagens de aprendizado de máquina e determinar o desempenho de sistemas preditivos.

2. Revisão da Literatura

Encontramos na literatura abordagens diferentes utilizando aprendizado de máquina aplicada à IDSs: o aprendizado supervisionado no qual modelos tentam distinguir entre o tráfego comum do malicioso, e o aprendizado não supervisionado que busca detectar anomalias dentro do tráfego, e o aprendizado semi-supervisionado, onde uma grande quantidade de dados não classificados está disponível juntamento com os dados previamente classificados.

2.1. Abordagens Supervisionadas, Semi-supervisionadas e Não Supervisionadas

O uso de aprendizado supervisionado tem sido amplamente utilizado em sistemas IDS [8] e este método busca classificar o tráfego de rede partindo de um conjunto de dados previamente rotulado de "Normal", "Suspeito" (e possivelmente "desconhecido").

MÉTODO	ABORDAGEM	DESCRIÇÃO
Extra Trees	Supervisionado	Implementa um meta estimador treinado sobre um número de árvores de decisão aleatórias ("*extra trees*") sobre vários subconjuntos dos dados e calcula a média para melhorar a acurácia de previsão e controlar sobreajustamento.
KNN	Supervisionado	Classificador implementando votação em k vizinhos mais próximos.
Bagging	Supervisionado	Um meta-classificador composto que agrupa classificadores de base a subconjuntos aleatórios dos dados originais e agrega as previsões individuais (por votação ou por média) para formar uma previsão final.
Random Forest	Supervisionado	Meta classificador que agrupa vários classificadores de árvores de decisão em várias subamostras do conjunto de dados e usa a média para melhorar a precisão e o controle do excesso de ajustes.
Decision Tree	Supervisionado	Método que prevê o valor de uma variável alvo, aprendendo regras simples de decisão inferidas a partir das características dos dados.
Gradient Boosting	Supervisionado	Constrói um modelo aditivo de modo progressivo que permite a otimização de funções de perda diferenciadas arbitrárias.
Ada Boost	Supervisionado	Meta-classificador que se inicia com um classificador no conjunto de dados original e depois adiciona cópias do classificador sobre o mesmo conjunto de dados, onde os pesos das instâncias classificadas incorretamente são ajustados de tal forma que os classificadores subsequentes se concentram mais nos casos difíceis.
One Class SVM	Não-supervisionado	Detector de pontos fora da curva não supervisionado que implementa Máquina de Suporte à Vetores.

Tabela 1. Métodos abordados neste estudo

Soluções propostas para resolver o problema de detecção de anomalias por meio da identificação de "*outliers*" e "*inliers*" podem ser divididas nestas três sub-categorias [1]:

1. **Supervisionadas** - *Lidam com os casos onde o conjunto de dados de treinamento é fornecido com ambos os rótulos ("outliers" e "inliers");*
2. **Semi-supervisionada** - *Requer apenas uma classe "pura" rotulada "inlier" ou "Normal";*
3. **Não-supervisionada** - *Lida com dados completamente não rotulados e misturados de "inliers" e "outliers".*

Para este estudo previlegiamos algoritmos computacionalmente menos exigentes e que viabilizariam a implementação de análise em tempo real de eventos de rede.

2.2. Modelos

2.2.1. Lineares

Para funcionarem adequadamente num modelo linear, os dados precisam ser altamente correlacionados e quando os dados não o forem e altamente agrupados em certas regiões este método é ineficaz. Estudos sugerem que correlações podem ser específicas para determinadas localidades de dados e neste caso subespaços globais localizados por PCA por exemplo são subótimas para detecção de *outliers* [3].

2.2.2. Baseados em Proximidade

Num problema multidimensional como o de detecção de anomalias em eventos de redes de computadores os pontos acabam por se mostrarem equidistantes um do outro e assim o contraste de diferenças é perdido [2, 9].

