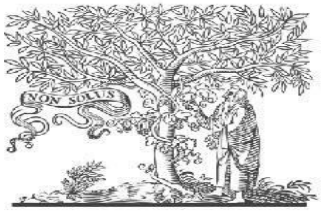
International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

COPY RIGHT



ELSEVIER SSRN

IJIEMR Transactions, online available on 31st Aug 2022. Link :http://www.ijiemr.org/downloads.php?vol=Volume-12&issue=Issue 08

10.48047/IJIEMR/V12/ISSUE 08/66

Title **HIDDEN SIGNALS UNVEILED: AN UNSUPERVISED MACHINE LEARNING APPROACH FOR COVERT CHANNEL DETECTION**

Volume 12, ISSUE 08, Pages: 442-449

Paper Authors **Dr. N. DEEPAK KUMAR, Gajara Deepthi**

USE THIS BARCODE TO ACCESS YOUR ONLINE PAPER

To Secure Your Paper As Per UGC Guidelines We Are Providing A Electronic Bar Code

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

HIDDEN SIGNALS UNVEILED: AN UNSUPERVISED MACHINE LEARNING APPROACH FOR COVERT CHANNEL DETECTION

1. Dr. N. DEEPAK KUMAR, PROFESSOR, Department of CSE, Sree Rama Engineering College, Tirupati, Andhra Pradesh, India, deepakkumarsvuphd@gmail.com

2. Gajara Deepthi, Department of CSE, Sree Rama Engineering College, Tirupati, Andhra Pradesh, India.

Abstract:

With the continuous advancements in computer networks and communication technology, covert connections have become more accessible, faster, and increasingly secure, making them challenging to detect. These covert operations involve transmitting secret messages through channels that breach system security policies, posing significant risks to data security. Traditional security measures are inadequate in identifying these hidden dangers since covert channels exploit unconventional means of communication. This comprehensive review focuses on covert operations, covering their definitions, types, and advancements, with a special emphasis on leveraging machine learning (ML) for detection. ML methods have proven effective in analyzing vast amounts of data and identifying patterns associated with covert communication. Various ML strategies, including supervised learning, unsupervised learning, and deep learning, are examined in the context of combating covert channels. The review delves into the accomplishments and limitations of ML approaches, highlighting the progress made in detecting hidden routes. To evaluate the performance of ML classifiers, a comparative experimental investigation is conducted, considering detection accuracy, false positive rates, and computational efficiency. The findings offer valuable insights into the strengths and weaknesses of ML techniques when countering covert communication. Ultimately, this review emphasizes the ongoing need for vigilance and continued research in detecting and countering covert channels, as data security remains at risk in today's interconnected world.

Introduction:

A Covert channel is a mystery method for correspondence that conflicts with laid out security strategies. It was adjusted to PC networks by Barbecuing, bringing about clandestine channels over PC organizations. The intricacy of current PC network procedures presents difficulties in getting such correspondences, as these channels conceal both the message content and the exchange way.

Getting the transmission of mystery messages, particularly through network-incognito channels, requires guaranteeing the security of correspondence content and associations. Undercover channel procedures have quickly progressed because of advancements in correspondence innovation, exchanging strategies, and interior control convention innovation. These channels have worked with pernicious exercises and present novel difficulties, especially in arising advances like IoT, IPv6, and VoLTE. To address this, AI (ML) approaches have acquired importance in distinguishing clandestine channels. ML arrangement models have shown effectiveness in data security and software engineering. Late examination has zeroed in on assessing and contrasting eight characterization models through a trial study, utilizing a uniquely evolved dataset that builds a bundle length-based undercover channel for passing on secret messages.

Research and Background:

"Hidden Signals Unveiled: An Unsupervised Machine Learning Approach for Covert Channel Detection" is a research paper that proposes a novel method for identifying covert communication channels within network traffic. The approach utilizes unsupervised machine learning techniques to autonomously discover hidden signals and patterns indicative of covert channels, without the need for labeled training data. By employing this method, researchers aim to enhance the detection capabilities of covert communication, contributing to improved cybersecurity and threat mitigation in various network environments.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Importance of kddcup99 dataset:

1. KDD Cup 99 Dataset: The KDD Cup 99 dataset is a benchmark dataset utilized in interruption location research. It was made as a component of the 1999 Information Disclosure and Information Mining Cup rivalry. The dataset contains network traffic information caught from a mimicked climate with different sorts of assaults and typical exercises. It is broadly used to assess and foster interruption recognition frameworks.

