Received March 5, 2022, accepted March 27, 2022, date of publication April 4, 2022, date of current version April 14, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3164392

Covert Channel Detection: Machine Learning Approaches

MUAWIA A. ELSADIG AND AHMED GAFAR
Deanship of Scientific Research, Imam Abdulrahman Bin Faisal University (IAU), Dammam 34212, Saudi Arabia

Corresponding author: Muawia A. Elsadig (muawiasadig66@gmail.com)

This work was supported by the Deanship of Scientific Research, Imam Abdulrahman Bin Faisal University, under Project 2019342-DSR.

ABSTRACT The advanced development of computer networks and communication technologies has made covert communications easier to construct, faster, undetectable and more secure than ever. A covert channel is a path through which secret messages can be leaked by violating a system security policy. The detection of such dangerous, unwatchable, and hidden threats is still one of the most challenging aspects. This threat exploits methods that are not dedicated to communication purposes, meaning that traditional security measures fail to detect its existence. This review has introduced a brief introduction of covert channel definitions, types and developments, with a particular focus on detection techniques using machine learning (ML) approaches. It provides a thorough review of the most common covert channels and ML techniques that are used to counter them, as well as addressing their achievements and limitations. In addition, this paper introduces a comparative experimental study for some common ML approaches that are commonly used in this field. Accordingly, the performance of these classifiers was evaluated and reported. The paper concludes that our information is still at risk, nothing is said to be secured and more work on the detection of covert channels is required.

INDEX TERMS Classification algorithms, covert channel detection, machine learning, covert traffic, covert storage channel, cover timing channel, deep learning, network traffic, network covert channels, overt traffic.

I. INTRODUCTION

A covert channel is a way to initiate communication between two parties to covertly leak information. This communication violates the established security policies of an organization. This illegitimate communication was initially defined in 1973 by Lampson [1], [2], after which Grilling extended this concept to computer network platforms [3], [4], enabling the initiation of covert channels over computer networks. The advanced development of computer network techniques has presented a rich environment in which to establish many scenarios of covert channels that are complicated enough to be detected and therefore pose many challenges for those seeking to establish secure communication [5]–[8]. Network-covert channels have proven to be effective in supporting many malicious activities. The creation of covert channels is a popular and effective way of information hiding that provision insecurity concerns [9]. Moreover, with the emergence of covert channel tools and techniques, hackers and attackers are capable of avoiding detection by network security devices [10].

A covert channel is unlike traditional secret message transfer methods in which not only the transmission content is hidden, but the transfer path itself is also protected [11]. In particular, network-covert channels maintain two aspects to secure the transmission of secret messages. These aspects include the security of communication content and connections. Network-covert channels effectively improve the security of both aspects [12].

Covert channel techniques are being rapidly developed owing to the influence of advanced communication technology. Some factors that play major roles in developing covert channel techniques are summarized in [13]. These factors include the advanced developments in network and communication technologies, switching techniques and internal control protocol technology.

The authors in [14] highlighted that continued work to counter this type of ongoing threat is urgently needed. In addition, there is a lack of countermeasures that focus on multiple types of covert channels. Mostly, each covert channel countermeasure is dedicated to countering one type

The associate editor coordinating the review of this manuscript and approving it for publication was Tyson Brooks.

of covert channel, instead of dealing with multiple types. Although there have been many attempts to develop methods capable to counter many types of covert channels, however, these methods are either inefficient or cause considerable overheads.

Long-term research efforts to increase the awareness of developers and engineers regarding the risks caused by network-covert channels are required to find common ground and avoid duplicated efforts and overlapping solutions [9]. In the early design phases of protocols, services, etc., awareness helps to avoid the weaknesses that can be exploited by these attacks.

This research paper focuses on covert channels as security threats that breach our networks and data; however, some research papers have presented useful uses of covert channels [15]–[21], which is normal because many techniques have a double-edged sword effect [22].

This section provides a brief introduction to covert channel attacks and shows how this type of threat can be rapidly developed to cause real challenges that need to be considered. The next section provides a very short introduction to covert channel types with greater focus on the two main types of covert techniques, and Section III discusses the wide spread of covert channel techniques among new technologies including the Internet of Things (IoT), IPv6 protocol and VoLTE technologies. It reflects how these technologies and techniques are vulnerable to being exploited by covert channel attacks and provides a rich environment in which to establish different covert channel techniques that pose many security challenges. Section IV is the core of our work, providing a thorough review of covert channel detection using machine learning (ML) approaches. ML classification models have ensured their ability and efficiency in the field of information security, as well as their importance and benefits in the general area of computer science. This section discusses the achievements and limitations of these techniques in detail and is especially focused on recent research to provide state-of-the-art information in this area. This is followed by Section V, which introduces an experimental comparative study of eight classification models to demonstrate their performance in terms of accuracy and error. A dataset was developed by constructing a packet-length-based covert channel that exploits network packet length to convey secret messages. A thorough discussion is provided in Section VI and the paper is concluded in Section VII.

II. COVERT CHANNEL TYPES

A covert channel is a communication channel used to transmit information by exploiting system resources that are not designed to convey data [23]. Commonly, there are two types of covert channels: timing and storage. In timing channels, the covert message is modulated into the timing behavior of an entity on the sending side to be retrieved by the receiving side [24], whereas in storage channels, a sender writes a covert message directly or indirectly into storage objects to be read by the receiver side [25]. The timing channels can be further divided into two subtypes: active and passive [26]. Some researchers refer to the combination of storage and timing channels in one approach as a third type, known as a hyper-covert channel [27]. This type can pose significant challenges, making it difficult to detect [28].

Tian *et al.* [12] highlighted that the construction of covert channels, which divides them into timing and storage channels, does not involve covert channels that are constructed based on changing the transmission network architecture. Therefore, they proposed dividing the key technologies for network-covert channel construction into two levels or aspects: the transmission network and communication content. For more details on this classification, interested readers can refer to [12].

Moreover, the classification of covert channels based on their behaviors, techniques, similarities, patterns, protocols, etc. has recently received the attention of the research community to help develop countermeasures that are capable of targeting multiple covert channels instead of having a countermeasure for each covert channel technique. Studies on this trend are presented in [6], [7].

III. COVERT CHANNELS & NEW TECHNOLOGIES

This section highlights the widespread use of covert channel techniques among some new technologies such as Internet of Things (IoT), IPv6 protocol, and VoLTE technologies. We also discuss how these technologies represent enriched environments to promote the construction of many covert channel techniques that pose real challenges.

A. COVERT CHANNELS OVER IoT

IoT applications and associated new technologies have enriched the spread of covert channels. Many covert communication methods have been introduced that exploit IoT protocols, either in the form of storage or timing channels.

It has been highlighted that covert channel threats against security and privacy in the IoT have been recently recognized and have raised the attention of security professionals; however, research in this field has not been significantly explored [29]. Most Internet of Things (IoT) devices have network interfaces that expose them to the public. These devices are characteristic, with limited resources, such as batteries, memory, and processing power, and lack appropriate security measures. Therefore, they are vulnerable to exploitation by different types of attacks.

Cabaj *et al.* [29] stated that most of the published papers regarding covert channels in IoT utilized data-hiding techniques in some IoT protocols. For example, some storage covert channels exploit the extensible messaging and presence Protocol [30], one timing covert channel and two storage covert channels use the building automation and control networking protocol [31] and two timing covert channels and six storage covert channels exploit the constrained application protocol (CoAP) [32], an extended work which includes power consumption analysis of these covert channels of CoAP, which is given in [33]. Moreover, Smith [34] indicated that while CoAP is widely used in IoT, it has mostly been ignored in covert channel research. They pointed out that distributed covert channels are a new technology that requires research attention. It spreads a covert message over many hiding techniques which makes detection more difficult. Accordingly, the authors presented two covert channels, one of which exploited unverified fields of the CoAP protocol and another used domain-generating algorithms (DGAs) for the virtual distribution of hidden messages to create a timing-based distributed covert channel [34].

The authors in [35] demonstrated the possibility of hiding data in a cyber physical system, such as a smart building, by making slight modifications to its components (e.g., controllers, sensors, etc.) or by exploiting unused registers to store secret data.

Moreover, the study in [36] aimed to demonstrate the vulnerability of IoT environments to the covert timing channels over mobile networks. They investigated different types of covert timing channel construction approaches to examine their ability to build covert timing channels for the IoT. This study classifies five types of covert timing channel construction approaches for IoT over 4G/5G mobile networks. These five timing covert channels include a packet reordering-based covert channel, retransmission-based covert channel, rate switching-based covert channel, scheduling-based covert channel, and a packet loss-based covert channel.