2.2.3. Probabilísticos

Modelos paramétricos são muito suscetíveis à ruidos e sobreajustes. A escolha incorreta de parâmetros pode levar à classificação de dados espúrios como *outliers* ou quando o

AMOSTRAS	ARQUIVO
172.838	CIDDS-001-external-week1.csv
159.374	CIDDS-001-external-week2.csv
153.027	CIDDS-001-external-week3.csv
186.005	CIDDS-001-external-week4.csv
8.451.521	CIDDS-001-internal-week1.csv
10.310.734	CIDDS-001-internal-week2.csv
6.349.784	CIDDS-001-internal-week3.csv
6.175.898	CIDDS-001-internal-week4.csv
31.959.181	TOTAL

Tabela 2. Conjuntos de dados CIDDS

modelo é muito genérico o número de parâmetros necessários para descrevê-lo se torna proibitivo fazendo com que *outliers* se percam em resultado de sobreajuste e na redução do sobreajuste para resolver o problema acabamos incorrendo em subajuste.

3. Materiais e Métodos

3.1. Conjunto de Dados

Para este trabalho utilizamos o conjunto de dados **CIDDS-001** ("*Coburg Intrusion Detection Data Sets*"), cujo conceito é o de criar um conjunto de dados para avaliação de sistemas de detecção de intrusão [17, 18].

O conjunto de dados é composto por dois grupos de arquivos, um de acessos de origem externa e outro de origem interna. Cada grupo está segmentado em arquivos de captura de dados contendo uma semana de observações, num total de oito arquivos no formato **CSV**.

Dado o grande número de dados, trabalhamos com uma amostragem dos dados totais.

- `Date first seen` - `object` - Data e hora do início da sessão
- `Duration` - `float64` - Duração da sessão (em milisegundos)
- `Proto` - `object` - Protocolo
- `Src IP Addr` - `object` - IP de origem
- `Src Pt` - `int64` - Porta de origem
- `Dst IP Addr` - `object` - IP de destino
- `Dst Pt` - `float64` - Porta de destino
- `Packets` - `int64` - Número de pacotes
- `Bytes` - `object` - Número de bytes
- `Flows` - `int64` - Quantidade de fluxos de transmissão
- `Flags` - `object` - Indicadores TCP
- `ToS` - `int64` - Tipo de serviço
- `class` - `object` - Classificação do ataque
- `attackType` - `object` - Tipo de ataque
- `attackID` - `object` - Identificação do ataque
- `attackDescription` - `object` - Descrição do ataque

3.2. Técnicas

Para tratar a natureza desbalanceada dos dados classificados (ver Figura 1), foi aplicado tanto "*Upsample*" da classe minoritária quanto "*Downsample*" da classe majoritária e os desempenhos dos métodos de classificação foram comparados.

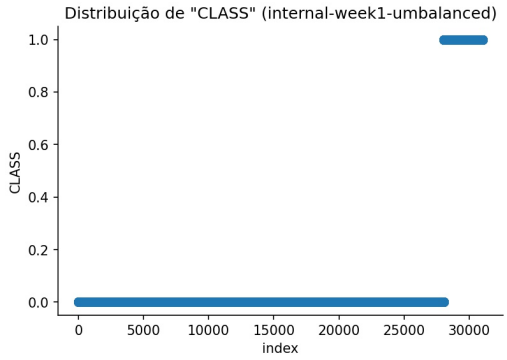

Figura 1. Distribuição da classe

4. Experimentos e Resultados

Curva "*Receiver Operating Characteristic*" (**ROC**) é uma ferramenta importante de diagnóstico do desempenho de algoritmos de aprendizado de máquina que nos mostra a taxa de verdadeiros positivos contra falsos negativos. A área sob a curva **ROC** é chamada de **AUC** é uma medida de previsibilidade do algorítmo. Um **AUC** mais alto indica uma previsão mais apurada.

