future internet

MDPI

Article

Enhancing Network Security: A Machine Learning-Based Approach for Detecting and Mitigating Krack and Kr00k Attacks in IEEE 802.11

Zaher Salah [1,*] and Esraa Abu Elsoud [2]

[1] Department of Information Technology, Faculty of Prince Al-Hussein bin Abdullah II for Information Technology, The Hashemite University, P.O. Box 330127, Zarqa 13133, Jordan
[2] Department of Computer Information Systems, Faculty of Prince Al-Hussein bin Abdullah II for Information Technology, The Hashemite University, P.O. Box 330127, Zarqa 13133, Jordan; esraa.alkhawaja20673@gmail.com
* Correspondence: zaher@hu.edu.jo

Abstract: The rise in internet users has brought with it the impending threat of cybercrime as the Internet of Things (IoT) increases and the introduction of 5G technologies continues to transform our digital world. It is now essential to protect communication networks from illegal intrusions to guarantee data integrity and user privacy. In this situation, machine learning techniques used in data mining have proven to be effective tools for constructing intrusion detection systems (IDS) and improving their precision. We use the well-known AWID3 dataset, a comprehensive collection of wireless network traffic, to investigate the effectiveness of machine learning in enhancing network security. Our work primarily concentrates on Krack and Kr00k attacks, which target the most recent and dangerous flaws in IEEE 802.11 protocols. Through diligent implementation, we were able to successfully identify these threats using an IDS model that is based on machine learning. Notably, the resilience of our method was demonstrated by our ensemble classifier's astounding 99% success rate in detecting the Krack attack. The effectiveness of our suggested remedy was further demonstrated by the high accuracy rate of 96.7% displayed by our neural network-based model in recognizing instances of the Kr00k attack. Our research shows the potential for considerably boosting network security in the face of new threats by leveraging the capabilities of machine learning and a diversified dataset. Our findings open the door for stronger, more proactive security measures to protect IEEE. 802.11 networks' integrity, resulting in a safer online environment for all users.

Keywords: wireless; IDS; machine learning; Krack; Kr00k; IEEE8021.11

check for updates

Citation: Salah, Z.; Abu Elsoud, E. Enhancing Network Security: A Machine Learning-Based Approach for Detecting and Mitigating Krack and Kr00k Attacks in IEEE 802.11. *Future Internet* **2023**, *15*, 269. https://doi.org/10.3390/fi15080269

Academic Editors: Mario Di Mauro, Francesco Pascale and Marco Tambasco

Received: 19 July 2023
Revised: 6 August 2023
Accepted: 8 August 2023
Published: 14 August 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

The next generation of wireless networks will require a unified platform to support the vast numbers of devices, users, and services with different data rates and latency requirements. Current wireless technologies like 3G and 4G-LTE have several limitations that restrict any possible enhancement of the systems to meet these demands. Accordingly, researchers have developed an advanced wireless communication technology called 5G to satisfy the requirements above. After several scientific research studies, it was found that this fifth-generation technology has limitations too: it cannot be used for long-distance communication or low-power wide-area technology. This indicates that current communication technology will not be fully and effectively able to meet demands in the future. A highly advanced digital civilization backed by limitless wireless connectivity is also anticipated to have arisen by the year 2030 [1].

Within the context of 5G technology, the Internet of Things (IoT) has arisen as a groundbreaking idea, where technologies and solutions are incorporated to connect objects, people, platforms, and software via the Internet. To create comprehensive IoT networks, devices

will be endogenously fully equipped with IoT modules that enable D2D communication with one another [2]. Furthermore, RAT will be supported by 5G to connect these devices. New radio technologies, such as NOMA, massive MIMO, mmWave, and several other IoT communication technologies, will be introduced in the 5G network.

One of the primary requirements for 5G systems and beyond is security related to 5G technologies. To investigate security in networks, 4G networks use cryptography protocols for user authentication [3], while 3G networks use two-way authentication to prevent connection establishment with fake base stations [4]. A new age has begun with the introduction of 5G technology, which presents distinct security and privacy challenges beyond those faced by earlier systems. An innovative approach to security procedures is necessary given the changing architecture and the introduction of new services. Researchers have added new ideas like visibility and centralized policy in addition to the fundamental security principles of confidentiality, integrity, and availability. These updates are intended to strengthen data protection in the face of evolving threats by enhancing security safeguards [5,6] as shown in Figure 1.

Figure 1. Evolved security requirements for 5G technologies.