A recent study aimed to discover whether the message queuing telemetry transport (MQTT) protocol is subject to exploitation by covert channel attacks, as MQTT has become a popular protocol in IoT applications. This is a lightweight and publish-subscribe protocol. Practically, the authors investigated MQTT version 5 and reported that the number of covert channels can exploit this protocol but are not feasible for previous versions, as these covert techniques are based on some features of MQTT version 5. This reflects the ongoing development of covert channels and their ability to be deployed, even with the advanced development of network techniques [37], especially the IoT, which has become a common platform of communication. Moreover, Vaccari *et al.* proposed a tunnelling system capable of encapsulating messages over MQTT by exploiting its features that allow cyberattacks to be executed. They indicated that this protocol is a good choice over other protocols [38].

This section demonstrates the spread of different covert channel techniques among IoT protocols to reflect the amount of work required to counter these developed threats by considering them during the design phases.

B. COVERT CHANNELS AND IPV6

Although the IPv6 security issues have been addressed and improved, some issues remain and require further investigation. These issues concern inherent design vulnerabilities of the IPv6 and its incomplete implementation in all operating systems. Moreover, the successful deployment of the IPsec protocol within this protocol does not provide any guarantee or additional security against hidden channel attacks [39]. Lucena *et al.* introduced and analyzed twenty-two different covert channels that exploit the IPv6 protocol [40]. Interested readers for more information in IPv6 covert channels can refer to [12], [40]–[44].

C. COVERT CHANNELS AND VOLTE

In covert timing channels, a secret (covert) message is modulated into the IPDs of normal traffic; however, this is not applicable for VoLTE because the inter-packet delays of VoLTE traffic are fixed, and thus it is not possible to be modulated. This motivated the authors in [45] to introduce a covert channel in VoLTE traffic by adjusting periods of silence, in which a covert message can be modulated by extending or postponing periods of silence. To decrease the packet loss impact, the authors employed the grey code to encode the covert message. The authors demonstrated the undetectability of their proposed covert channel using statistical tests. In terms of robustness, the covert channel outperforms other IPD-based covert channels, as indicated by [45].

By exploiting the real-time interactive feature of VoLTE, in which data packets are sent in both directions (receiver side and sender side), Zhang *et al.* [25] constructed a two-way covert channel to ensure the receipt of a secret message so that the sender receives feedback from the recipient. The constructed covert channel involves two channels: timing and storage. In the timing channel, a secret message is modulated into the number of Silence Insertion Descriptor (SID) packets during the silence periods, while the storage channel is used to send feedback to ensure the receipt of the secret message. It exploits the real-time transport control protocol (RTCP) to inject the feedback message. The authors discussed the robustness and undetectability of the proposed covert channels. Moreover, a video packet reordering covert channels over VoLTE supported by ML algorithms was developed in [46] to confirm the construction of reliable covert communication over complex networking constraints.

This example shows that even with a technology that is difficult to exploit by covert techniques, attackers can find ways to establish covert attacks.

The advanced development of covert channel methods is undoubtedly noticeable, so the process of developing effective countermeasures still requires more attention. The presence of new ideas for constructing network-covert channels, such as reversible network-covert channels, has prompted the development of new detection methods. Reversible network covert channels can restore overt data without leaving any proof of their appearance [47]. In addition, prevention mechanisms should be considered in the early phases of designing protocols and services.

IV. COVERT CHANNELS DETECTION

Covert channel techniques use network resources that are not designed for communication purposes (e.g., timing information and packet headers) to leak information; therefore, conventional security measures fail to detect their existence. In addition, the available detection methods are dedicated to discovering specific covert channels and cannot be extended to include more covert channels [48].

This section mainly focuses on detection approaches that are based on machine-learning classification models to investigate their achievements and limitations. This paper gives more attention to recent work over the last five years with more focus on the papers that have been published in high-impact journals and conferences, as well as those that are highly cited.

The importance of ML techniques in supporting the security and privacy of several applications is notable. ML can contribute effectively to fulfilling the current real-world requirements in the security field. However, attackers can evade ML approaches by committing adversarial attacks. Therefore, assessing ML approach vulnerabilities in the early phases of development to deal with such attacks is critical. Sagar *et al.* analyzed different types of adversarial attacks that target ML approaches and represent defense strategies against them [51].

The authors in [52] presented a literature review on ML and deep learning techniques in network security, with focus on recent research. Their study introduced the latest applications in the field of intrusion detection. They indicated that each detection approach has its advantages and disadvantages; however, the most effective approach has not yet been established. A dataset is important, as no ML or deep learning approach works without data; however, the creation of an intrusion detection dataset is not easy and can be time consuming. Existing datasets suffer from many problems, such as outdated content and unevenness [52]. Moreover, many researchers have created their own network covert channel datasets for research purposes; unfortunately, these datasets are not publicly available [53].

Shaukat *et al.* indicated that ML techniques are advanced methods for the detection of cybercrime. They play an important role in fighting cybersecurity attacks and threats, such as malware detection, spam detection, intrusion detection, fraud detection, and phishing detection. However, they addressed some of the limitations listed in Table 1. These are limitations of certain ML models that are frequently used in cybersecurity [49].

ML for covert channel detection has been widely studied [54]. Nafea *et al.* indicated that the SVM algorithm is the best approach for detecting covert data [55]. However, the success of ML approaches depends on the availability of the traffic samples that represent many types of covert channels, and not only traffic samples that represent specific types of covert channels. Having a separate solution for every type of covert channel is not practical, as it may cause more overhead in network performance and capacity. Therefore, more research is required to obtain a standard dataset to imitate many types of covert channels for an effective ML solution instead of a separate solution for each technique.

TABLE 1. Limitations of some ML models.

MACHINE LEARNING MODEL	[49] Limitations
SVM	Incapable to manage large or noisy datasets efficiently. Does not deliver direct probability estimation. SVM consumes huge amount of space and time [49]. To achieve better results, in case of using dynamic datasets, it is necessary to train the dataset on different time intervals [50]
ANN	Time consuming and computationally cost. The influence of independent variables is difficult to be estimated.
RF	The cost of computation is high and slow to generate prediction
NB	The assumption that all features are completely independent is unrealistic. When a category in the testing dataset doesn't appear in the training dataset, a zero probability is assigned.
DT	Time consuming, complex and expensive

In addition, ML algorithms will not be effective unless there are statistical variations between normal and covert traffic. In other words, if covert traffic imitates normal traffic behavior, such detection techniques fail.

Sagar *et al.* reported that ML has a significant role to play in many areas e.g., real-time decision making, the processing of huge data, etc.; however, attackers can exploit ML vulnerabilities to commit many adversarial attacks, such as when a malicious user minimizes false positive (FP) rates and increases false negative (FN) rates in a way that does not affect the total error rate. This provides some leverage for attackers to commit sophisticated attacks [51].

Based on the comparison of the common classifier approaches presented in [49], Figure 1 shows the accuracy achieved by these classifiers using the NSL-KDD dataset to work as an anomaly based intrusion detection method. The figure shows the performance of these classification models in terms of their achieved accuracy, all of which had a high accuracy of more than 95%. It is believed that decision tree DT classifiers outperformed the other models by reaching an accuracy rate of 99.64%, followed by DBN and NB. SVM and ANN took the second line, whereas Random Forest (RF) showed the lowest accuracy.

Caviglione indicates that the detection of network covert timing channels considers that some statistical indicators or performance metrics can be used to evaluate the regularity of the time-based evolution of network traffic flow. This is true when the deviation in the timing statistics of the traffic flow is too large. However, it is too difficult to spot such channels when the attacker modifies the encoding approach or protocol or injects an appropriate amount of noise [9].

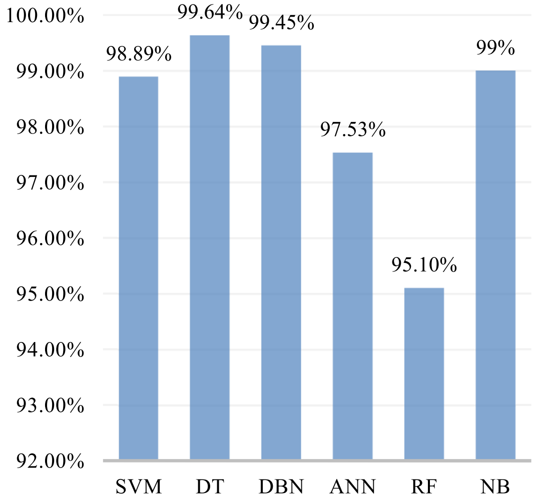

FIGURE 1. Classifiers accuracy comparison.

Qu *et al.* [16] stated that, in covert timing channels, when the threshold of packet delays to hide the covert message is equal to or less than a quarter of the mean of the interarrival times of overt traffic, distinguishing between covert and overt traffic will be difficult because of the overlaps in the time range of both overt and covert traffic. There is no overlap when the threshold is greater than or equal to the double mean interarrival time of overt traffic, and it is easy to distinguish between overt and covert traffic [16]. This reflects the difficulty in predicting covert timing channels with a threshold of packet delays that is equal to or less than a quarter of the mean of the interarrival times of overt traffic, which indicates more challenges in obtaining adequate detection methods for such scenarios.

