SVM-RF: A Hybrid Machine Learning Model for Detection of Malicious Network Traffic and Files

Prashant Mathur, Arjun Choudhary, Chetanya Kunndra, Kapil Pareek, and Gaurav Choudhary

1 Introduction

In the past couple of years information technology has seen a massive boom, along with this tremendous growth, cyberspace has also seen its fair share of malware, which are responsible for disrupting regular IT work. The term 'malware', originates from the term 'malicious software' and can be used to describe any software that is designed to hinder any computer resources in any form. It can be a piece of simplistic software that changes the time of a running computer, without the user's knowledge, or can be as sophisticated as '*Stuxnet*', which was used to target Iran's nuclear enrichment program [1].

The number of malware infections has increased from 12.4 million infections in 2009 to a whopping 812.67 million infections in 2018 alone [2]. Even during the COVID-19 pandemic, when the whole world came to a standstill, cybercrime saw exponential growth [3] owing to complete lockdowns, work-from-home mandates, and an exhausted, confused, and untrained population. Cybercriminals took this as an opportunity to further their cause. With The COVID-19 pandemic as their backdrop cybercriminals inculcated the fear of the pandemic into their tactics. Cyberspace saw a surge of COVID-19-themed phishing campaigns, malware being delivered through COVID-19-themed applications [25], and a widespread rampant abuse of the fear instilled within the general public by the pandemic. Organizations that were working on battling the pandemic were also a lucrative target of cyber criminals as seen in

P. Mathur (✉) · A. Choudhary · C. Kunndra · K. Pareek
Sardar Patel University of Police Security and Criminal Justice, Jodhpur, India
e-mail: mtcs20pm@policeuniversity.ac.in

A. Choudhary
e-mail: a.choudhary@policeuniversity.ac.in

G. Choudhary
Technical University of Denmark, Lyngby, Denmark

B. K. Roy et al. (eds.), *Cryptology and Network Security with Machine Learning*,
Algorithms for Intelligent Systems, https://doi.org/10.1007/978-981-99-2229-1_3

the case of Dr. Reddy's Laboratories, whose data servers were attacked days after they were approved to conduct trials of Russian made COVID-19 vaccine [26].

As per Kaspersky, a cybersecurity giant, there were a total of 666,809,967 attempts to launch malicious software via online services in 2020 [27] and 687,861,449 attempts in 2021 [28], 2022 is anyone's best guess. Among all the malware classes, the most notorious is the 'Ransomware' category. In this attack vector, the attacker designs the malware in such a way that it encrypts the victim's files and demands a ransom from the victim to decrypt those files, cryptocurrencies such as Bitcoin, Ethereum, etc. are demanded from the victim as the ransom, owing to the use of cryptocurrencies as a ransom it becomes difficult to trace the culprits behind the attack, due to the anonymous natures of all cryptocurrency transactions. As per Cisco, the average ransom paid against a ransomware attack in the year 2020 was nearly $312,493, with $10 million being the highest ransom paid, by 2031, ransomware are estimated to cost $250 billion annually, with the likelihood of a ransomware attack happening every two seconds [4].

Malware not only causes financial loss, but malware attacks can also leave the victim in a state of disarray, corporate victims can face service downtimes, critical data losses, and even loss of reputation, the list of setbacks caused by a malware attack goes on and on. Owing to the advancement in technology, malware authors have become more sophisticated, even with expensive sophisticated defenses in place, they can infect victims with relative ease. In all retrospect, malware have become more of a nuisance than a threat. Thus there arises a dire requirement to create an advanced intelligent system that is capable of identifying and stopping malware attacks before they are executed. This can be done using varying methods, such as scanning network traffic, scanning files, and monitoring user activity on a system. The solution should be capable of such features and should be on active lookout for the same. In this research we have tried to work on these issues and tried to come up with an effective solution that can help with the process of malware identification.

The paper is divided into 7 sections, Sect. 2 discusses the related work carried out in the field of malware detection using machine learning. Section 3 provides a brief overview on malwares, their types and their interaction with the victim. It also provides a brief classification of malwares based on their activities. Section 4 elaborates on our proposed machine learning model that is used to identify malwares. Section 5 explains about the experiments conducted by us, while Sect. 6 discusses the evaluation metrics used to determine the usability of our model, Sect. 7 lays down the results of the experiments performed. Section 7 is followed by the conclusion.