One of the most important security requirements in the 5G security model is data confidentiality; it is the parameter that can protect the transmission data from disclosure to unauthorized entities: it means ensuring that the sender's message in the 5G network is only readable by the proposed destination. To preserve data secrecy in the context of 5G network applications, encryption methods have evolved into essential instruments. Data within 5G networks can be safely secured and decrypted by using symmetric key encryption. Sensitive information is kept secure by being protected from unauthorized access or interception due to this critical security defense [7].

Data integrity relates to keeping data from being altered or modified while transformed from one location to another. To achieve the principle of integrity authentication technique used by 5G-AKA (authentication and key agreement), the fact that the 5G new radio (NR) offers user integrity protection is a significant improvement in 5G security. This is significant because user plane integrity protection was not supported by 4G. Small data transmissions can take advantage of this new feature, especially for IoT devices with constrained bandwidth [8,9].

Availability issues in most cases are related to DoS attacks and are conducted in a wireless network by jamming. Spread-spectrum techniques can prevent jamming effectively, but they cannot be used in IoT nodes with limited resources (e.g., sensors). Fifth-generation networks do not implement new techniques to test availability.

A centralized security policy can protect all enterprise endpoints from external threats and can be investigated through authentication. All network layers must be covered by comprehensive end-to-end security strategies for 5G networks. Fifth-generation network operators need to have full visibility, control, and monitoring of overall network layers to implement such a thorough security mechanism. To manage and control security policies, open application program interfaces (APIs) should be combined with 5G technologies.

The 5G network can thus have uniform software and hardware security policies. The implementation of the security mechanism in new 5G services will become easier with high visibility across the network. Furthermore, improved visibility helps in preventing and isolating the threats before attacks happen [7]. However, most research focuses on the three core requirements for 5G security, the CIA [10–12]. Figure 2 presents the core requirements for 5G security and the solutions to maintain these requirements.

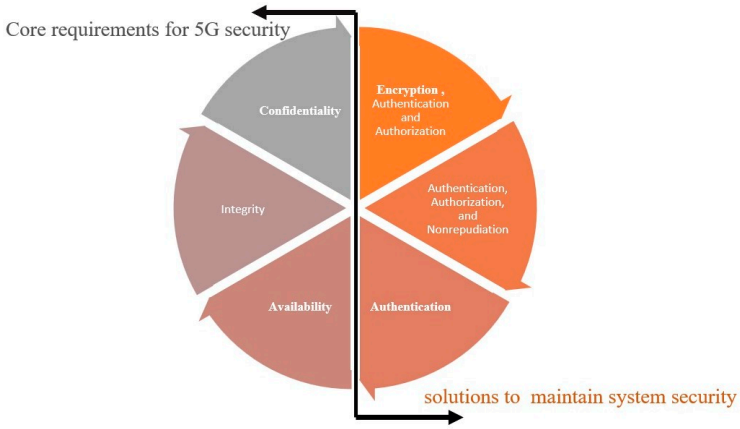

Figure 2. Core requirements for 5G security (**left**) and solutions to maintain system security (**right**).

A comprehensive security policy is the starting point for managing the countermeasures needed to secure wireless networks. Technical countermeasures that help in wireless security environments include hardware and software. Hardware countermeasures like smart cards, VPNs, and biometrics are hardware solutions while proper AP configuration, software patches, authentication, intrusion detection systems (IDS), and encryption are all examples of software countermeasures [13].

Researchers have put forward suggestions based on cybersecurity defensive systems, particularly the well-known IDS, to protect against cyber threats from wireless communications by anticipating and resolving flaws. Due to the heterogeneity and massive amounts of unstructured data present in the network, IDS procedures are useless for the real-time detection of potential intrusions in 5G [14]. One of the most promising methods in the field of artificial intelligence (AI) over the past few years has been machine learning (ML). Its remarkable abilities enable systems to learn from enormous amounts of data, considerably more than humans can process. Organizations can use ML to evaluate large datasets, improve their understanding, and provide accurate predictions. With its unique ability to predict and avert issues, this disruptive technology opens the door for more proactive and effective approaches in a variety of fields [15–17].

The machine learning field has enabled different paths that are effective in handling network intruders. Therefore, the employment of ML tools in 5G systems has attracted much interest from international projects and research, such as in [18–21].