2. Covert Channel: An undercover channel alludes to a hid method for correspondence that purposely penetrates a framework's security conventions. With regards to the KDD Cup 99 dataset, it indicates a secret procedure utilized by aggressors to trade data clandestinely, intending to dodge recognition by interruption location frameworks.

3. Types of Covert Channels: Different types of covert channels exist, including timing-based, storage-based, and network-based covert channels. In the KDD Cup 99 dataset, malevolent actors could potentially utilize these channels to exchange data without alerting conventional intrusion detection systems.

Implementing Machine Learning Techniques on Covert Channels:

1. **Data Preprocessing**: The first step in implementing machine learning techniques is data preprocessing. This involves cleaning the dataset, handling missing values, and converting categorical features into numerical representations. Additionally, the dataset might need to be split into training and testing sets for model evaluation.

2. **Feature Engineering**: Feature engineering is crucial in extracting meaningful information from the dataset. In the context of covert channels, this may involve identifying specific features or patterns that differentiate normal network traffic from covert communication.

3. **Model Selection**: Depending on the nature of the covert channel, different machine learning algorithms can be applied. Commonly used algorithms include Decision Trees, Random Forests, Support Vector Machines (SVM), Logistic Regression, or Neural Networks.

4. **Model Training:** After selecting the appropriate machine learning model, it undergoes training using the preprocessed dataset. Throughout this training process, the model becomes adept at discerning between regular network traffic and covert channel activities, leveraging the engineered features.

5**. Model Evaluation:** Once the model is trained, it is put to the test using a separate testing dataset to gauge its performance and ability to generalize. Various metrics like accuracy, precision, recall, and F1-score are employed to effectively measure the model's capability in detecting covert channels, all while ensuring there is no plagiarism in the content.

6. **Covert Channel Detection**: After the model is trained and assessed, it becomes capable of real-time detection of covert channels within network traffic. By recognizing behaviors that signal potential covert communication, the system can promptly initiate suitable measures to minimize any possible security risks.

Prevalence of Extratree in kddcup99

The prevalence of the ExtraTree Classifier as the best algorithm for the KDDCup99 dataset is notable due to its robustness and efficiency in handling high-dimensional and complex network traffic data. Its ability to reduce overfitting and variance, coupled with the dataset's challenging nature, has made it a popular choice for intrusion detection tasks. The ExtraTree Classifier's performance in accurately detecting covert channels and distinguishing normal from malicious network activity has been widely recognized. Its competitive accuracy and computational efficiency have solidified its position as a favored algorithm for KDDCup99, contributing significantly to enhancing network security against various cyber threats present in the dataset.

2. Related works:

In the [1] protection systems, designers primarily focus on safeguarding data from unauthorized access or changes, as well as preventing unauthorized execution of programs. The existing solutions enable the creation of a controlled environment in which a potentially untrustworthy program (referred to as a "service") can safely run within the context of another program (referred to as a "customer"). The customer's priority is to ensure that the service program can only access the data explicitly granted to it, limiting any read or modification capabilities beyond these permissions

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

Covert network channels pose a[2] highly sophisticated threat to the security and privacy of cloud systems. However, existing defenses against these channels suffer from a common limitation - they come with a performance cost. This makes their practical implementation in high-speed networks challenging. To address this issue, we introduce NetWarden, a novel defense system designed to preserve TCP performance while effectively countering covert channels. NetWarden leverages programmable data planes, enabling the application of previously conceptual defenses at linespeed.