4.1. Exploração de Dados

4.1.1. Atributos Mais Significativos

3. *Proto* - Protocolos de comunicação
11. *Flags* - Indicadores TCP - cada bit dos 16 (apenas 8 estão em uso) são reservados para este atributo e identificados como:
    - **CWR** - bit 7 - "*Congestion Window Reduction*" - identificado pela letra **C**
    - **ECE** - bit 6 - "*ECN (Explicit Congestion Notification) Capable*" - identificado pela letra **E**
    - **URG** - bit 5 - "*Urgent*" - identificado pela letra **U**
    - **ACK** - bit 4 - "*Acknowledgement*" - identificado pela letra **A**
    - **PSH** - bit 3 - "*Push*" - identificado pela letra **P**
    - **RST** - bit 2 - "*Reset*" - identificado pela letra **R**
    - **SYN** - bit 1 - "*Synchronization*" - identificado pela letra **S**
    - **FIN** - bit 0 - "*Finished*" - identificado pela letra **F**

    Valores que estão fora do padrão determinados pela IETF e documentados pelo IEEE no conjunto de dados deverão ser traduzidos:
    - `0xDB - 11011011 - CE.AP.SF`
    - `0xD2 - 11010010 - CE.A..S.`
    - `0xC2 - 11000010 - CE....S.`
    - `0xDA - 11011010 - CE.AP.S.`
    - `0xD7 - 11010111 - CE.A.RSF`
    - `0x53 - 01010011 - .E.A..SF`
    - `0xDF - 11011111 - CE.APRSF`
    - `0xD6 - 11010110 - CE.A.RS.`
    - `0xD3 - 11010011 - CE.A..SF`
13. *class* - Classificação (atributo alvo)

	DURATION	PROTOCOL	PACKETS	BYTES	FLAGS	CLASS
count	8.451520e+06	8451520	8.451520e+06	8451520	8451520	8451520
unique	0.000000e+00	4	0.000000e+00	89693	20	3
top	0.000000e+00	TCP	0.000000e+00	66	.A....	normal
freq	0.000000e+00	7393818	0.000000e+00	2279907	2652182	7010897
mean	1.141597e-01	0	1.503053e+01	0	0	0
std	7.683694e-01	0	9.768317e+02	0	0	0
min	0.000000e+00	0	1.000000e+00	0	0	0
25%	0.000000e+00	0	1.000000e+00	0	0	0
50%	0.000000e+00	0	2.000000e+00	0	0	0
75%	2.500000e-02	0	4.000000e+00	0	0	0
max	2.244120e+02	0	2.087680e+05	0	0	0

Tabela 3. Estatísticas dos dados brutos

FEATURE	KURTOSIS
DURATION	3035.547966
PROTOCOL	2.521929
PACKETS	13423.894329
BYTES	25936.752752
FLAGS	-1.380831
CLASS	5.502495

Tabela 4. Curtose

Os protocolos ICMP [12] e GRE [7] não são geralmente iniciados ou recebidos em comunicações normais e podem ser aplicados por vetores de ataque e não podem ser negligenciados. Já os protocolos TCP [13] e UDP [15] representam toda a comunicação via Internet com que os usuários costumam interagir normalmente. Endereços e portas IP também serão desconsiderados, mesmo porque atacantes costumam mascarar seus endereços e o uso de NAT pela maioria das redes para publicar serviços externos limita a utilidade de analisar esta informação [6].

4.1.2. Análise de Dispersão e Distribuição

Comparando a curtose (ver Tabela 4) com a média e a mediana (ver Tabela 7), observamos aquela bem acima destas e um grande número de objetos ocorrem frequentemente fora da distribuição "Normal". A obliquidade de todos estes atributos é positiva (ver Tabela 5), e a maioria dos valores tendem ao mínimo e está associada ao grande desvio padrão observado nos atributos "*Duration*" e "*Bytes*".