ML techniques can examine the features of the network data to distinguish between attacks and normal traffic. Some attributes improve the accuracy of the intrusion detection system, whereas the network data contains noisy attributes as well that decrease the detection accuracy. Feature selection techniques are considered crucial for an IDS because FS helps in increasing the accuracy of detection, which is used as the input to the learning approaches.

A reliable intrusion detection system must be implemented because the IEEE 802.11 protocol-based short-distance transmission wireless network has faced security issues. Many researchers have proposed various IDS systems, and it has been found that data-mining-based techniques are very effective at detecting abnormal network behavior in these systems [22]. Intrusion detection has new difficulties as network data keeps growing exponentially; these difficulties are caused by the data's nonlinear structure and enormous volume. The characteristics of the data, notably the existence of redundant features, have a

substantial impact on the efficacy of existing approaches. The AWID dataset has become a popular experimental dataset in the field of wireless network environments. The "curse of dimensionality" problem is exacerbated by the large complexity and intrinsic duplication of this dataset. Designing effective intrusion detection strategies requires addressing these issues [22]. The large volume of the blind spot regions in the dimensionality dataset can lead to significantly variable estimates of actual model performance when algorithm designers use insufficient sample sizes to train and evaluate algorithms for finding patterns in a complex construct. Due to this variability, it is challenging to predict how well a model performs on data that has not yet been observed [23]. The imbalance between benign and attack samples might increase the false positive rate in addition to high dimensionality, which can influence the effectiveness of any suggested IDS [24].

The AWID dataset was released in 2016 [25]. It is the first dataset of its kind that focuses on IDS, more specifically on WIDS. Studying the AWID dataset will help researchers to be familiar with 802.11 network vulnerabilities and attacks, as well as educate them on the true effects of these attacks in daily life. The AWID dataset has 156 features and 37 million packets. A new version of this data was published in 2021 [26] by capturing and analyzing the traces of attacks that were sent into the IEEE 802.1X extensible authentication protocol (EAP) environment. It focuses on WPA2 Enterprise, 802.11w, and Wi-Fi 5. It includes multi-layer attacks like Krack and Kr00k.

In the context of IEEE 802.11 networks, we make important advancements in the field of network security in this paper. First, we carry out a focused examination of the most current and important vulnerabilities, concentrating particularly on Krack and Kr00k attacks.

Building upon this analysis, we develop an innovative and effective intrusion detection system (IDS) model using MATLAB. Our IDS model's main objective is to effectively identify and prevent Krack and Kr00k attempts to enhance network security. Our suggested model achieves excellent accuracy in identifying and mitigating targeted attacks by utilizing the strength of machine learning techniques.

We evaluate the IDS model using the AWID3 dataset (real-world datasets) to validate the effectiveness of our method, showing strong evidence of its utility in protecting networks from developing threats. We also provide a thorough comparison analysis of the three models, highlighting the advantages and benefits that each model makes in identifying Krack and Kr00k attacks. Additionally, we demonstrate the superiority of our strategy in-network security by thoroughly contrasting the results of our research with those of other cutting-edge techniques. Finally, we investigate the potential for proactive security measures by utilizing machine learning's capabilities in detecting and preventing new attacks, opening the way for more robust IEEE 802.11 networks.

The remainder of the paper is divided into the following sections: Section 2 offers a thorough analysis of the relevant work, examining research that has been undertaken in this area. The technique used in this study is presented in Section 3, along with the step-by-step process we used to create our IDS model. We show and discuss the findings from the experiments we conducted in Section 4, highlighting how well our IDS model performed in identifying Krack and Kr00k attacks.

Finally, we conclude our research in Section 5, where we go over the consequences, restrictions, and possible directions for further study in IEEE 802.11 security.

By arranging our work in this way, from the initial literature review to the concluding reflections on the significance of our findings, we hope to provide a thorough and coherent discussion of our research.

2. Literature Review

The evolution of communication networks has resulted in an exponential rise in the number of Internet of Things (IoT) devices that are connected to Wi-Fi networks. These devices generate enormous amounts of data traffic, which can be malicious, and makes it difficult to detect such attacks. Feature selection is used to reduce the amount of data

for intrusion detection model classifiers by removing noisy information and choosing the best features in the data, which participates in improving the IDS performance and solving these challenges.