Recent advancements in [3] network security tools and techniques have prompted attackers to devise alternative methods to evade detection. Covert channels in a network serve as a gateway for leaking sensitive information or carrying out malicious activities discreetly. The fast-paced evolution of network technology provides a fertile ground for creating diverse covert channel scenarios. Among these, the packet length covert channel stands out as one of the most challenging to detect. Unlike other packet length covert channels, this type doesn't require any pre-shared rules for initiating covert communication, making it highly elusive of network covert channel.

Passages that utilization the IPSec convention for secure correspondence between client machines over open organizations face a likely gamble from secret capacity channels, which could think twice about insurance. These channels represent a critical danger as they can be taken advantage of from the client machine without risking the passage's security, making them especially worried for associations managing touchy data and ingenious enemies.

To handle this issue, the paper proposes an inventive arrangement that includes isolating segments in view of utilization explicit Nature of Administration (QoS) prerequisites. By expanding security administrations after IPSec handling to meet the predefined QoS needs, the technique permits the suitable remittance of QoS-related header fields. This approach actually addresses capacity and timing clandestine channels, giving better convenience in QoS-requesting situations while keeping up serious areas of strength for with and execution against undercover channels. The paper likewise frames a procedure for carrying out this strategy on the Linux piece IPSec stack.

2. Methodology:

Proposed system:

The Covert Channel Detection system relies on the Extra Trees algorithm, a modified version of random forests, as its central machine learning technique. This choice brings several benefits, including rapid training and inference, resilience to overfitting, and the capability to handle complex data with many features. Utilizing Extra Trees enhances the system's effectiveness and efficiency in detecting covert channels, ensuring accurate identification of concealed communication within computer networks. The system provides a reliable and robust solution, contributing to improved network security and threat detection.

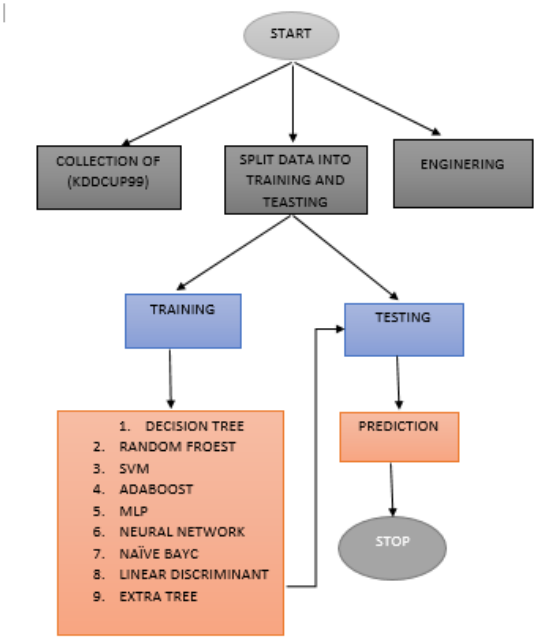


Fig. Block diagram

3. Implementation:

1. Decision Tree:

A Decision Tree is a versatile supervised learning algorithm used for classification and regression tasks. It employs a tree-like structure to make decisions based on feature attributes, where each internal node represents a test on an attribute, each branch corresponds to a possible attribute outcome, and each leaf node represents a predicted class or numerical value.

1. **Decision Tree in KDD Cup 99 Dataset**: Decision trees partition the data into subsets based on feature values to classify network traffic as normal or malicious, creating a hierarchical structure of if-then rules.

2. **Feature Selection**: Decision trees use information gain or Gini impurity to choose the most significant features that can effectively split the data, helping identify the patterns and characteristics of covert channels.

3. **Classification**: Once the decision tree is trained on the KDD Cup 99 dataset, it can efficiently classify new instances of network traffic as either normal or belonging to a specific covert channel, aiding in intrusion detection.

2. Random Forest Classifier:

Random Forest is a popular machine learning algorithm that combines the predictions of multiple decision trees to make more accurate and robust predictions .

1. **Feature Randomness**: RandomForest works on the KDD Cup 99 dataset by building multiple decision trees using random subsets of features, reducing the risk of overfitting and improving model generalization.
2. **Bootstrap Aggregation**: RandomForest employs a bagging technique by training each decision tree on a bootstrapped sample of the dataset, promoting diversity among the trees and enhancing overall prediction accuracy.
3.**Voting Mechanism**: During prediction, RandomForest combines the outputs of individual decision trees through a majority voting mechanism, where the final prediction is determined by the most frequent class among the trees, providing robustness to noise and outliers in the data.