2 Related Work

As technology evolved from its nascent stages, an increased number of issues associated with it also emerged. Malwares over the years has become one such predominant issue. With the increasing sophistication in malwares, traditional detection mechanisms are not able to effectively detect malwares. In order to overcome this hurdle,

machine learning comes into picture. There have been numerous researches in the field using various machine learning and deep learning techniques. This section discusses some of the recent work done in the fields of malware detection using machine learning.

Liu et al. [5] proposes a machine learning model that is composed of three components that perform data processing, making decisions and malware detection. Their first module "data processing" is responsible for extracting features from the inputs. The second layer is used to detect suspicious nature of the malware, finally the third layer uses Shared nearest neighbor (SNN) to categorize input into malware families. Their proposed model gives an accuracy of 86.7% for new malware samples. Their model is trained, validated and tested on their own dataset collected in their home computer lab using Anubis, Kingsoft and ESET NOD32.

Rodrigo et al. [6] proposes BrainSheild, a hybrid machine learning model that employs a three neural network architecture packed with Relu activation function, ADAM optimizer to detect malwares in the android environment. The first neural network is used to for static analysis on the input and has an accuracy of 92.9%, their second neural network is used to carry out dynamic analysis of the input and shows an accuracy of 81.1%, their last neural network running their proposed model gives an accuracy of 91.1%. They use Omnidroid dataset to train, test and validate their proposed model.

Similar to [6], Kim et al. [24] proposes a deep learning based model for detection of malwares in the android environment, they use CNN to extract common features from the API call graph of the application and then use a lightweight classifier, Jaccard similarity algorithm, to classify the application based on similar characteristics. Their proposed model is trained, tested and validated on android applications downloaded from Google Play store and VirusShare and has an accuracy of 91.27%.

Hardy et al. [7] proposes a Stacked AutoEncoder (SAE) based deep learning model. The proposed model has two phases: "unsupervised pre-training" and "supervised backpropagation". Their model is trained, validated and tested using a dataset obtained from Comodo Cloud Security Center and has an accuracy of 95.64%.

Kan et al. [8] proposes a light-weight deep CNN model that detects malwares based on their grouped instructions. Their model takes raw inputs and groups the input based on instruction sets, the CNN model is used to classify the input as malicious. Their model is trained, validated and tested against a private dataset of 70,000 samples and has an accuracy of 95%.

Table 1 provides a brief overview of the recent research work done on the topic of malware detection using machine learning.

3 Overview

Malware can be considered as any software that is designed to bring harm to the victim by performing malicious actions without the knowledge of the victim. There are various ways to classify malware, here we will classify malware based upon

Table 1 Comparative overview of papers on malware detection using machine learning

Paper	Dataset	Accuracy (%)
Liu et al. [5]	ESET NOD32, VX Heavens	86.7
Hardy et al. [7]	Private dataset	95.64
Kan et al. [8]	Private dataset	95
Rodrigo et al. [6]	Omnidroid dataset	91.1
Kim et al. [24]	Google Play store + VirusShare	91.27
Our model	CTU-13, UNSW-NB15, MMCC	95.92

two most common classification methods, namely, the classification based on the general characteristics and the classification based on the actions a particular malware performs on the victim.

Based on the general characteristics such as propagation type, and general functions malware can be classified into the following categories—

- **Virus—**A virus is a malicious software program that attaches itself to a safe to execute file often by altering the code of the said file when the file is executed the malicious code is also executed, it then replicates to other files often infecting them in the same process [29]. To exist a computer virus must attach itself to a host file.
- **Worm—**A worm is also a replicating malware, but unlike a virus, it doesn't require a host program to propagate itself, it copies throughout the system and some even have the capabilities to propagate themselves over a network [30].
- **Trojan—**A Trojan malware takes its name from the infamous tale of 'The Trojan Horse' that was used by the Greeks during the Trojan War. This form of malware impersonates a legitimate and safe-to-use file tricking victims into executing it [31].
- **Rootkit—**A rootkit is a fairly advanced and stealthy malware, unlike other categories of malware, a rootkit is fairly hard to detect and remove from the system as it is designed to embed itself deep into the operating system, often employing legacy API calls to evade detection from antivirus software [32].
- **Keyloggers—**Keyloggers are predominantly used in Spyware, a keylogger is a piece of software that keeps track and logs all of the victims' keystrokes [33], based upon the intention of the author they can be considered as malicious or benign, a keylogger that is used to collect and exfiltrate all victim's PII is considered a malicious keylogger.
- **Backdoor—**This malware opens up an alternate communication channel between the victim and the attacker in a way that the attacker can bypass all authentication and security mechanisms put into place by the victim [34].
- **Mobile malware—**This umbrella term is used to categorize all malware that is present in the mobile device ecosystem, this category can include mobile ransomware, spyware, backdoors, or Trojans as well. Since the usage of mobile devices has increased exponentially since their conception, they too are a treasure