To provide a clearer and more readable picture regarding the use of ML techniques to counter covert channel attacks by focusing on their cons and pros, Table 2 provides a thorough and in-depth review of recent ML methods and approaches to discover covert channels. Table 2 reflects the spread of ML techniques to predict the existence of covert channel attacks and focuses on their achievements and limitations. It can be noticed that most of the recent work has focused on DNS covert channels and most datasets were collected by the researchers themselves from real traffic of their investigated network. This addresses the challenge of having a common dataset that sufficiently reflects the normal behavior of DNS traffic, considering all network types. Interested readers can find more information on DNS insecurity [56] and DNS tunnel detection [57].

V. A COMPARATIVE STUDY AMONG EIGHT ML DETECTION METHODS

A. METHOD

This section presents a comparative scenario that includes the most commonly used classification methods in the areas of information security and covert channels. Eight classification models were used in this study. One of these is an ensemble classification model based on a stacking technique that takes the outputs of the other classifiers with the expectation of improving classification accuracy. A packet-length covert channel was selected for investigation by this work. This type of covert channel exploits the variation in the network packet lengths to modulate a covert message, i.e., odd length refers to 1 and even refers to 0 or vice versa. When an attacker wants to send a message, he or she modifies the network packet lengths according to the message, and the receiver watches the packet lengths to retrieve the encoded message.

Because of the lack of a public dataset for this type of covert channel technique and the fact that most research depends on a self-made dataset that considers a specific situation, a dataset of 180 instances has been developed. These instances included 90 instances of overt traffic and 90 instances of covert traffic. Wireshark, Python, and Scapy were used to construct the aforementioned dataset.

The test mining tools offered by the Orange software were used for dataset pre-processing. Orange is an open-source machine learning and data visualization tool which is a powerful platform for data analysis and equipped with diverse toolbox. Some feature selection methods have been applied to improve the classification accuracy. Feature selection is an important process for considering only the features that have a strong influence on the classification results and ignoring other features. This process improves both classification accuracy and performance in terms of computation overhead. The classification models were trained and tested using two different training-testing sets: 70%–30% and 90%–10%. For each experiment, a random validation method was used to repeat the training and testing phases to ensure valid and reliable results.

B. IMPLEMENTATION AND RESULTS

The eight classification models which include Stack (multi-classifier approach), neural network (NN), naïve bayes (NB), logistic regression (LR), random forest (RF), SVM, decision tree (DT), and KNN were trained and tested using our enhanced dataset described in the previous section. Random validation techniques were repeated 20 times for each experiment to obtain reliable and valid results. Each classifier was trained and tested using two different sets of training-testing. The classification accuracy, recall and precision of each model were computed using Equation 1, Equation 2, and Equation 3 respectively. In addition, the confusion matrix (error matrix) was computed for each classification model, which indicates the classification performance in terms of the FP and FN classification errors.

$$\mathit{Classification\ Accuracy} = \frac{(\mathrm{TP}+\mathrm{TN})}{(\mathrm{TP}+\mathrm{TN}+\mathrm{FP}+\mathrm{FN})} \quad (1)$$

$$\mathit{Recall} = \frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FN})} \quad (2)$$

$$\mathit{Precision} = \frac{\mathrm{TP}}{(\mathrm{TP}+\mathrm{FP})} \quad (3)$$

TABLE 2. Machine learning approach description, achievements and limitations.

Author	Machine Learning Approach Description, Achievements and Limitations	Covert Channel Type	Year
Shrestha et al. [48]	In this study, four statistical measures were derived from test traffic to train their classification model which was used to predict covert traffic. These statistical measures include the regularity score, K-S score, corrected conditional entropy and entropy. Their classification model which is based on SVM focuses on the four types of timing covert channels presented in [58-61]. Their proposed model was classified the investigated covert channels fairly and accurately as the authors indicated [48]. However, the false-negative rate (FN) is high [62]. In addition, extensive computations and prior knowledge of network IPD metrics are required to treat the classifier [63].	Four types of CTCs that use different techniques: On-Off, Jitterbug, Time Reply and L-bits.	2015
Salih et al. [44]	To detect covert storage channels in the IPv6 protocol, this paper proposes a detection technique based on the NB classifier and applies the hyper method for feature selection that uses C4.5 decision trees with the information gain technique. The paper has shown that IPv6 is vulnerable to covert channel attacks and, accordingly, has presented the aforementioned machine-learning approach to detect such attacks. Their results indicate that the proposed detection approach achieved high accuracy with a low false positive error rate compared to other classification models that were investigated in this study. However, the proposed model was limited to discovering covert channels that exploit ICMPv6 packets [64], and unqualified features were included in their study [65]. Moreover, although this is a promising result, it is questionable whether these attacks should be detected by an intrusion detection system because misuse of IPv6 header fields may already be prohibited by a packet filter [66].	Storage covert channel in IPV6	2015
Rezaei et al. [67]	The authors stated two major issues in the current detection methods used to detect covert timing channels. These issues are: (1) the current methods have a narrow focus and can only detect one or two covert techniques with inaccurate performance for other similar covert techniques; (2) they are computationally expensive and require extensive knowledge on the behaviors and characteristics of the previous network traffic. Therefore, the authors propose an approach for detecting several covert timing channels. Its aim is to detect covert channels that are based on inter-packet delay (IPD) distributions of network traffic. They presented three different nonparametric statistical tests to generate distinct statistical test scores for IPD traffic (normal and covert). This approach detects various CTCs that have similar impact on IPD distributions, with minimal lag between the point of detection and the start of the covert channel activity. The authors stated that their approach precisely segregated between normal and covert traffic. However, this approach results in high false-positive error rate [63].	CTCs	2017
Li et al. [62]	In this study, a random forest (RF) classifier was used to predict covert traffic. RF is an ensemble classifier based on the bagging technique. Commonly, ensemble classifier approaches attain high accuracy compared with single classification models, especially in a complex network environment. Eight features were extracted to train the classification model and four covert channels were investigated. The authors performed a	Four types of CTCs	2017

TABLE 2. *(Continued.)* **Machine learning approach description, achievements and limitations.**

	comparison between SVM and RF classifiers and reported that RF outperformed the SVM. Therefore, they proposed a detection method based on the RF classifier. In addition, the authors stated that their proposed classifier could also detect unknown CTCs based on their conducted experimental work when they tested their model using three untrained CTCs. The results showed good performance, as they claimed. However, their model causes high false positive (FP) rate in which the number of normal traffic that is classified as covert traffic is high.		
Lglesias and Zseby [68]	This study aimed to analyze the possibility of detecting CTCs as anomalies based on their statistical properties using unsupervised machine learning approaches. Accordingly, they constructed a single dataset of overt and covert traffic data. The covert traffic is generated based on the seven CTC techniques presented in [58, 59, 69-73]. Subsequently, three unsupervised methods were investigated using the aforementioned dataset. The results showed that the detection of covert channels using unsupervised outlier methods is unsatisfactory. Therefore, the authors were motivated to investigate supervised machine-learning models and accordingly, a Bayesian network was selected for testing. This method is robust and operates with a small amount of training data. The results showed that supervised machine learning models were satisfactory for differentiating between covert and overt traffic. The authors observed some false positive (FP) errors, indicating that some normal traffic cases were classified incorrectly as covert traffic.	Seven techniques of CTCs	2017
Iglesias et al. [74]	Eight CTC techniques were implemented to conduct the experiments to train, check, and evaluate the proposed detection approach presented by this paper. These covert techniques include packet presence (CAB), differential/derivative (ZAN), fixed intervals (BER), jitterbug/modulus (SHA), Huffman coding (JIN), timestamp manipulation (GIF), one threshold (GAS) and packet bursts (LUO). They constructed covert traffic based on these channels and used the MAWI database for overt traffic. For classification, the basic decision tree (DT) algorithm was used to differentiate between covert and overt flows. The DT algorithm is selected because it is embedded with feature selection methods and can potentially discard redundant and irrelevant features. Moreover, this algorithm depends on recursive partitioning (not on more complex options such as random forest) [74]. The classifier showed high accuracy; however, it revealed some false positives and negatives. The aim of this work is to test and evaluate the framework (descriptive analytics of traffic DAT) dedicated to CTC detection. This framework was theoretically described in [75]. The authors stated that their work proved the theoretical foundation of the DAT framework. Vázquez et al. [76] pointed out that DAT represents traffic flows using a set of statistical measurements and estimations. However, in [74] and [68], the scope was reduced to CTC analysis and detection using the supervised algorithms in [74] and unsupervised algorithms in [68].	Eight CTCs	2017
Elsadig et al. [77]	In this paper, it was reported that the packet length covert channel - which exploits the variation of network packet lengths to pass a covert message - is difficult to be detect because it generates covert traffic that closely imitates normal traffic. In this sense, this study demonstrated the capability of machine learning techniques to detect	Packet length covert channel	2018