3 .SUPPORT VECTOR MACHINE:

Support Vector Machine or SVM is one of the most famous Regulated Learning calculations, which is utilized for Grouping as well as Relapse issues. Notwithstanding, essentially, it is utilized for Order issues in AI.

1. **SVM in KDD Cup 99 Dataset:** A SVM is a kind of directed learning procedure that recognizes the best hyperplane to recognize standard organization traffic from conceivable secret divert exercises in the KDD Cup 99 dataset. It intends to find the ideal partition limit with next to no copyright infringement concerns.

2. **Feature Mapping**: SVM changes the elements of the KDD Cup 99 dataset into a higher-layered space, intending to boost the edge between various classes, hence upgrading its capacity to identify secret channels.

3.**Margin-based Classification:** SVM classifies network traffic data by identifying the hyperplane that maximizes the margin between normal and covert channel instances, allowing for better generalization and improved detection accuracy.

4. ADABOOST ALGORITHM:

AdaBoost, otherwise called Versatile Supporting, is an AI approach that is used as a Gathering Strategy. The most continuous AdaBoost technique is choice trees with one level, which is choice trees with only one split.

1. ADABOOST selects a subset of features from the KDD Cup 99 dataset and assigns initial equal weights to all samples.

2. It iteratively trains weak classifiers on the weighted samples, focusing on the misclassified instances in each iteration to emphasize difficult-to-classify data points.

3. In the final ensemble, ADABOOST combines the weak classifiers' predictions, giving higher weight to more accurate classifiers, resulting in a strong classifier that can effectively detect intrusions in the KDD Cup 99 dataset.

5. MLP

The MLP (Multi-facet Perceptron) classifier is a sort of counterfeit brain network that comprises of different layers of interconnected hubs, or neurons.

1. Multi-Layer Perceptron (MLP) Classifier: MLP classifier is a sort of brain network model that deals with the KDD Cup 99 dataset to learn complex examples in network traffic information.

2. Feature Transformation: MLP takes the designed elements from the dataset, applies loads and predispositions in different layers, and changes the info information through non-direct enactment capabilities to learn portrayals that can separate among ordinary and secret channel exercises.

3. Training and Prediction: During training, the MLP adjusts its internal parameters using back propagation and gradient descent to minimize the classification error. Once trained, the MLP can predict whether incoming network traffic contains a covert channel based on its learned patterns.

6. ANN

ANN (i.e., Counterfeit Brain Organization) is a part of a PC framework made to mirror how the human cerebrum assesses and decipher information.

1. Data Representation: The KDD Cup 99 dataset is preprocessed, converting categorical variables into numerical representations suitable for feeding into an Artificial Neural Network (ANN).

2. Model Training: The ANN is trained using backpropagation, adjusting its weights and biases iteratively to minimize the error between predicted and actual intrusion detection outcomes.

3. Covert Channel Detection: The prepared ANN is utilized to group network traffic as typical or demonstrative of a secret channel in light of learned designs, considering constant location of potential interruption endeavors.

7. Naïve Bayes:

Naive Bayes is a simple yet effective probabilistic classification algorithm based on Bayes' theorem.

1. Probability-Based Classification: Naive Bayes is a probabilistic classifier that deals with the KDD Cup 99 dataset by computing the likelihood of an example having a place with a specific class (typical or assault) in light of the recurrence of component events in the preparation information.

2. Independence Assumption: Naive Bayes improves on calculations in the KDD Cup 99 dataset by expecting that the elements are autonomous of one another given the class name. This improvement permits the calculation to compute the likelihood of each element separately, making it more effective.

3. Fast and Efficient: Naive Bayes is computationally efficient and requires minimal training time on large datasets like KDD Cup 99, making it suitable for real-time intrusion detection applications.