4.2. Pré-processamento

Tratamos os dados de forma uniforme para todos os algoritmos testados e geramos três conjuntos de dados:

1. Não balanceado - onde foram preservados os dados originais e aplicadas apenas normalização de valores;

FEATURE	SKEW
DURATION	47.795086
PROTOCOL	2.041001
PACKETS	111.699528
BYTES	157.458199
FLAGS	0.235055
CLASS	2.739003

Tabela 5. Obliquidade

DURATION	PROTOCOL	PACKETS	BYTES	FLAGS	CLASS
0	1	1	66	4	0

Tabela 6. Moda

	DURATION	PROTOCOL	PACKETS	BYTES	FLAGS	CLASS
count	31029.000000	31029.000000	31029.000000	3.102900e+04	31029.000000	31029.000000
mean	0.001224	0.567550	0.000102	5.501999e-05	0.380647	0.096200
std	0.013110	0.172793	0.007367	5.950411e-03	0.269249	0.294871
min	0.000000	0.000000	0.000000	0.000000e+00	0.000000	0.000000
25%	0.000000	0.500000	0.000000	6.934413e-08	0.222222	0.000000
50%	0.000000	0.500000	0.000006	2.600405e-07	0.222222	0.000000
75%	0.000000	0.500000	0.000018	1.262641e-06	0.666667	0.000000
max	1.000000	1.000000	1.000000	1.000000e+00	1.000000	1.000000

Tabela 7. Estatísticas após o pré-processamento (dados originais)

2. *Upsample* - onde aplicamos *Upsampling* da classe minoritária; e
3. *Downsample* - onde aplicamos *Downsampling* da classe majoritária.

4.3. Experimentos

Nas Tabelas 8 a 10 comparamos os resultados dos experimentos.

Aplicamos sobre os conjunto de dados os algoritmos (ver Tabela 1), divididos em dois grupos, os **Supervisionados** (onde selecionamos 7 algoritmos, sendo cinco *Ensembles*, 1 de *Árvore* de decisão, e 1 de *Proximidade*) e um **Não supervisionado** dentro do subgrupo de *Support Vector Machine*. O principal fator de escolha dos algoritmos é o de baixo custo computacional, e assim sendo, desprezamos Redes Neurais e outros que podem vir a ser computacionalmente intensos.

5. Discussão

Os resultados obtidos e listados nas Tabelas 8 a 10 mostram uma "*acurácia*" (ACC) bastante próxima em todos os modelos experimentados (entre 98.6% com **Ada Boost** a até 99.9% com **KNN**, todos aplicados sobre dados não balanceados).

Comparando os algoritmos entre si, **Extra Trees** se destacou aplicada à dados balanceados com *Upsampling* e *Downsampling* ou não balanceados, com acurácia de 99.5%, 99.7% e 99.8% respectivamente. Curiosamente **Gradient Boosting** destacou-se aplicando-se *Upsampling* aos dados, com 99.5%, porém ficou ligeiramente aquém das demais aos aplicarmos *Downsampling*, com 99.1%.

No geral todos os algoritmos se mostraram consistentes e indiferentes ao tratamento às classes das observações da amostra de dados e o impacto da distribuição de

Algoritmo	Acurácia	Precisão	AUC
KNN	0.999355	0.995871	0.996528
Extra Trees	0.998550	0.995510	0.996084
Random Forest	0.997261	0.994932	0.995373
Bagging	0.997100	0.994860	0.995284
Decistion Tree	0.996777	0.994716	0.995107
Gradient Boosting	0.992749	0.992579	0.992887
Ada Boost	0.986465	0.983349	0.983968

Tabela 8. Semana 1 - não balanceado

Algoritmo	Acurácia	Precisão	AUC
Extra Trees	0.995812	0.991809	0.996013
Gradient Boosting	0.995812	0.991809	0.996013
Random Forest	0.995812	0.991809	0.996013
KNN	0.995812	0.991809	0.996013
Bagging	0.994975	0.991854	0.995131
Decistion Tree	0.994975	0.991854	0.995131
Ada Boost	0.993300	0.992960	0.993283