TABLE 2. ***(Continued.)*** **Machine learning approach description, achievements and limitations.**

	such types of covert channels. They presented a comparative study among five machine learning classification approaches using a developed dataset including normal and covert instances. The classification algorithms include SVM, Neural Network, NB, LR, and RF. Their results showed a remarkable accuracy rate for all classifiers. Neural Network topped them by attaining an accuracy rate that reached 98% with zero FN and then NB, while RF and SVM were at the bottom of this assessment.		
Xu et al. [78]	In this study, an SVM classifier is proposed to detect storage covert channels that exploit the LTE-A protocol. The authors highlighted the lack of research on LTE-A covert channel construction; therefore, there has been no research on detecting such types of covert channels. This motivated them to present their detection scheme. They used Wireshark to capture normal LTE-A communication to form overt traffic, while using the MATLAB Simulink module, they constructed covert traffic using ten headers' fields of LTE-A RLC to send covert messages, and then again, they used the Wireshark tool to capture covert traffic. The authors showed that their proposed scheme can accurately distinguish between covert and overt traffic. However, the overhead of running this scheme to monitor the life network must be examined.	Storage covert channel using LTE-A protocol	2018
Nadler et al. [79]	The authors pointed out that previous work has focused on a specific class of DNS data leakage, known as DNS tunneling, without paying attention to DNS exfiltration malware which is an important class of DNS data leakage. Therefore, the authors presented a detection method that is capable of detecting both classes; however, this method fails when an attacker exploits several domain names to commit an attack [80]. Moreover, the authors addressed certain limitations regarding the assumptions in which they were based on their detection method.	DNS covert channel	2019
Çavuşoğlu [81]	This thesis used four features to train a decision tree classifier to detect covert traffic. These features include kurtosis, skewness, variance and mean. The aim was to discover the presence of two types of CTCs, Jitterbug and fixed interval. Jitterbug works by delaying a network packet for a predefined time, whereas a fixed interval covert channel works by defining an interarrival time for each symbol of the covert message. They constructed covert traffic, and used the MAWI dataset for normal traffic to form their dataset. The decision tree algorithm is suitable for data with categorical target classes; however, this algorithm is prone to overfitting problems; therefore, careful use of this algorithm is required.	CTCs, Jitterbug and fixed Interval	2019
Zhang et al. [82]	This study indicates that in DNS covert channel detection methods, most features are usually taken at the time of data exfiltration. Therefore, to prevent data exfiltration, the authors proposed a detection method to detect malicious queries from a single DNS request, they did not base their work on the features of network traffic or the features extracted from DNS behavior to ensure that their detection method is capable of detecting DNS tunneling prior to data exfiltration. The detection method is built on three deep learning models and implemented in a real network environment. As indicated by the authors, the detection method achieved an accuracy of 99.90%. However, they do not pay much attention to the generalization capability [83]. In addition, the proposed method is unscalable to large enterprise networks [84].	DNS covert channel	2019

TABLE 2. ***(Continued.)*** **Machine learning approach description, achievements and limitations.**

Ayub et al. [53]	In this study, the dataset was created as follows: they modified the C program presented in [85] to embed a covert channel into the header fields of TCP and IP protocols, whereas for the DNS protocol, they utilized the DNS2TCP application presented in [86]. Therefore, the covert channels were constructed using different network layers. Covert channel in network layer (IP covert channel): The linear kernel SVM was found to be more effective than the logistic regression, while the Gaussian kernel SVM outperformed both by achieving a high detection accuracy rate of 99.78%. Covert channel in the transport layer (TCP covert channel): The results show that logistic regression is not suitable for detecting covert channels in this layer. An accuracy rate of 64.29%. was achieved. The linear-kernel SVM achieved an accuracy of 72.29% with zero FN. This ensures minimum false alarms; however, it is not suitable when a high detection rate is required. Owing to the nonlinear nature of the Gaussian kernel SVM, high accuracy rate of 99.15% was achieved. However, it introduces more FN errors compared to the linear kernel SVM. Covert channel in application layer (DNS protocol): There is an enhancement in the accuracy rate for the logistic regression classifier, which reaches 93.22% with an average precision value. However, it is still not appropriate as stand-alone detection method. The K-nearest neighbors (K-NN) was more accurate than the logistic regression classifier. It achieved an accuracy rate of 94.74%. The decision tree has proven its effectiveness in detecting DNS covert channels, as it was found to be very effective by achieving an accuracy rate that reached 94.96% with a false alarm rate less than that of the K-NN.	Storage covert channels: TCP, IP and DNS	2019
Yang et al. [87]	In this work, the characteristics of DNS covert traffic were analyzed and the features that support the distinguishing process between covert and overt traffic were extracted. An ensemble machine learning classification model based on the stacking technique was used for detection purposes. It combines three classifiers: SVM, KNN, and Random Forest. The detection accuracy of the proposed model is 99%. However, stacking techniques may cause more overhead in terms of network performance compared to single-classification approaches; therefore, the model efficiency, especially in large networks, needs to be examined.	DNS covert channel	2020
Han et al. [10]	In this study, a detection scheme is proposed to detect covert timing channels based on the K-NN algorithm. Features from a series of statistics relevant to payload lengths and time intervals were used to train their scheme. Their scheme accuracy rate reached 0.96 with AUC of 0.9737. Four machine learning classification algorithms were compared to select the best model. These classification algorithms include SVM, NB, KNN, and logistic regression (LR). KNN and LR are perform well, whereas NB and SVM lag behind. The authors highlighted the shortcomings of several existing detection methods that are dedicated to discovering specific types of the covert timing channels; therefore, they proposed a detection scheme to overcome these issues. Many types of CTCs were implemented to ensure the capability of their proposed model to counter them. However, the authors reported that, in the long run, hackers began to learn how to avoid the statistical analysis of CTCs. Therefore, the extracted features that are currently successful may fail in the future.	CTCs	2020

TABLE 2. *(Continued.)* Machine learning approach description, achievements and limitations.

	In other words, the detection scheme fails when an attacker knows how to evade the statistical analysis of the channel.		
Al-Eidi et al. [88]	This study was proposed a detection approach based on machine learning and image processing techniques. They converted the network packet interarrival times into two-dimensional colored images. Subsequently, features were extracted to train a classifier to distinguish between covert and overt traffic. Four classifiers were trained and tested using features obtained from a set of images generated by converting DNS covert traffic into a dataset of colored images. The classifiers included SVM, DT, NB and ANN. Their proposed model achieved a remarkable accuracy of 95.83%. The robustness of their model was evaluated by utilizing covert messages of different sizes. In addition, the author stated that they proposed a mechanism to pinpoint the traffic part that contains covert messages; therefore, the system alerts when detecting covert traffic. This allowed the employed covert mitigation technique to start its work to mitigate this threat. However, the overhead of this approach needs to be evaluated thoroughly to examine its impact on network performance and to ensure that the approach does not affect the quality of service (QoS). In other words, no significant justification is presented to ensure that the proposed solution will not diminish the network performance as the proposed solution consists of many stages; therefore, more work is required to examine its overheads considering different network scenarios.	CTC (Based on packet inter-arrival times)	2020
Sattolo [89]	In this thesis, a detection approach based on a logistic regression classifier is presented to predict covert storage traffic that embeds covert messages into the identification (ID) field of the internet protocol (IP) header. They constructed their dataset by taking samples of real network traffic and then they were embedding bits into the ID to generate covert traffic. Two bytes of covert traffic are sent per network packet. Four covert messages with different sizes (4, 16, 64,256 bytes) were tested and the results showed remarkable accuracy rates, even for a covert message with a small size. However, their detection system works only for a relatively simple covert channel.	Storage covert channel (ID of IP header)	2021
Chen et al. [83]	A long short-term memory (LSTM) model was proposed to detect DNS covert channels. The datasets used in this study were constructed using multiple DNS covert channel tools including Iodine Dnscat2, Dns2tcp, DNShell v1.7, Ozymandns, Cobaltstrike, DNSExfiltrator and DET. These tools were used to simulate DNS covert traffic, and Wireshark was used to capture traffic. The proposed model consists of one hidden layer. An accuracy rate of 99.38% was achieved. Compared to the CNN model, this approach showed better performance. However, it has been reported that some approaches for DNS tunneling detection, including this approach, are effective in detecting tunneling traffic from malware, but they use features that are easily obfuscated by advanced DNS tunneling techniques [80].	DNS covert channel	2021
Yang et al. [90]	An ensemble classification scheme based on the stacking technique is presented. This ensemble classifier combined three classifiers: SVM, RF and KNN. The authors stated that there is an improvement in terms of the area under the carve (AUC) which reached 0.999 compared to the methods presented in [79, 91-93]. However, the claim that this method can detect even unknown covert traffic requires more verification as the method's capability to detect two types of unknown traffic is not a satisfactory reason to support this claim.	DNS covert channel	2021

TABLE 3. Classifiers performance, 70% for training and 30% for testing.