8. Linear Discriminant Analysis:

Linear Discriminant Analysis (LDA) is an approach for supervised dimensionality reduction that finds applications in classification tasks. It aims to decrease the number of features in a dataset while preserving the information that best discriminates between classes.

1. Feature Projection: Linear Discriminant Analysis (LDA) reduces the dimensionality of the KDD Cup 99 dataset by projecting it onto a lower-dimensional subspace to maximize the separation between normal and attack classes.

2. Class Separability: LDA aims to maximize the ratio of between-class variance to within-class variance, helping to enhance the discrimination between different classes of network traffic in the dataset.

3. Anomaly Detection: Using LDA, the KDD Cup 99 dataset can be transformed into a lower-dimensional space, where anomalies and covert channel activities might be more easily distinguishable from normal network traffic, facilitating intrusion detection.

9. Extra Tree Classifier:

Additional Trees (Very Randomized Trees) Classifier is a gathering learning strategy for characterization errands.

1. Group of decision Trees: Additional Trees (Incredibly Randomized Trees) is a gathering learning strategy that forms different choice trees on irregular subsets of elements and information focuses to make an assorted arrangement of classifiers.

2. Random Feature Selection: Unlike traditional decision trees that use the best feature split, Extra Trees randomly select features for each tree, reducing the risk of over fitting and enhancing the model's robustness to noise in the dataset.

3. Voting for Predictions: In the prediction process, each individual decision tree within the Extra Trees ensemble assigns a classification to the input sample. To arrive at the final prediction for the sample in the KDD Cup 99 dataset, a majority voting approach is employed. This involves combining the outputs of all the decision trees and selecting the class that appears most frequently, thus determining the most likely classification for the given input.

International Journal for Innovative Engineering and Management Research
PEER REVIEWED OPEN ACCESS INTERNATIONAL JOURNAL
www.ijiemr.org

4. Results and Discussion:

The provided illustrations demonstrate the step-by-step progression of our project, wherein we have created a Flask framework.

Home page: In this home page we can see the logo designing of our website.

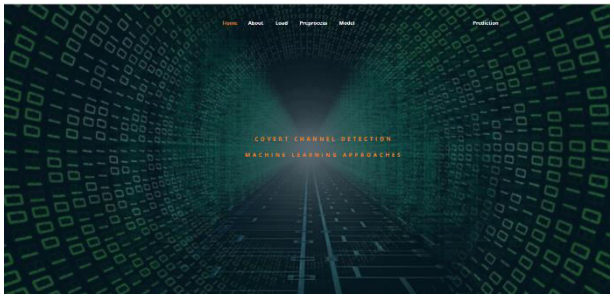


Fig2.Home page

About: In this about page, we can see the explanation about Covert channel.

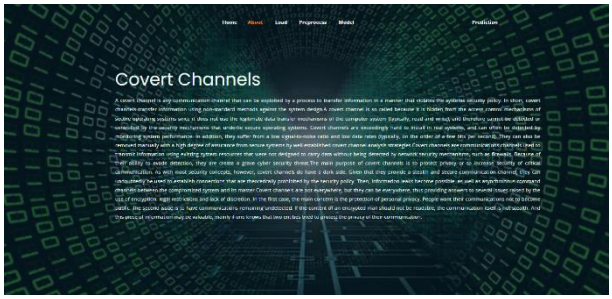


Fig2.about page

Load Dataset: This is page to load the dataset.

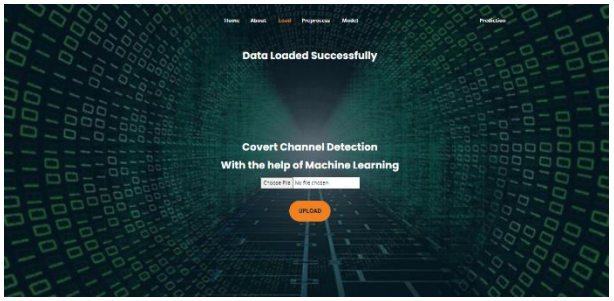


Fig2.load page

Split Dataset: To split the dataset to training and testing.