Tabela 9. Semana 1 - "*Upsampling*"

Algoritmo	Acurácia	Precisão	AUC
Extra Trees	0.997504	0.997149	0.997504
KNN	0.997237	0.996965	0.997236
Bagging	0.996613	0.996072	0.996612
Random Forest	0.996523	0.995894	0.996523
Decistion Tree	0.995097	0.993032	0.995095
Gradient Boosting	0.991442	0.987837	0.991439
Ada Boost	0.984667	0.980849	0.984663

Tabela 10. Semana 1 - "*Downsampling*"

classes não foi significativo sobre os algoritmos experimentados. Creditamos isto à natureza do conjunto de dados que não possui um espectro muito variado de ataques disponíveis (para gerar o tráfego malicioso são simulados ataques do tipo *Denial of Service (DoS)*, ataques de força bruta e escrutíno de portas [17, 18]).

Comparando o desempenho com o algoritmo não supervisionado **One Class SVM** aplicado aos dados não balanceados, que pode ser observado na Tabela 11, atingiu acurácia de 80.5% (a mais baixa) a até 91.7% com diferentes *kernels* (tanto "**Linear**" quanto "**Sigmoid**" apresentaram o mesmo desempenho, enquanto que "**RBF**" foi inferior), onde $\nu = 0.125$ para todos os casos.

A pontuação F_1 é outra medida de verificação da "*acurácia*" de um modelo, dada pela média harmônica entre a precisão e *recall*:

$$F_1 = \frac{TP}{TP + \frac{1}{2}(FP + FN)},$$

onde:

- TP: *True Positive* - Verdadeiro Positivo;
- FP: *False Positive* - Falso Positivo; e
- FN: *False Negative* - Falso Negativo.

A pontuação F_1 de observações classificadas como "**Suspeitas**" é significativamente inferior (36.5% usando "**RBF**" e 28.2% para os demais "*kernels*") à das classificadas como "**Normais**" (88.5% "**RBF**" e 95.6% para os demais "*kernels*"), o que indica uma tendência do modelo a identificar como "Suspeitos" eventos de classe "Normal".

Algoritmo	Acurácia	Precisão	AUC	F1 (Normal)	F1 (Suspeito)	MCC (Normal)	MCC (Suspeito)
One Class SVM (Linear)	0.917499	0.217552	0.584388	0.956232	0.282913	0.332133	0.332133
One Class SVM (Sigmoid)	0.917499	0.217552	0.584388	0.956232	0.282913	0.332133	0.332133
One Class SVM (RBF)	0.805833	0.648940	0.714537	0.885380	0.365456	0.304000	0.304000

Tabela 11. Semana 1 - não supervisionado

MCC	Interpretação
$[0.0, 0.3)$	*Negligenciável*
$[0.3, 0.5)$	*Fraca*
$[0.5, 0.7)$	*Moderada*
$[0.7, 0.9)$	*Forte*
$[0.9, 1.0]$	*Muito Forte*

Tabela 12. Interpretação do coeficiente de correlação *Matthews*

Aplicamos o coeficiente de correlação *Matthews* como uma medida da qualidade da classificação binária de nosso modelo. Esta medida leva em consideração verdadeiros e falsos positivos e negativos e é considerada uma medida balanceada em classes desbalanceadas, que retorna valores entre $+1$ (uma previsão perfeita) e -1 (discordância total entre observação e previsão), enquanto que 0 indica que a previsão é tão boa quanto uma previsão aleatória [10]:

$$|MCC| = \sqrt{\frac{x^2}{n}},$$

onde n é número total de observações, ou a partir da matriz de confusão:

$$MCC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP) \times (TP + FN) \times (TN + FP) \times (TN + FN)}},$$

onde:

- TP: *True Positive* - Verdadeiro Positivo;
- TN: *True Negative* - Verdadeiro Negativo;
- FP: *False Positive* - Falso Positivo; e
- FN: *False Negative* - Falso Negativo.