trove of PII for the attackers and hence a very lucrative target [35]. Owing to such significance, it is also the need of the hour to protect mobile devices from malware and malicious actors. 'AbstractEmu' is a fairly recent and extremely dangerous android malware that impersonates 19 types of safe-to-use applications and locks the victim out of their device due to its ability to gain root access to the device [36].

Based on the actions performed by the malware on a victim can be classified into the following categories—

- **Droppers—**This malware is designed to infect the victim with another piece of malware, primarily via covert methods. In other terms, droppers, download another piece of malware and infect the victim using that malware [16].
- **Launchers—**This malware is designed to covertly execute another malware [17].
- **Ransomware—**This malware encrypts user data using strong encryption techniques and then demands a ransom to decrypt, usually the ransom is asked to be paid via cryptocurrency, making the ransom untraceable. One of the prime examples of this category of malware is the infamous WannaCry ransomware [20].
- **Fearware—**This malware is used to instill fear in its victims, they do so by using varying methods, such as displaying threatening messages to victims or damaging their data. Ransomware can also be put in the umbrella category of fearwares. The COVID-19 pandemic saw an unprecedented increase in the use of fearwares to further cybercrimes [21].
- **Bots and Botnets—**This category of malware takes control of a device and executes commands from the attacker, unlike other compromised devices, a bot is part of a large network called the botnet and the said network is under the control of the attacker. An attacker uses a large network of bots to perform nefarious activities such as carrying out Distributed Denial of Service attacks (DDoS) as seen in the case of the infamous Mirai botnet.
- **Spyware—**These malwares are used to exfiltrate victims' PII(Personal Identifiable Information) and can even use them to steal their identity. Pegasus malware [22] is an iOS spyware that falls under this category.

There can be various other classifications of malware based on various factors such as the API calls a malware makes or whether the malware can mutate itself to avoid detection.

Traditional malware detection techniques rely on signature-based detection methods, such as pattern matching of unique strings specific to malware [18] to identify malware, some may even analyze function calls by performing static analytical operations on a file, but signature-based detection methods are unable to detect new strains of malware [19] and can be easily evaded. Threat actors hide their malicious code or obfuscate it to avoid being discovered by these techniques. The conventional method of detection is hampered by obscuration. Threat actors have come up with sophisticated and unique ways to deliver malware, they most commonly use emails or MS office documents to deliver malware either in the form of malicious links or in

the form of embedded macros. Once installed malware, based on its design can set up a remote connection to the attacker or can establish connections to a command and control server (C2 server), after installation, data can be exfiltrated or the malware can remain dormant until it receives some commands either from the attacker or a C2 server. Malware software updates can also be sent to the victim to install a newer version of the malware. SolarWinds supply chain attack is a prime example of how attackers attack and interact with victims, in the said attack, the supply chain of SolarWinds was compromised and attackers inserted a malicious code with the legitimate software update [23], it remained undetected for quite some time and attackers used the initial compromise to perform nefarious activities. All these interactions can be found in the network traffic. All these entities, namely, files being downloaded, the structure of the file, actions performed by malware, and network traffic among many others are key factors in determining an anomalous entity.

Manually looking out for all these threats is a very tedious and exhilarating task. Artificial intelligence can ease and speed up this process of threat detection, with robust machine learning or deep learning algorithms in place, erroneous detections can be minimized and can protect organizations and individuals to a much greater extent. The **Support Vector Machine** and the **Random Forest Classifier** algorithms are combined in our proposed hybrid model, which is capable of quickly identifying harmful files and malicious network traffic. It is trained, validated, and tested using three datasets, namely: **CTU-13** [9], **UNSW-NB15** [10–14], and **Microsoft Malware Classification Challenge** (MMCC) [15] datasets. The model created using the **MMCC** dataset is used to categorize harmful files, whereas the model created using the **CTU-13** and **UNSW-NB15** datasets are used to detect malicious network traffic (Fig. 1). For ease of understanding, we have classified the problem statements this research intends to solve into the following categories—

- **P1—**Problem of detection of malicious network traffic.
- **P2—**Problem of detection of malicious files.