We will focus on the AWID dataset, a Wi-Fi network intrusion benchmark dataset introduced due to the lack of a dataset in wireless intrusion detection systems (WIDSs) where the oldest datasets were about IDSs in general [27]. Moreover, this dataset was the first dataset produced using wireless network traffic. AWID has been enhanced and extended to AWID3 by capturing and examining the traces of cyberattacks sent into the IEEE 802.1X extensible authentication protocol (EAP) environment. It concentrates on 5G wireless networks, 802.11 w, and WPA2 Enterprise. It includes multi-layer and modern attacks like Krack and Kr00k [26]. This section will present some previous studies that used the AWID dataset in their research.

In Ref. [28], the authors present a novel method that combines stacked feature extraction with weighted feature selection using deep learning techniques. This is a groundbreaking strategy for a Wi-Fi impersonation detection attack. The goal is to increase the detection of such attacks' precision and effectiveness. Three distinct algorithms—ANN, C4.5, and SVM—were used for the experiments and were developed and assessed using the AWID datasets as the basis.

By utilizing this method, the authors were able to identify and detect Wi-Fi impersonation assaults with an astounding accuracy rate of 99.918%. This represents an important development in the area and illustrates the effectiveness and potential of the suggested deep-feature extraction and selection strategy. The results of this study emphasize the significance of using sophisticated methodologies and utilizing extensive datasets to address the constantly changing problems caused by Wi-Fi impersonation attacks. The achieved accuracy rate underscores the value of ongoing research and development in increasing Wi-Fi security in addition to demonstrating the usefulness of the suggested approach. A. Diro et al. [29] used deep learning algorithms to identify and analyze critical attacks and threats on Internet of Things devices, particularly those that take advantage of weaknesses in wireless communications. They have achieved a high accuracy with 99.91% for the ISCX dataset and 98.22% for the AWID dataset.

In [30], the authors built a wireless intrusion detection system (IDS) designed specifically to operate access points in passive mode. The complex nature of wireless attacks, which frequently include the deceitful fabrication of fake access points to dupe unwary users, served as the inspiration for this strategy. The proposed IDS sought to efficiently detect and counteract such malicious activity by concentrating on the passive mode. When experimenting with the AWID dataset, the approach proposed by the authors produced encouraging results, obtaining an exceptional accuracy rate of 98%. These results demonstrate how well the suggested method works for precisely recognizing and thwarting wireless attacks. The IDS demonstrated its capacity to distinguish between legal and fraudulent access points by utilizing the passive mode and applying complex detection algorithms.

S. M. Kasongo et al. [31] proposed a feed-forward deep neural network and feature extraction-based wireless intrusion detection system (IDS). Two datasets, UNSW-NB15 and AWID, were used to assess the system's performance. They also contrasted their findings with those of well-known machine learning algorithms including Random Forest (RF), Support Vector Machine (SVM), Naive Bayes (NB), Decision Tree (DT), and k-Nearest Neighbor (kNN). Four categories—binary and multiclass attacks, full features, and selected features—were used to categorize the experimental research. They used the extra trees (ET) approach to reduce the number of features in the AWID dataset to 26. The test set had 115,128 instances, which made up 20% of the AWID-CLS dataset, whereas the training set had 359,115 instances. The suggested model's binary classification accuracy was 98.6% on the validation data and 98.69% on the test data. The highest accuracy for multiclass classification was 98.47% on the validation data and 98.59% on the test data. Furthermore, the performance of the suggested model showed increased accuracy outcomes when the number of attributes was decreased. The authors' high precision

percentage for binary classification was 99.67% on the validation data and 99.66% on the test data. Similar results were obtained for multiclass classification, where 99.78% of the validation data and 99.77% of the test data were correctly classified. These results demonstrate the efficiency of the suggested wireless IDS system by demonstrating its precision in classifying and identifying various attack types. Deep neural networks and feature extraction were combined, and the accuracy significantly increased, especially when the number of characteristics was decreased.

In [32], the authors addressed the imbalanced and high-dimensional network traffic issues with a system for intrusion detection. To improve classification accuracy, the suggested system makes use of feature selection and ensemble learning approaches. In this respect, the authors devised a hybrid strategy that combines the bat algorithm (BA) and correlation-based feature selection (CFS), which maximizes the effectiveness of the feature selection process and enhances classification accuracy. The authors applied their suggested model to the NSL-KDD, AWID-CLS-R-Tst, and CIC-IDS2017 datasets to assess its efficacy. The AWID-CLS-R-Tst dataset underwent several preprocessing processes. This required both the replacement of missing values with zeros and the filtering out of features with constant values. The original 155 features were consequently.