A pontuação MCC nos dá o "**produto - momento**" do coeficiente de correlação *Pearson* (diferente do próprio coeficiente de correlação *Pearson* que mede a relação entre os coeficientes), que com o valor 30.4% (para ambas as classes) usando "*kernel*" "**RBF**" e 33.2% para os demais é interpretado como uma previsão "**Fraca**" e próxima de "*Negligenciável*" (ver Tabela 12) [14].

6. Conclusões e trabalhos futuros

Apresentamos uma análise de modelos de aprendizado aplicado ao conjunto de dados de estudos em intrusão de redes **CIDDS**. Obtivemos sucesso na identificação de eventos de invasão com uma acurácia de 99% utilizando modelos supervisionados.

As diferenças entre os conjuntos de dados tratados ou não com *Upsampling* e *Downsampling* não foram significativas o suficiente para justificarem este tratamento que pode ser custoso para este caso particular e uma análise detalhada de conjuntos de dados não balanceados deveria ser aplicada antes [5].

Comparamos com um modelo de detecção não supervisionado que apresentou uma acurácia de 80%, porém como podemos observar após uma análise mais detalhada

utilizando o coeficiente de correlaçao *Matthews*, atingiu uma pontuação entre 30% e 33%, considerado um fator de previsão "**Fraco**".

Modelos de detecção utilizando algoritmos não supervisionados podem vir a ser explorado com o intuito de obtermos índices de acurácia próximos dos modelos preditivos supervisionados, e com fatores de previsão melhores que os apresentados pelo algoritmo **One Class SVM** neste exercício.

7. Siglas

GRE Encapsulamento de Roteamento Genérico - do inglês: "*Generic Routing Encapsulation*" - é o protocolo de tunelamento desenvolvido pela Cisco Systems (fabricante de dispositivos de rede) que permite embutir vários outros protocolos dentro de conexões de rede ponto-a-ponto. 5

ICMP Protocolo de Controle de Mensagens Internet - do inglês: "*Internet Control Message Protocol*" - é um protocolo utilizado por dispositivos de rede para envio de mensagens de rede e informações operacionais 5

IDS Sistema de Detecção de Intrusos - do inglês: "*Intrusion Detection System*" - é um dispositivo ou *software* monitora uma rede ou sistemas buscando e alertando os operadores na ocorrência de atividades maliciosas ou violações de políticas de segurança 1

IEEE Instituto dos Engenheiros de Elétrica e Eletrônica - do inglês: "*Institute of Electrical and Electronics Engineers*" - associação profissional de engenheiros de eletrônica e elétrica e disciplinas associadas 4

IETF Força Tarefa de Engenharia da Internet - do inglês: "*Internet Engineering Task Force*" - uma organização de padrões abertos que trabalha voluntariamente para padronizar a adoção do protocolo TCP/IP na Internet 4

IP Endereço Protocolo Internet - do inglês: "*Internet Protocol Address*" - é um código numérico assinalado a cada dispositivo conectado à rede Internet 5

NAT Tradução de endereço de Rede - do inglês: "*Network Address Translation*" - é um método de mapeamento de endereço de rede em outro modificando informações do cabeçalho IP 5

PCA Análise do Componente Principal - do inglês: "*Principal Component Analysis*" - é o processo de computar os componentes principais e utilizá-los para executar uma mudança de base (mudança relativa de coordenadas no espaço) dos dados. 2

TCP Protocolo de Controle de Transmissão - do inglês "*Transmission Control Protocol*" - um dos principais protocolos do conjunto implementado pela Internet que provê transmissão de cadeias de dados de forma ordenada, com correção de erros e assegurando a entrega 5

UDP Protocolo de Datagrams do Usuário - do inglês "*User Datagram Protocol*" - um dos principais protocolos do conjunto implementado pela Internet e utilizado para transmitir mensagens de aplicações 5

8. Glossário

bit Unidade básica de informação em computação e comunicação digital, da abreviação do inglês - "*binary digit*" - e que representa um estado lógico de dois possíveis valores, comumente sendo "0" ou "1" 4

byte Unidade de informação digital comumente composta por oito bits 3

9. Referências

[1] AGGARWAL, C. C. Outlier analysis second edition. In *Data mining* (2016), Springer.