4 Proposed Model

Support Vector Machine (SVM) is a linear model that can be used to tackle the problems of classification and regression. It can solve both linear and nonlinear problems and due to its adaptability, it can be used to solve a variety of problems. SVM works on a very basic concept: By drawing a line or hyperplane through the dataset, the method separates input into classes. The input data point is then plotted on the hyperplane and in whichever sector the data point lies, it belongs to that class.

Random forest is a supervised learning approach. It is a popular machine-learning algorithm. It can also be used to tackle the problems of regression and classification. It is based on ensemble learning, which is a technique used to solve complex problems by combining several classifiers into a single more refined classifier, this process

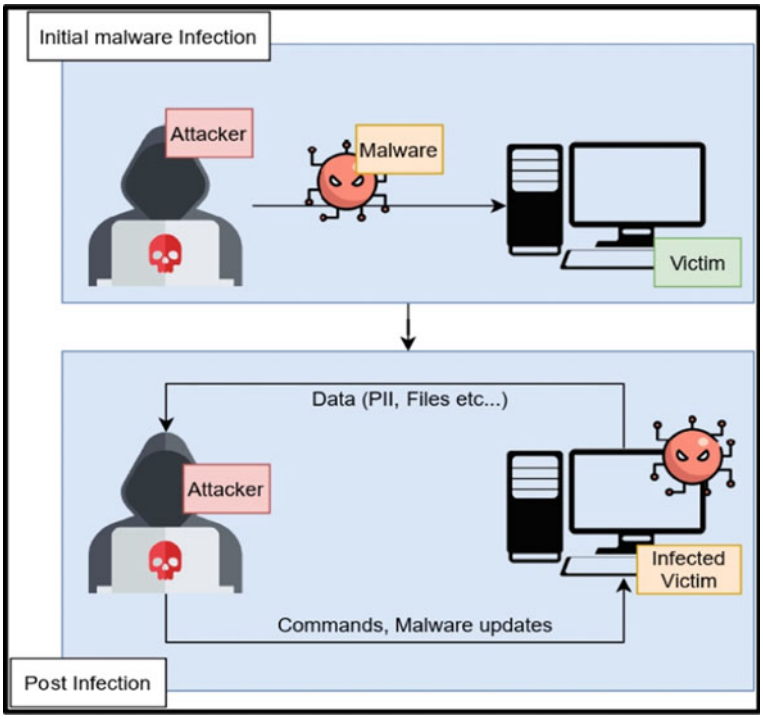

Fig. 1 A typical malware interaction with a victim

increases the overall performance of the combined classifier, making the learning process more efficient.

Our hybrid model uses SVM and RF algorithms and can be used to solve both problems, P1 and P2, SVM takes the raw data as an input and classifies the input based on its features, this step helps us in filtration and pooling our input, the output of the SVM layer is fed into the Random Forest classifier layer. RF enhances the classification done by SVM layers by fine-tuning the output of SVM layers, making the classification more precise and accurate. Figure 2 depicts the logic flow of our proposed model.

5 Experiments

To compute our SVM-RF model's effectiveness we compared its performance to KNN, SVM, RF, CART, CNN, and DF models, using the same datasets and in the same computational environment. All of the tests were performed on a 2.6 GHz Intel i7-8850H CPU with 16 GB RAM, 1 TB of hard disk space, and Ubuntu 20.04 LTS operating system. As a result of our experiments, we found out that our proposed