Condensed to 84 features, an ensemble classifier was used to build the classification model. The superiority of the suggested strategy was shown by the experimental findings. Regarding the AWID dataset specifically, a subset of just eight chosen attributes was used to obtain a remarkable accuracy rate of 99.52%. In contrast, the accuracy was 98.2% when feature selection was not used.

Hence, many kinds of literature studied the impact of network intrusion detection systems; Z. Aydın et al. [33] mentioned in their research both wireless and wired intrusion detection systems using the AWID dataset, which includes wireless network attacks, and the UNSW-NB15 dataset, which consists of a wired network attack. In addition, they focused on the other performance matrices that are critical in intrusion detection systems such as F1, recall, and precision. After they preprocessed the datasets, they remedied the imbalance problem via SMOTET (synthetic minority over-sampling technique), the feature selection was performed using XGBoost, and finally, Bayesian optimization was applied before applying different ML algorithms.

D. L. Robert Wilson [34] attempted to address the issue of dataset imbalance and enhance the performance measures of machine learning methods applied to intrusion detection in his study. The AWID-CLS-F-Trn and AWID-CLS-R-Trn subsets of the AWID dataset were the subject of the investigation. These subsets represent various techniques for classifying assaults and actual attacks, respectively. It is noteworthy that the AWID dataset comprises two identical datasets that only vary in labeling strategies. Wilson used feature selection (FS) approaches and included all features in the analysis to improve the performance metrics. After that, the collected findings were contrasted, paying close attention to flooding and impersonation attacks. During the training phase, independent feature drop and group feature drop techniques were used. The results showed that when employing the Random Forest (RF) classifier, the independent features approach performed better. The group feature drop, on the other hand, showed enhanced dropping patterns for the flooding attack. However, when using the Logistic Regression (LR) classifier, the drop patterns for both independent and group feature drops were constant and comparable. The results showed that the independent features strategy performed better when the Random Forest (RF) classifier was used. On the other hand, the flooding attack's group feature drop showed enhanced dropping patterns. The drop patterns for both independent and group feature drops were constant and similar when the Logistic Regression (LR) classifier was used, however.

In [35], utilizing tree-based classification algorithms like Random Forest, XGBoost, LightGBM, and CatBoost, the researchers examined the effects of feature selection. The feature set of the AWID dataset was reduced by the authors from 155 to 15 features using the Shapley additive explanations (SHAP) approach. Their investigation findings showed

which characteristics had the greatest impact on detection models. The features wlan.da, wlan.fc-subtype, and wlan.lc.ds were shown to be the most significant in the detection process. The AWID dataset's patterns of intrusion and attack were distinguished and identified with the use of these attributes. The results indicate the value of features in enhancing the performance of intrusion detection models and emphasize the effectiveness of feature selection based on tree-based classification methods. The authors were able to improve the accuracy and efficacy of the detection procedure by limiting the feature set and concentrating on the most useful qualities.

Regarding a scalable ML-based intrusion detection system for IoT, the authors in [36] addressed the disadvantages of centralized IDS for devices with limited resources by utilizing two approaches—semi-distributed and distributed. The authors used feature selection approaches followed by classification in their investigation of the AWID dataset. The dataset was split into three parts, each of which has 68 features. Seven features were chosen from each dataset after performing feature selection, giving the full training set a total of 21 features. A semi-distributed strategy was used to find the most accurate feature selection technique for each dataset. The distribution method's classifier was also chosen to be a multi-layer perceptron (MLP) classifier. Experiments were used to determine the efficacy of the two proposed structures. Even with a substantial CPU time of 186.26 s, an excellent accuracy of 99.97% was attained using the semi-distributed technique. While retaining a detection accuracy of 97.80%, the distributed technique showed the lowest CPU time of 73.52 s. These outcomes show that both suggested topologies performed well in the tests that were conducted. The semi-distributed method demonstrated great accuracy, albeit requiring more CPU time. In contrast, the distributed technique reduced CPU time while preserving acceptable detection accuracy.

The AWID2 dataset, which is the foundation of the wireless IDS literature and comprises a substantial collection of packets and its WEP-based infrastructure, has more than 150 different features. AWID has been enhanced, and AWID3 has been introduced by capturing and examining the traces of cyberattacks that were transmitted into the IEEE 802.1X extensible authentication protocol (EAP) environment. Our study seeks to fill important gaps and address tough challenges in-network security, particularly as it relates to IEEE 802.11 networks, based on the thorough literature review that has been provided. The literature already in existence has emphasized the rise of sophisticated attacks, including Krack and Kr00k, which pose serious risks to network integrity. Considering these discoveries, our main goal is to create an innovative and efficient intrusion detection system (IDS) model that is specially designed to identify and thwart these targeted attacks. We aim to achieve high accuracy and efficiency in recognizing malicious activity and minimizing any security breaches by leveraging the most current advancements in machine learning techniques.