Classifier	Performance key indicators		
	Training size: 70% Testing size: 30%		
	Recall	Precision	Accuracy
Stack	98%	98%	98%
Neural Network (NN)	97.9%	97.9%	97.9%
Naïve Bayes (NB)	97.8%	97.8%	97.8%
Logistic Regression (LR)	97.8%	97.9%	97.8%
Random Forest (RF)	94.2%	94.2%	94.2%
SVM	96.8%	96.9%	96.8%
Decision Tree (DT)	88.3%	88.4%	88.3%
KNN	67.3%	80.2%	67.3%

TABLE 4. Classifiers performance, 90% for training and 10% for testing.

Classifier	Performance key indicators		
	Recall	Precision	Accuracy
Stack	98.9%	98.9%	98.9%
Neural Network (NN)	98.9%	98.9%	98.9%
Naïve Bayes (NB)	98.9%	98.9%	98.9%
Logistic Regression (LR)	98.6%	98.6%	98.6%
Random Forest (RF)	96.4%	96.4%	96.4%
SVM	96.9%	97.1%	96.9%
Decision Tree (DT)	87.5%	87.6%	87.5%
KNN	68.6%	80.7%	68.6%

The experimental results show an outstanding performance of some classifiers and moderate performance of others. As expected, the multi-classifier approach achieved better performance in both scenarios when the training sample size was 70% and 90%; however, when using 90% as the training size, some single classification models reached the same accuracy rate as the multi-classifier model. These classifiers include NB, NN, and LR. In this case, the single classifier is preferable as multiclassification approaches cause higher computational costs and are more time-consuming compared to single classification approaches. Table 3 and Table 4 show the obtained results of all experiments based on two scenarios: the training size in the first scenario was 70%, whereas in the second scenario, the training size was 90% of the dataset. Table 3 shows the results of the first scenario, whereas Table 4 shows the results of the second scenario. The computed performance indicators were accuracy, recall and precision. Based on the obtained results, we classified these classifiers into four groups according to the aforementioned performance indicators: very good, good, moderate, and poor. Stack, NB, NN, and LR performed well, achieving very good accuracy rates above 97.5% followed by the second group, which involves RF and SVM. They achieved accuracies of 96.4% and 96.9%, respectively. This was followed by the DT classifier, which recorded a moderate accuracy of 88.3%, whereas the KNN classifier lagged behind with a poor accuracy rate of 68.6%. Figure 2 shows the accuracy rate achieved by each classifier over the two different training-test sets. It can be seen that the four classifiers achieved a considerable accuracy rate, while the stack classifier topped them as it achieved a high accuracy rate over the two scenarios of the training-test sets.

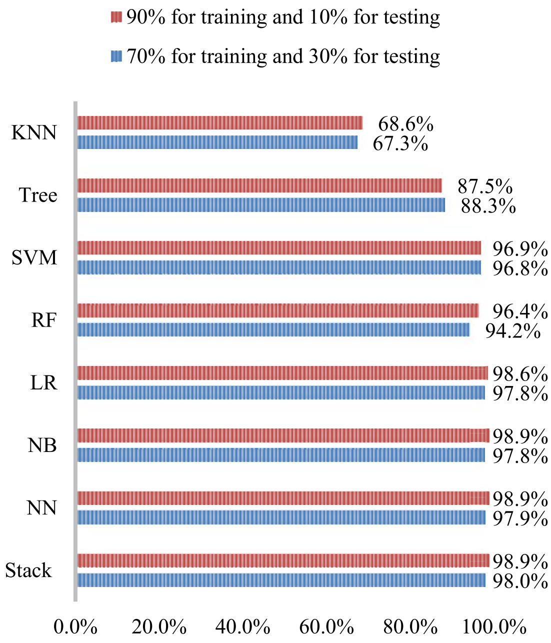

FIGURE 2. Classifiers accuracy.

To evaluate the classification errors, the confusion matrix was computed for all classifiers over the two training sets. The confusion matrix shows the FN and FP rates; FN indicates the number of covert instances that are classified incorrectly as overt instances, whereas FP indicates the number of overt cases that are classified incorrectly as covert instances. Table 5 lists the FP and FN values for all the classifiers over the two training sets. It can be seen that the stack classifier performed better by casing the least classification errors in terms of FN, with 0.017% FN and 0.026% FN for the training sets of 90% and 70%, respectively. It is noteworthy that NB, NN and LR also performed well by causing error rates that were closely related to those caused by the stack classifier. Next are SVM and RF, whereas DT and KNN caused a considerable amount of error.

The ROC curves of the experiments conducted throughout this work for eight classifiers over two sets of different

TABLE 5. Classification errors.

Classifier	Training size 70%		Training size 90%	
	FP	FN	FP	FN
Stack	0.015%	0.026%	0.006%	0.017%
NB	0.015%	0.029%	0.006%	0.017%
NN	0.015%	0.027%	0.006%	0.017%
LR	0.007%	0.036%	0.006%	0.022%
SVM	0.005%	0.058%	0.006%	0.056%
RF	0.054%	0.062%	0.072%	0.033%
DT	0.085%	0.087%	0.100%	0.150%
KNN	0.000%	0.656%	0.000%	0.628%

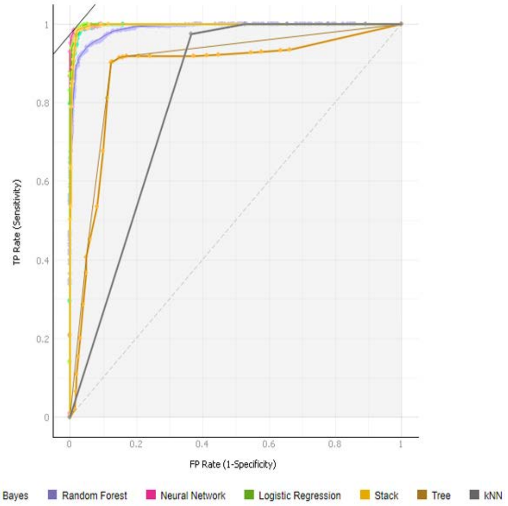

FIGURE 3. The ROC curves for all classifiers when using 70% of data for training and 30% for testing.

training-testing data were presented in Figure 3 and Figure 4. These curves reflect the performance of the investigated classifiers in graph form.

Figure 3 shows the ROC curves of all classifiers when using 70% of the dataset to train the classifiers and the rest of the dataset for testing, whereas Figure 4. shows the ROC curves for the same classifiers when using 90% of the dataset for training and 10% for testing.

The ROC curves show the performance of the eight classifiers and support the aforementioned findings and results.

VI. DISCUSSION AND RECOMMENDATIONS

Machine learning algorithms work when there is some variation between normal and covert traffic; therefore, any attempt from an adversary to imitate normal traffic, the ML algorithm will either fail to detect or its detection accuracy will be poor.

For a ML algorithm to be effective and monitor a network life traffic, it needs to be trained periodically to maintain its performance; otherwise, its efficiency will gradually decrease. In other words, the rapid development of both covert and overt traffic requires classification models to be updated and periodically retrained to be capable of countering these attacks. However, periodic retraining affects network performance and quality of service as it causes more overhead.

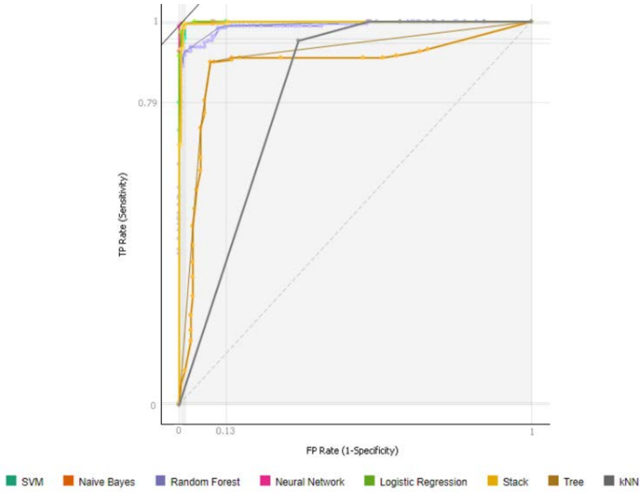

FIGURE 4. The ROC curves for all classifiers when using 90% of data for training and 30% for testing.

Ongoing competition between security professionals and attackers requires a self-trained approach that automatically updates itself.

For the datasets developed by the most covert channel detection approaches presented, an important question is raised on how researchers make sure that normal traffic, on which they are basing their work, is really overt. It may be that an undiscovered type of covert channel exists; therefore, researchers have to validate their findings using multiple traffic obtained from different networks under different situations to generate trusted normal traffic and then construct their covert traffic on this basis. Additionally, the creation of covert traffic requires validation.