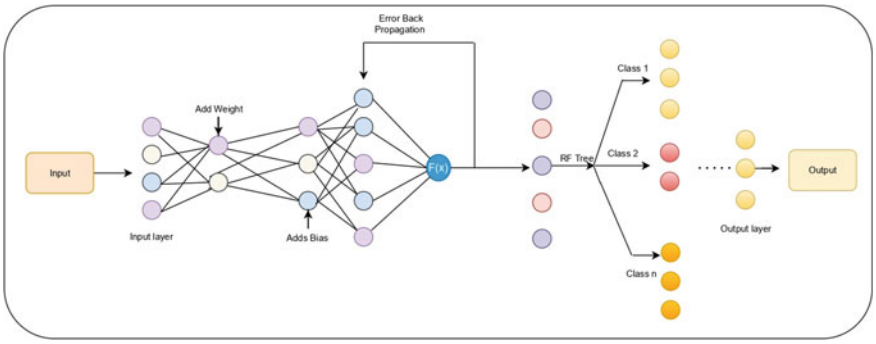

Fig. 2 Proposed SVM + RF model

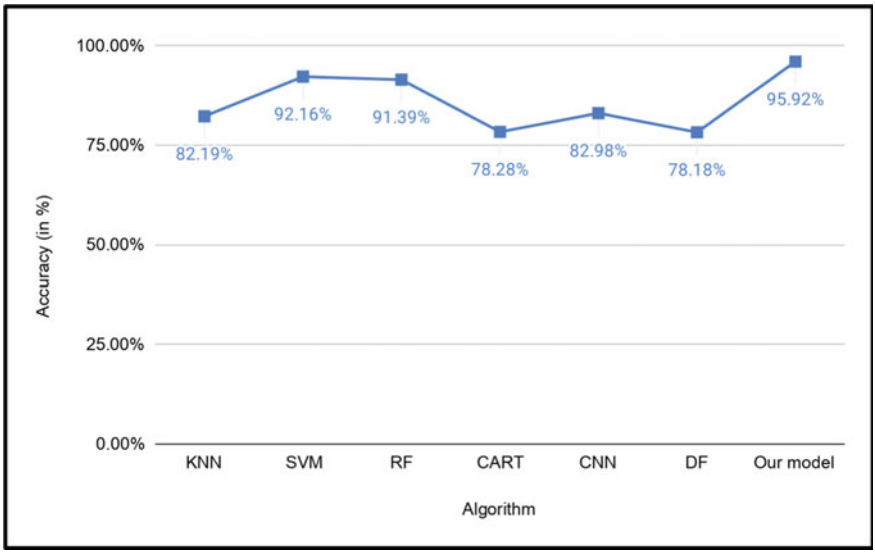

Fig. 3 Accuracy of various algorithms for problem P1

hybrid model showed the best results and accuracy for malicious network traffic and file detection as compared to other models. Figures 3 and 4 show a comparative result of our experimental runs.

6 Evaluation Metrics

Before evaluating the proposed model, we need to understand the following terms—

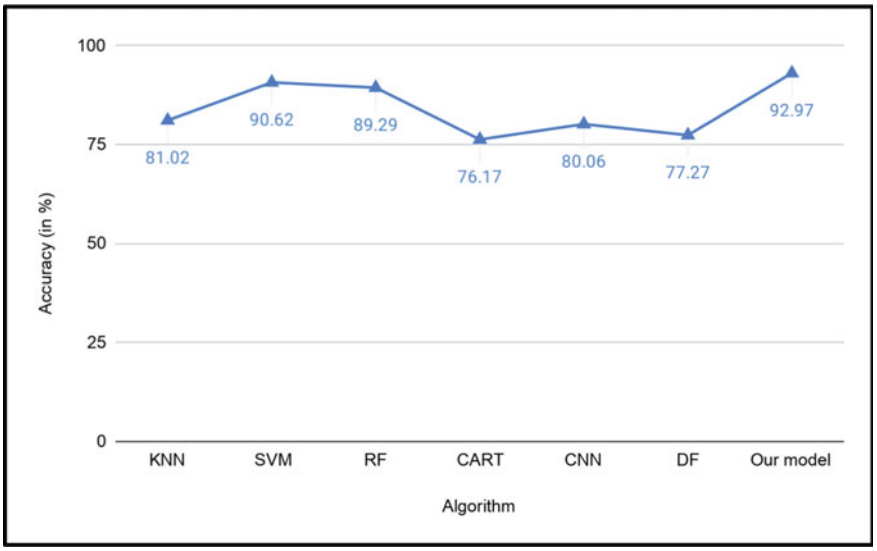

Fig. 4 Accuracy of various algorithms for problem P2

- **True Positives—**True positives (TP) refers to the prediction that is predicted to be true and is also true.
- **True Negatives—**True negatives (TF) refers to the prediction that is predicted to be false and is also false.
- **False Positives—**False positives (FP) refer to the prediction that is predicted to be true but is false.
- **False Negatives—**False negatives (FN) refers to the prediction that is predicted to be false but is true.