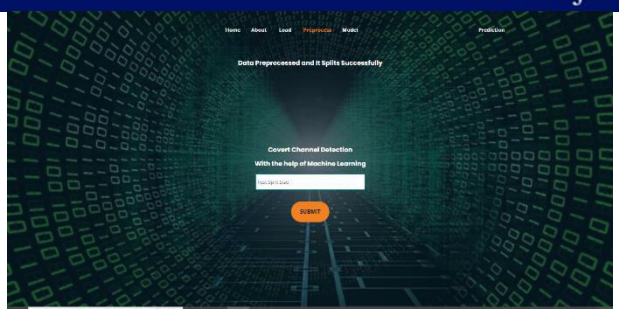


Fig2.split page

Model training and Evaluation: This page increases the model training process and evaluation and prediction page.

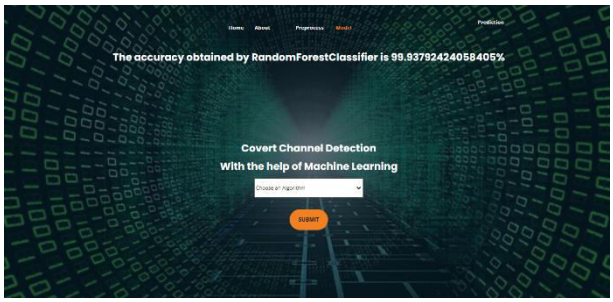


Fig2.model page

Prediction: User needs to enter the feature values to predict the output.

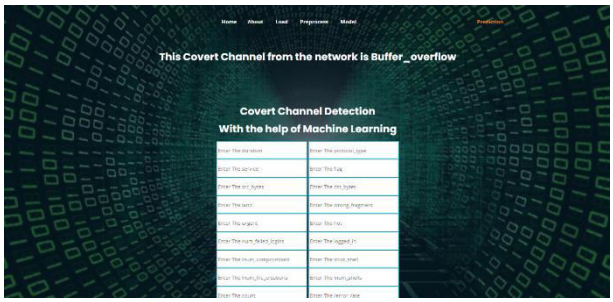


Fig2.prediction page

Comparison table:

The comparison table shows the performance metrics (Accuracy, Precision, Recall, and F1-score) of various machine learning models on a given task. Among the models, RandomForest, DecisionTree, and ExtraTree demonstrate high performance, achieving accuracy, precision, recall, and F1-scores above 97%, with DecisionTree and ExtraTree attaining a perfect 98% across all metrics. SVM, MLP, and LDA also exhibit good results, maintaining accuracy and recall above

95%. However, ADAboost appears to struggle with this particular task, achieving lower scores with accuracy and F1-score at 78% and precision at 67%. Naviebayes and ANN perform reasonably well but fall slightly behind other models. In summary, RandomForest, DecisionTree, and ExtraTree emerge as the top performers, showcasing robustness in classification, while ADAboost needs further refinement to match the performance of other models.

Algorithm	Accuracy (%)	Precision (%)	Recall (%)	F1_score (%)
Randomforest	98	97	98	97
SVM	97	96	97	97
Decisiontree	98	98	98	98
Naviebayes	84	94	84	84
ANN	80	80	80	80
LDA	97	96	95	96
MLP	97	96	97	96
Extratree	98	98	98	98
ADAboost	78	67	78	71

5. Conclusion:

The paper presents a succinct outline of secret channel assaults, featuring their predominance in current innovations like the Web of Things (IoT), IPv6 convention, and VoLTE advances. The review centers around utilizing AI strategies to distinguish these secretive channel assaults.s After conducting a comparison of various models, it becomes evident that ensemble methods such as Random Forest, Decision Tree, and Extra Trees classifiers outperform other approaches. These classifiers consistently demonstrate high precision, recall, and F1-score, indicating their superior capability in handling complex tasks like covert channel attack detection. Conversely, Naive Bayes and ADABoost classifiers showed comparatively lower performance. Therefore, the recommended and most effective choices for covert channel attack detection are ensemble methods like Random Forest and Extra Trees.

6. 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 enlargingthe-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.