Many researchers have generated their own network-covert channel datasets for experimentation purposes; however, they are not available for public use. Moreover, existing datasets suffer from many problems, such as outdated content and unevenness.

Having a separate solution for every type of covert channel is not a practical solution because it may cause more overhead to network performance and capacity; therefore, the quality of service (QoS) will be degraded. In most of the available detection methods, each detection method focuses on discovering specific types of covert channels and cannot be extended to involve more covert channels. Therefore, developing multi-detection approaches that are capable of detecting different types of covert channels is highly recommended. However, developing such approaches requires careful design to ensure a high detection accuracy rate with minimum overheads, as multi-detection approaches are subject to more overheads that may breach network performance and therefore QoS. However, this balance is a challenge. In addition, the lack of publicly validated datasets describing a covert channel or group of covert channels aims to assist research and base their proposed solutions on them.

To increase the knowledge and understanding of covert channel techniques, the authors encouraged similar efforts to the work presented in [94], which introduced a network security laboratory on data analysis to detect TCP/IP covert channels. This laboratory is for teaching purposes; therefore, similar work that covers different types of covert channels is highly recommended.

VII. CONCULSION

This paper investigates the efficiency of machine learning approaches to discover covert channel attacks. The paper provides a brief introduction to covert channel attacks, highlighting the widespread use of covert channel techniques among new technologies such as the Internet of Things (IoT), IPv6 protocol, and VoLTE technologies. This reflects how these technologies and techniques are vulnerable to being exploited by covert channel attacks and how they provide a rich environment in which to establish different covert channel attack techniques that pose many challenges. This review article has mainly contributed by examined the efficiency of machine learning techniques to counter covert channel attacks, with a deep focus on their pros and cons. In addition, it introduced a comparative study of eight ML classification approaches and the associated experimental results in terms of their performance and detection accuracy were reported.

The paper concluded that ML algorithms make a significant contribution to detect covert channel attacks and can effectively fulfil current real-world requirements in the security field; however, any attempt to imitate normal traffic, ML algorithms either fail to detect the existence of covert channels or their detection accuracy decreases. In addition, ML algorithms have many vulnerabilities that allow attackers to commit sophisticated attacks. Therefore, assessing ML approach vulnerabilities in the early phases of development to deal with such attacks is urgently needed.

It is difficult to have ML algorithms that work efficiently with multiple covert channels; if this happens, then this approach will certainly be computationally costly and lead to increased network overhead, thus diminishing the quality of service (QoS).

In the long term, if an attacker successfully avoids the statistical analysis of a covert channel, the extracted features of the employed detection model will fail. Therefore, it is noteworthy to mention that the research doors are widely open to more contributions in this area.

REFERENCES

[1] B. W. Lampson, "A note on the confinement problem," *Commun. ACM*, vol. 16, no. 10, pp. 613–615, 1973.

[2] D. Frolova, K. Kogos, and A. Epishkina, "Traffic normalization for covert channel protecting," in *Proc. IEEE Conf. Russian Young Researchers Electr. Electron. Eng. (ElConRus)*, Jan. 2021, pp. 2330–2333.

[3] L. Zhang, G. Liu, and Y. Dai, "Network packet length covert channel based on empirical distribution function," *J. Netw.*, vol. 9, no. 6, pp. 1440–1446, Jun. 2014.

[4] C. G. Girling, "Covert channels in LAN's," *IEEE Trans. Softw. Eng.*, vol. SE-13, no. 2, p. 292, Feb. 1987.

[5] M. Elsadig and Y. Fadlalla, "Survey on covert storage channel in computer network protocols: Detection and mitigation techniques," *Int. J. Adv. Comput. Netw. Secur.*, vol. 6, pp. 11–17, Dec. 2016.

[6] S. Wendzel, S. Zander, B. Fechner, and C. Herdin, "Pattern-based survey and categorization of network covert channel techniques," *ACM Comput. Surv.*, vol. 47, no. 3, p. 50, 2015.

[7] S. Wendzel, W. Mazurczyk, and S. Zander, "Unified description for network information hiding methods," *J. Universal Comput. Sci.*, vol. 22, no. 11, pp. 1456–1486, 2016.

[8] M. Wojciech, W. Steffen, Z. Sebastian, H. Amir, and S. Krzysztof, "Control protocols for reliable network steganography," in *Information Hiding in Communication Networks: Fundamentals, Mechanisms, Applications, and Countermeasures*. Hoboken, NJ, USA: Wiley, 2016, p. 296.

[9] L. Caviglione, "Trends and challenges in network covert channels countermeasures," *Appl. Sci.*, vol. 11, no. 4, p. 1641, Feb. 2021.

[10] J. Han, C. Huang, F. Shi, and J. Liu, "Covert timing channel detection method based on time interval and payload length analysis," *Comput. Secur.*, vol. 97, Oct. 2020, Art. no. 101952.

[11] L. Zhang, T. Huang, W. Rasheed, X. Hu, and C. Zhao, "An enlarging-the-capacity packet sorting covert channel," *IEEE Access*, vol. 7, pp. 145634–145640, 2019.

[12] J. Tian, G. Xiong, Z. Li, and G. Gou, "A survey of key technologies for constructing network covert channel," *Secur. Commun. Netw.*, vol. 2020, pp. 1–20, Aug. 2020.

[13] A. Epishkina and K. Kogos, "A traffic padding to limit packet size covert channels," in *Proc. 3rd Int. Conf. Future Internet Things Cloud*, Aug. 2015, pp. 519–525, doi: 10.1109/FiCloud.2015.20.

[14] M. A. Elsadig and Y. A. Fadlalla, "Survey on covert storage channel in computer network protocols detection and mitigation techniques," in *Proc. 4th Int. Conf. Adv. Inf. Process. Commun. Technol. (IPCT)*, Aug. 2016, pp. 79–85.

[15] R. deGraaf, J. Aycock, and M. J. Jacobson, "Improved port knocking with strong authentication," in *Proc. 21st Annu. Comput. Secur. Appl. Conf. (ACSAC)*, 2005, p. 10.

[16] H. Qu, Q. Cheng, and E. Yaprak, "Using covert channel to resist DoS attacks in WLAN," in *Proc. ICWN*, 2005, pp. 38–44.

[17] W. Mazurczyk and Z. Kotulski, "New security and control protocol for VoIP based on steganography and digital watermarking," in *Proc. 5th Int. Conf. Comput. Sci. Res. Appl. (IBIZA)*, Poland, Kazimierz Dolny, Feb. 2006.

[18] D. D. Dhobale, V. R. Ghorpade, B. S. Patil, and S. B. Patil, "Steganography by hiding data in TCP/IP headers," in *Proc. 3rd Int. Conf. Adv. Comput. Theory Eng. (ICACTE)*, Aug. 2010, pp. 20–22, doi: 10.1109/ICACTE.2010.5579643.

[19] H. Xie and J. Zhao, "A lightweight identity authentication method by exploiting network covert channel," *Peer-to-Peer Netw. Appl.*, vol. 8, no. 6, pp. 1038–1047, Nov. 2015.

[20] S. Vanderhallen, J. Van Bulck, F. Piessens, and J. T. Mühlberg, "Robust authentication for automotive control networks through covert channels," *Comput. Netw.*, vol. 193, Jul. 2021, Art. no. 108079.

[21] X. Ying, G. Bernieri, M. Conti, and R. Poovendran, "TACAN: Transmitter authentication through covert channels in controller area networks," in *Proc. 10th ACM/IEEE Int. Conf. Cyber-Physical Syst.*, Apr. 2019, pp. 23–34.

[22] Z. Wu, J. Guo, C. Zhang, and C. Li, "Steganography and steganalysis in voice over IP: A review," *Sensors*, vol. 21, no. 4, p. 1032, Feb. 2021.

[23] S. Al-Eidi, O. Darwish, and Y. Chen, "Covert timing channel analysis either as cyber attacks or confidential applications," *Sensors*, vol. 20, no. 8, p. 2417, 2020.

[24] X. Zhang, L. Zhu, X. Wang, C. Zhang, H. Zhu, and Y.-A. Tan, "A packet-reordering covert channel over VoLTE voice and video traffics," *J. Netw. Comput. Appl.*, vol. 126, pp. 29–38, Jan. 2019.

[25] X. Zhang, L. Guo, Y. Xue, and Q. Zhang, "A two-way VoLTE covert channel with feedback adaptive to mobile network environment," *IEEE Access*, vol. 7, pp. 122214–122223, 2019.

[26] S. Wu, Y. Chen, H. Tian, and C. Sun, "Detection of covert timing channel based on time series symbolization," *IEEE Open J. Commun. Soc.*, vol. 2, pp. 2372–2382, 2021.

[27] M. A. Elsadig and Y. A. Fadlalla, ''Network protocol covert channels: Countermeasures techniques,'' in *Proc. 9th IEEE-GCC Conf. Exhib. (GCCCE)*, May 2017, pp. 1–9.