Keeping the above terms in mind, we use the following evaluation metrics to test the effectiveness of our proposed algorithm and to compare it with other algorithms—

- **Accuracy—**The ratio of the entire number of accurate forecasts—including both genuine positives and negatives—to the total number of predictions is known as accuracy. The formula provides a definition—

$$Acc = (TP + TN) / (TP + \backslash TN + FP + FN) \tag{1}$$

- **Error Rate—**The ratio of all inaccurate predictions—including false positives and negatives—to all forecasts is known as the error rate. The formula provides a definition—

$$Err = (FP + FN) / (TP + TN + FP + FN) \tag{2}$$

7 Results

Tables 2 and 3 depict the accuracy of various models on problems P1 and P2. In both the scenarios our proposed model performs better than other models with an accuracy of 95.92% while detecting malicious network traffic and an accuracy of 92.97% for detecting malicious files, under the current experimental conditions, our model is better suited for detecting malicious network traffic.

Table 2 Experimental results of various algorithms for problem P1

Dataset	Algorithm	Accuracy (%)	Error rate
CTU-13, UNSW-NB15	KNN	82.19	3.97
	SVM	92.16	2.28
	RF	91.39	2.96
	CART	78.28	3.48
	CNN	82.98	3.81
	DF	78.18	4.82
	SVM + RF	95.92	1.62

Table 3 Experimental results of various algorithms for problem P2

Dataset	Algorithm	Accuracy (%)	Error rate
MMCC	KNN	81.02	4.61
	SVM	90.62	2.89
	RF	89.29	3.01
	CART	76.17	4.89
	CNN	80.06	3.92
	DF	77.27	4.88
	SVM + RF	92.97	1.95

8 Conclusion

Detection of malwares is a challenging technological problem. This study presents a hybrid machine learning approach that can be used to detect malicious files and malicious network traffic. This model can help organizations and individuals in making their technology infrastructure more secure and reliable. As per the accuracy of the results we found that model is better suited for detection of malicious network traffic as our experiments show that the model gives an accuracy of 95.92% while detecting malicious network traffic and an accuracy of 92.97% while detecting malicious files. This model can be used to create a hybrid all-in-one security solution that can protect against malicious files and detect malicious network traffic, thus making an organization and even cyberspace a more secure space.

References

1. Michael Holloway (2015) URL: https://large.stanford.edu/courses/2015/ph241/holloway1/
2. 2021 Cyber Security Statistics Trends & Data (2021) URL: https://purplesec.us/resources/cyber-security-statistics/
3. Harjinder Singh Lallie et al. (2021) "Cyber security in the age of covid-19: A timeline and analysis of cyber-crime and cyber-attacks during the pandemic". Comput & Secur 105: 102248
4. Rachel Ackerly (2021) The cost of ransomware attacks: Why and how you should protect your data. URL: https://umbrella.cisco.com/blog/cost-of-ransomwareattacks
5. Liu et al. (2017) "Automatic malware classification and new malware detection using machine learning". Front Inf Technol & Electron Eng 18(9): 1336–1347
6. Rodrigo, Corentin et al. (2021) "BrainShield: A hybrid machine learning-based malware detection model for android devices. Electronics 10(23): 2948
7. William Hardy et al. (2016) "DL4MD: A deep learning framework for intelligent malware detection". In: Proceedings of the international conference on data science (ICDATA). The Steering Committee of The World Congress in Computer Science, Computer, p 61
8. Kan Z, "Towards light-weight deep learning based malware detection". In, et al (2018) IEEE 42nd annual computer software and applications conference (COMPSAC). vol. 1. IEEE 2018:600–609
9. The CTU-13 Dataset. A labeled dataset with botnet, normal and background traffic. URL: https://www.stratosphereips.org/datasets-ctu13
10. Nour Moustafa, Gideon Creech, Jill Slay (2017) "Big data analytics for intrusion detection system: Statistical decision-making using finite dirichlet mixture models". Data analytics and decision support for cybersecurity. Springer, pp 127–156
11. Nour Moustafa, Jill Slay (2016) "The evaluation of network anomaly detection systems: Statistical analysis of the UNSW-NB15 data set and the comparison with the KDD99 data set". Inf Secur J: Glob Perspect 25(1–3): 18–31
12. Nour Moustafa, Jill Slay (2015) "UNSW-NB15: A comprehensive data set for network intrusion detection systems (UNSW-NB15 network data set)". In: 2015 Military communications and information systems conference (MilCIS). IEEE. pp 1–6
13. Nour Moustafa, Jill Slay, Gideon Creech (2017) "Novel geometric area analysis technique for anomaly detection using trapezoidal area estimation on large-scale networks". IEEE Trans Big Data 5(4): 481–494
14. Mohanad Sarhan et al. (2020) "Netflow datasets for machine learning-based network intrusion detection systems". arXiv preprint arXiv:2011.09144