[2] AGGARWAL, C. C., HINNEBURG, A., AND KEIM, D. A. On the surprising behavior of distance metrics in high dimensional space. In *International conference on database theory* (2001), Springer, pp. 420–434.

[3] AGGARWAL, C. C., AND YU, P. S. Finding generalized projected clusters in high dimensional spaces. In *Proceedings of the 2000 ACM SIGMOD international conference on Management of data* (2000), pp. 70–81.

[4] ANDERSON, J. P. Computer security threat monitoring and surveillance, james p. *Anderson Co., Fort Washington, PA* (1980).

[5] BARELLA, V. H., GARCIA, L. P., DE SOUTO, M. C., LORENA, A. C., AND DE CARVALHO, A. C. Assessing the data complexity of imbalanced datasets. *Information Sciences 553* (2021), 83–109.

[6] DE CARVALHO BERTOLI, G., PEREIRA, L. A., VERRI, F., MARCONDES, C., SANTOS, A. L., AND SAOTOME, O. Abordagem fim-a-fim para uso de aprendizado de máquina em ids–caso de detecç ao stateless para tcp scan.

[7] FARINACCI, D., LI, T., HANKS, S., MEYER, D., AND TRAINA, P. Rfc2784: Generic routing encapsulation (gre), 2000.

[8] HE, Z., ZHANG, T., AND LEE, R. B. Machine learning based ddos attack detection from source side in cloud. In *2017 IEEE 4th International Conference on Cyber Security and Cloud Computing (CSCloud)* (2017), IEEE, pp. 114–120.

[9] HINNEBURG, A., AGGARWAL, C. C., AND KEIM, D. A. What is the nearest neighbor in high dimensional spaces? In *26th Internat. Conference on Very Large Databases* (2000), pp. 506–515.

[10] MATTHEWS, B. W. Comparison of the predicted and observed secondary structure of t4 phage lysozyme. *Biochimica et Biophysica Acta (BBA)-Protein Structure 405*, 2 (1975), 442–451.

[11] MILENKOSKI, A., VIEIRA, M., KOUNEV, S., AVRITZER, A., AND PAYNE, B. D. Evaluating computer intrusion detection systems: A survey of common practices. *ACM Computing Surveys (CSUR) 48*, 1 (2015), 1–41.

[12] POSTEL, J. Ietf rfc 0792: Internet control message protocol, 1981.

[13] POSTEL, J., ET AL. Transmission control protocol.

[14] POWERS, D. M. Evaluation: from precision, recall and f-measure to roc, informedness, markedness and correlation. *arXiv preprint arXiv:2010.16061* (2020).

[15] PROTOCOL, U. D. Rfc 768 j. postel isi 28 august 1980. *Isi* (1980).

[16] RADFORD, B. J., APOLONIO, L. M., TRIAS, A. J., AND SIMPSON, J. A. Network traffic anomaly detection using recurrent neural networks. *arXiv preprint arXiv:1803.10769* (2018).

[17] RING, M., WUNDERLICH, S., GRÜDL, D., LANDES, D., AND HOTHO, A. Creation of flow-based data sets for intrusion detection. *Journal of Information Warfare 16*, 4 (2017), 41–54.

[18] RING, M., WUNDERLICH, S., GRÜDL, D., LANDES, D., AND HOTHO, A. Flow-based benchmark data sets for intrusion detection. In *Proceedings of the 16th European Conference on Cyber Warfare and Security. ACPI* (2017), pp. 361–369.