[28] S. Z. Goher, B. Javed, and N. A. Saqib, ''Covert channel detection: A survey based analysis,'' in *High Capacity Opt. Netw. Emerging/Enabling Technol.*, Dec. 2012, pp. 057–065, doi: 10.1109/HONET.2012.6421435.

[29] K. Cabaj, P. Żórawski, P. Nowakowski, M. Purski, and W. Mazurczyk, ''Efficient distributed network covert channels for Internet of Things environments,'' *J. Cybersecurity*, vol. 6, no. 1, Jan. 2020.

[30] R. Patuck and J. Hernandez-Castro, ''Steganography using the extensible messaging and presence protocol (XMPP),'' 2013, *arXiv:1310.0524*.

[31] S. Wendzel, ''Covert and side channels in buildings and the prototype of a building-aware active warden,'' in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2012, pp. 6753–6758.

[32] A. Mileva, A. Velinov, and D. Stojanov, ''New covert channels in Internet of Things,'' in *Proc. 12th Int. Conf. Emerg. Secur. Inf., Syst. Technol. (SECURWARE)*, Venice, Italy, Sep. 2018, pp. 30–36.

[33] A. Velinov, A. Mileva, and D. Stojanov, ''Power consumption analysis of the new covert channels in coap,'' *Int. J. Adv. Secur.*, vol. 12, no. 1, pp. 42–52, 2019.

[34] S. Smith, ''Hiding in the noise: Creation and detection analysis of modern covert channels,'' Ph.D. dissertation, Dept. Comput. Sci., Tennessee Technol. Univ., Cookeville, TN, USA, 2020.

[35] S. Wendzel, W. Mazurczyk, and G. Haas, ''Don't you touch my nuts: Information hiding in cyber physical systems,'' in *Proc. IEEE Secur. Privacy Workshops (SPW)*, May 2017, pp. 29–34.

[36] Y.-A. Tan, X. Zhang, K. Sharif, C. Liang, Q. Zhang, and Y. Li, ''Covert timing channels for IoT over mobile networks,'' *IEEE Wireless Commun.*, vol. 25, no. 6, pp. 38–44, Dec. 2018.

[37] A. Mileva, A. Velinov, L. Hartmann, S. Wendzel, and W. Mazurczyk, ''Comprehensive analysis of MQTT 5.0 susceptibility to network covert channels,'' *Comput. Secur.*, vol. 104, May 2021, Art. no. 102207.

[38] I. Vaccari, S. Narteni, M. Aiello, M. Mongelli, and E. Cambiaso, ''Exploiting Internet of Things protocols for malicious data exfiltration activities,'' *IEEE Access*, vol. 9, pp. 104261–104280, 2021.

[39] A. Salih, X. Ma, and E. Peytchev, ''Implementation of hybrid artificial intelligence technique to detect covert channels attack in new generation internet protocol IPv6,'' in *Leadership, Innovation and Entrepreneurship as Driving Forces of the Global Economy*. Cham, Switzerland: Springer, 2017, pp. 173–190.

[40] N. B. Lucena, G. Lewandowski, and S. J. Chapin, ''Covert channels in IPv6,'' in *Proc. Int. Workshop Privacy Enhancing Technol.* Berlin, Germany: Springer, 2005, pp. 147–166.

[41] W. Mazurczyk, K. Powójski, and L. Caviglione, ''IPv6 covert channels in the wild,'' in *Proc. 3rd Central Eur. Cybersecurity Conf.*, Nov. 2019, pp. 1–6.

[42] L. Caviglione, M. Zuppelli, W. Mazurczyk, A. Schaffhauser, and M. Repetto, ''Code augmentation for detecting covert channels targeting the IPv6 flow label,'' in *Proc. IEEE 7th Int. Conf. Netw. Softwarization (NetSoft)*, Jun. 2021, pp. 450–456.

[43] P. Bedi and A. Dua, ''Network steganography using extension headers in IPv6,'' in *Proc. Int. Conf. Inf., Commun. Comput. Technol.* Singapore: Springer, 2020, pp. 98–110.

[44] A. Salih, X. Ma, and E. Peytchev, ''Detection and classification of covert channels in IPv6 using enhanced machine learning,'' in *Proc. Int. Conf. Comput. Technol. Inf. Syst. (ICCTIS)*, Dubai, UAE, 2015, pp. 1–7.

[45] X. Zhang, Y.-A. Tan, C. Liang, Y. Li, and J. Li, ''A covert channel over VoLTE via adjusting silence periods,'' *IEEE Access*, vol. 6, pp. 9292–9302, 2018.

[46] X. Zhang, L. Pang, L. Guo, and Y. Li, ''Building undetectable covert channels over mobile networks with machine learning,'' in *Proc. Int. Conf. Mach. Learn. Cyber Secur.* Cham, Switzerland: Springer, 2020, pp. 331–339.

[47] T. Schmidbauer and S. Wendzel, ''Hunting shadows: Towards packet runtime-based detection of computational intensive reversible covert channels,'' in *Proc. 16th Int. Conf. Availability, Rel. Secur.*, Aug. 2021, pp. 1–10.

[48] P. L. Shrestha, M. Hempel, F. Rezaei, and H. Sharif, ''A support vector machine-based framework for detection of covert timing channels,'' *IEEE Trans. Dependable Secure Comput.*, vol. 13, no. 2, pp. 274–283, Apr. 2015.

[49] K. Shaukat, S. Luo, V. Varadharajan, I. A. Hameed, S. Chen, D. Liu, and J. Li, ''Performance comparison and current challenges of using machine learning techniques in cybersecurity,'' *Energies*, vol. 13, no. 10, p. 2509, May 2020.

[50] S. S. Iyer and S. Rajagopal, ''Applications of machine learning in cyber security domain,'' in *Handbook of Research on Machine and Deep Learning Applications for Cyber Securityw*. Hershey, PA, USA: IGI Global, 2020, pp. 64–82.

[51] R. Sagar, R. Jhaveri, and C. Borrego, ''Applications in security and evasions in machine learning: A survey,'' *Electronics*, vol. 9, no. 1, p. 97, Jan. 2020.

[52] Y. Xin, L. Kong, Z. Liu, Y. Chen, Y. Li, H. Zhu, M. Gao, H. Hou, and C. Wang, ''Machine learning and deep learning methods for cybersecurity,'' *IEEE Access*, vol. 6, pp. 35365–35381, 2018.

[53] M. A. Ayub, S. Smith, and A. Siraj, ''A protocol independent approach in network covert channel detection,'' in *Proc. IEEE Int. Conf. Comput. Sci. Eng. (CSE) IEEE Int. Conf. Embedded Ubiquitous Comput. (EUC)*, Aug. 2019, pp. 165–170.

[54] A. Epishkina, M. Finoshin, K. Kogos, and A. Yazykova, ''Timing covert channels detection cases via machine learning,'' in *Proc. Eur. Intell. Secur. Informat. Conf. (EISIC)*, Nov. 2019, p. 139.

[55] H. Nafea, K. Kifayat, Q. Shi, K. N. Qureshi, and B. Askwith, ''Efficient non-linear covert channel detection in TCP data streams,'' *IEEE Access*, vol. 8, pp. 1680–1690, 2020.

[56] G. Schmid, ''Thirty years of DNS insecurity: Current issues and perspectives,'' *IEEE Commun. Surveys Tuts.*, vol. 23, no. 4, pp. 2429–2459, 2021.

[57] Y. Wang, A. Zhou, S. Liao, R. Zheng, R. Hu, and L. Zhang, ''A comprehensive survey on DNS tunnel detection,'' *Comput. Netw.*, vol. 197, Oct. 2021, Art. no. 108322.

[58] S. Cabuk, C. E. Brodley, and C. Shields, ''IP covert timing channels: Design and detection,'' in *Proc. 11th ACM Conf. Comput. Commun. Secur. (CCS)*, 2004, pp. 178–187.

[59] G. Shah, A. Molina, and M. Blaze, ''Keyboards and covert channels,'' in *Proc. USENIX Secur. Symp.*, vol. 15, 2006, p. 64.

[60] S. H. Sellke, C. C. Wang, S. Bagchi, and N. Shroff, ''TCP/IP timing channels: Theory to implementation,'' in *Proc. IEEE INFOCOM*, Apr. 2009, pp. 2204–2212.

[61] S. Cabuk, ''Network covert channels: Design, analysis, detection, and elimination,'' Ph.D. dissertation, Dept. Electron. Elect. Eng., Purdue Univ., West Lafayette, IN, USA, 2006.

[62] Q. Li, P. Zhang, Z. Chen, and G. Fu, ''Covert timing channel detection method based on random forest algorithm,'' in *Proc. IEEE 17th Int. Conf. Commun. Technol. (ICCT)*, Oct. 2017, pp. 165–171.