15. Microsoft Malware Classification Challenge (2015) URL: https://www.kaggle.com/c/malware-classification/rules
16. Kwon, Bum Jun, Jayanta Mondal, Jiyong Jang, Leyla Bilge, Tudor Dumitras (2015) "The dropper effect: Insights into malware distribution with downloader graph analytics." In: Proceedings of the 22nd ACM SIGSAC conference on computer and communications security, pp 1118–1129
17. Sikorski, Michael, Andrew Honig (2012) Practical malware analysis: The hands-on guide to dissecting malicious software. no starch press
18. Venugopal D, Guoning H (2008) Efficient signature based malware detection on mobile devices. Mob Inf Syst 4(1):33–49
19. Bazrafshan, Zahra, Hashem Hashemi, Seyed Mehdi Hazrati Fard, Ali Hamzeh (2013) "A survey on heuristic malware detection techniques." In: The 5th conference on information and knowledge technology, pp 113–120. IEEE
20. Chen, Qian, Robert A (2017) Bridges. "Automated behavioral analysis of malware: A case study of wannacry ransomware." In: 2017 16th IEEE international conference on machine learning and applications (ICMLA), pp 454–460. IEEE
21. Tripathi, Rahul (2017) "'Fearware' in the times of covid-19 pandemic." The Economic Times. The Economic Times, https://economictimes.indiatimes.com/tech/internet/fearware-in-the-times-of-covid-19-pandemic/articleshow/75664689.cms?from=mdr
22. Agrawal, Mayank, Gagan Varshney, Kaushal Pratap Singh Saumya, Manish Verma "Pegasus: Zero-click spyware attack—its countermeasures and challenges"
23. Wolff, Evan D, Growley KM, Gruden MG (2021) "Navigating the solarwinds supply chain attack." Procure Lawyer 56(2)
24. Kim, Jinsung et al. (2022) "MAPAS: a practical deep learning-based android malware detection system." Int J Inf Secur: 1–14
25. Naidoo R (2020) A multi-level influence model of COVID-19 themed cybercrime. Eur J Inf Syst 29(3):306–321
26. Bharadwaj, Swati (2020) "Hyderabad: Cyber Hit on Dr Reddy's labs as covid vaccine work begins: Hyderabad news—Times of India." The Times of India, The Times of India, https://timesofindia.indiatimes.com/city/hyderabad/cyber-hit-on-dr-reddys-labs-as-covid-vaccine-work-begins/articleshow/78818872.cms
27. Kaspersky (2020) "Kaspersky security bulletin 2020. Statistics: 26
28. Kaspersky (2021) "Kaspersky security bulletin 2021. Statistics: 26
29. David M Chess, Steve R White (2000) "An undetectable computer virus." Proc Virus Bull Conf 5
30. Kerr PK, Rollins J, Theohary CA (2010) The stuxnet computer worm: Harbinger of an emerging warfare capability. Congressional Research Service, Washington, DC
31. Etaher, Najla, George RS Weir, Mamoun Alazab (2015) "From zeus to zitmo: Trends in banking malware." 2015 IEEE Trustcom/BigDataSE/ISPA. vol. 1. IEEE
32. Embleton S, Sparks S, Zou CC (2013) SMM rootkit: A new breed of OS independent malware. Secur Commun Netw 6(12):1590–1605
33. Ladakis, Evangelos, et al. (2013) "You can type, but you can't hide: A stealthy GPU-based keylogger." In: Proceedings of the 6th European workshop on system security (EuroSec)
34. Zhang, Yin, Vern Paxson (2000) "Detecting backdoors." USENIX Secur Symp
35. Felt, Adrienne Porter, et al. (2011) "A survey of mobile malware in the wild." In: Proceedings of the 1st ACM workshop on security and privacy in smartphones and mobile devices
36. Alerts, Cyware (2021) "ABSTRACTEMU—the rooting malware with a global spread: Cyware Hacker News." Cyware Labs, Cyware Labs, https://cyware.com/news/abstractemu-the-rooting-malware-with-a-global-spread-92f9b995