[63] S. Lu, Z. Chen, G. Fu, and Q. Li, ''A novel timing-based network covert channel detection method,'' *J. Phys., Conf.*, vol. 1325, no. 1, Oct. 2019, Art. no. 012050.

[64] M. Tayyab, B. Belaton, and M. Anbar, ''ICMPv6-based DoS and DDoS attacks detection using machine learning techniques, open challenges, and blockchain applicability: A review,'' *IEEE Access*, vol. 8, pp. 170529–170547, 2020.

[65] A. H. B. Alghuraibawi, R. Abdullah, S. Manickam, and Z. A. Alkareem Alyasseri, ''Detection of ICMPv6-based DDoS attacks using anomaly based intrusion detection system: A comprehensive review,'' *Int. J. Electr. Comput. Eng.*, vol. 11, no. 6, p. 5216, Dec. 2021.

[66] M. Schrötter, T. Scheffler, and B. Schnor, ''Evaluation of intrusion detection systems in IPv6 networks,'' in *Proc. 16th Int. Joint Conf. e-Bus. Telecommun.*, 2019, pp. 408–416.

[67] F. Rezaei, M. Hempel, and H. Sharif, ''Towards a reliable detection of covert timing channels over real-time network traffic,'' *IEEE Trans. Depend. Secure Comput.*, vol. 14, no. 3, pp. 249–264, May 2017.

[68] F. Iglesias and T. Zseby, ''Are network covert timing channels statistical anomalies?'' in *Proc. 12th Int. Conf. Availability, Rel. Secur.*, Aug. 2017, pp. 1–9.

[69] S. Zander, G. Armitage, and P. Branch, ''An empirical evaluation of IP time to live covert channels,'' in *Proc. 15th IEEE Int. Conf. Netw.*, Nov. 2007, pp. 42–47, doi: 10.1109/ICON.2007.4444059.

[70] V. Berk, A. Giani, and G. Cybenko, ''Detection of covert channel encoding in network packet delays,'' Dept. Comput. Sci., Dartmouth College, Hanover, NH, USA, Tech. Rep. TR2005-536, Aug. 2005.

[71] J. Wu, Y. Wang, L. Ding, and X. Liao, ''Improving performance of network covert timing channel through Huffman coding,'' *Math. Comput. Model.*, vol. 55, nos. 1–2, pp. 69–79, 2012.

[72] W. Gasior and L. Yang, ''Network covert channels on the Android platform,'' in *Proc. 7th Annu. Workshop Cyber Secur. Inf. Intell. Res. (CSIIRW)*, 2011, p. 1.

[73] X. Luo, E. W. W. Chan, and R. K. C. Chang, ''TCP covert timing channels: Design and detection,'' in *Proc. IEEE Int. Conf. Dependable Syst. Netw. With FTCS DCC (DSN)*, Jun. 2008, pp. 420–429, doi: 10.1109/DSN.2008.4630112.

[74] F. Iglesias, V. Bernhardt, R. Annessi, and T. Zseby, "Decision tree rule induction for detecting covert timing channels in TCP/IP traffic," in *Proc. Int. Cross-Domain Conf. Mach. Learn. Knowl. Extraction*. Cham, Switzerland: Springer, 2017, pp. 105–122.

[75] F. Iglesias, R. Annessi, and T. Zseby, "DAT detectors: Uncovering TCP/IP covert channels by descriptive analytics," *Secur. Commun. Netw.*, vol. 9, no. 15, pp. 3011–3029, 2016.

[76] F. I. Vázquez, R. Annessi, and T. Zseby, "Analytic study of features for the detection of covert timing channels in NetworkTraffic," *J. Cyber Secur. Mobility*, vol. 6, no. 3, pp. 245–270, 2017.

[77] M. A. Elsadig and Y. A. Fadlalla, "Packet length covert channel: A detection scheme," in *Proc. 1st Int. Conf. Comput. Appl. Inf. Secur. (ICCAIS)*, Apr. 2018, pp. 1–7, doi: 10.1109/CAIS.2018.8442026.

[78] G. Xu, W. Yang, and L. Huang, "Supervised learning framework for covert channel detection in LTE—A," *IET Inf. Secur.*, vol. 12, no. 6, pp. 534–542, Nov. 2018.

[79] A. Nadler, A. Aminov, and A. Shabtai, "Detection of malicious and low throughput data exfiltration over the DNS protocol," *Comput. Secur.*, vol. 80, pp. 36–53, Jan. 2019.

[80] N. Ishikura, D. Kondo, V. Vassiliades, I. Iordanov, and H. Tode, "DNS tunneling detection by cache-property-aware features," *IEEE Trans. Netw. Service Manage.*, vol. 18, no. 2, pp. 1203–1217, Jun. 2021.

[81] İ. G. Çavuşoğlu, "Covert channel detection using machine learning methods," M.S. thesis, Dept. Comput. Eng., Middle East Tech. Univ., Ankara, Turkey, 2019.

[82] J. Zhang, L. Yang, S. Yu, and J. Ma, "A DNS tunneling detection method based on deep learning models to prevent data exfiltration," in *Network and System Security*. Cham, Switzerland: Springer, 2019, pp. 520–535.

[83] S. Chen, B. Lang, H. Liu, D. Li, and C. Gao, "DNS covert channel detection method using the LSTM model," *Comput. Secur.*, vol. 104, May 2021, Art. no. 102095.

[84] I. Jawad, J. Ahmed, I. Razzak, and R. Doss, "Identifying DNS exfiltration based on lexical attributes of query name," in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–7, doi: 10.1109/IJCNN52387.2021.9534276.

[85] C. H. Rowland, "Covert channels in the TCP/IP protocol suite," *First Monday*, vol. 2, no. 5, 1997, doi: 10.5210/fm.v2i5.528.

[86] M. Bittan, A. Michal, and R. Hennion. (2016). *Cyber Academy: Formations et Certifications en Cyberscurit*. (Jan. 2019). [Online]. Available: http://www. hsc.fr/ressources/outils/dns2tcp/

[87] P. Yang, Y. Li, and Y. Zang, "Detecting DNS covert channels using stacking model," *China Commun.*, vol. 17, no. 10, pp. 183–194, Oct. 2020.

[88] S. Al-Eidi, O. Darwish, Y. Chen, and G. Husari, "SnapCatch: Automatic detection of covert timing channels using image processing and machine learning," *IEEE Access*, vol. 9, pp. 177–191, 2021.

[89] T. A. V. Sattolo, "Real-time detection of storage covert channels," Ph.D. dissertation, Dept. Syst. Comput. Eng., Carleton Univ., Ottawa, ON, Canada, 2021.

[90] P. Yang, X. Wan, G. Shi, H. Qu, J. Li, and L. Yang, "Identification of DNS covert channel based on stacking method," *Int. J. Comput. Commun. Eng.*, vol. 10, no. 2, pp. 1–15, 2021.

[91] G. Farnham and A. Atlasis, "Detecting DNS tunneling," *SANS Inst. InfoSec Reading Room*, vol. 9, pp. 1–32, Aug. 2013.

[92] A. Karasaridis, K. Meier-Hellstern, and D. Hoein, "Detection of DNS anomalies using flow data analysis," in *Proc. Global Telecommun. Conf. (GLOBECOM)*. IEEE, 2006, pp. 1–6.

[93] S. Shafieian, D. Smith, and M. Zulkernine, "Detecting DNS tunneling using ensemble learning," in *Proc. Int. Conf. Netw. Syst. Secur.* Cham, Switzerland: Springer, 2017, pp. 112–127.

[94] T. Zseby, F. I. Vázquez, V. Bernhardt, D. Frkat, and R. Annessi, "A network steganography lab on detecting TCP/IP covert channels," *IEEE Trans. Educ.*, vol. 59, no. 3, pp. 224–232, Aug. 2016, doi: 10.1109/TE.2016.2520400.

MUAWIA A. ELSADIG received the bachelor's degree in computer engineering, the M.Sc. degree in computer networks, and the Ph.D. degree in computer science (information security). Currently, he is an Assistant Professor of cybersecurity with the Deanship of Scientific Research, Imam Abdulrahman Bin Faisal University (IAU), Dammam, Saudi Arabia. He worked for different accredited international universities and has a rich record of publications at recognized international journals and conferences. He has many years of teaching experience and considerable industry contributions. He contributed as a reviewer for many international reputable journals and received many awards for his research activity. His research interests include the area of information security, network security, cybersecurity, wireless sensor networks, bioinformatics, information extraction, and ranging from theory to design to implementation.

AHMED GAFAR received the B.Sc. and M.Sc. degrees from Omdurman Islamic University, in 2009 and 2011, respectively. He is currently working as a Lecturer with the Deanship of Scientific Research, Imam Abdulrahman Bin Faisal University, Saudi Arabia. His research interests include statistical prediction models, machine learning, and its associated applications.