3. Methodology

This chapter outlines the process for implementing the framework for intrusion detection in wireless networks using machine learning techniques.

Wireless technologies have increased rapidly in recent years. While serious efforts have been made to secure these technologies, most security measures have proven inadequate in practice. The AWID project aims to provide a solid basis for researchers to develop robust security mechanisms for current and future generations of wireless networks by providing tools, methodologies, and datasets, as the previous datasets were not specific to wireless networks.

Wi-Fi (IEEE 802.11) has taken over as the standardized technology for connecting digital devices in wireless local area networks due to the rise of smart portable devices such as smartphones, tablets, and Internet of Things (IoT) devices. Wi-Fi is frequently used in critical locations as well as in homes, businesses, and organizations. Unsurprisingly, extensive academic research has focused on 802.11 protocol security as well as Wi-Fi network security. With frequent modifications and corrective actions, vulnerabilities have been found in even the most recent versions of the software although these vulnerabilities

have existed for more than 20 years. Security in wireless technology is a major issue that has long remained unresolved. External security measures should therefore be used as crucial elements of 802.11 wireless networks for defending against known or unknown attacks [26].

The AWID dataset was extracted in 2016, and then it was developed into a new version in 2021 called AWID3. The main differences between the old version and the new one can be summarized as follows:

(1) AWID3 includes recently identified attacks against the 802.11 protocol, including well-known instances like Krack and Kr00k. This inclusion enables researchers to investigate and create practical defenses against these particular dangers within the framework of the dataset.
(2) A network's packet-level details are contained in the PCAP format used to supply the data in AWID3. Researchers now have access to extensive data that can be utilized to assess network features and meet specific research objectives. The dataset also includes the pairwise master key (PMK) and TLS keys.
(3) The enterprise versions of the 802.11 standards are the main emphasis of AWID3. Stronger security features, like support for alternative network architectures and the use of protected management frames (PMF), which were introduced with the 802.11 w revision, are often present in these versions. By focusing on enterprise versions, the dataset is more applicable to actual security issues.
(4) The link layer of the 802.11 protocol is initially targeted via attacks on the AWID3 dataset. These assaults, nevertheless, quickly spread to higher layers, affecting protocols that run at different levels of the network stack. Researchers can examine the interrelated nature of network vulnerabilities due to this comprehensive perspective of attack propagation.
(5) Every scenario in the dataset is covered in detail by AWID3. Researchers may undertake detailed analysis and evaluation with the help of this documentation, which also helps them grasp the nuances of assault scenarios.

3.1. Structure of AWID3 Dataset

The AWID3 dataset has been carefully curated to record and examine the traces of different assaults within the IEEE 802.1X extensible authentication protocol (EAP) environment. It is valuable and publicly available. It is significant for being the first dataset to offer a review of the IEEE 802.11 w standard, which is necessary for hardware to be approved for use with the WPA3 protocol. The AWID dataset, from which AWID3 was built, has 254 features, of which 253 are general features and one is used for labeling. The dataset is offered in CSV format for simple access and interoperability with many different data analysis tools and methodologies. A thorough understanding of network activity and attack patterns is made possible by the extracted features, which cover both the MAC (media access control) layer and the application layer. The dataset consists of 36,913,503 instances, 30,387,099 of normal traffic, and 6,526,404 malicious ones. Malicious traffic includes 13 types of attacks.

- Deauthentication Attack.
- Disassociation Attack.
- Re-association Attack.
- Rogue AP Attack.
- Krack Attack.
- Kr00k Attack.
- SSH Brute Force Attack.
- Botnet Attack.
- Malware.
- SSDP Amplification.
- SQL Injection Attack.
- Evil Twin.

- Website Spoofing.

In our research, we used two types of these attacks which are Krack and Kr00k, since these two types are the most recent type of attacks discovered in IEEE 802.11 [37].

3.1.1. Krack Attack

The Krack attack has been noted as a potential security risk to the current encryption techniques used to preserve and protect Wi-Fi networks for the past 15 years. Publicly available information on the Krack attack includes information about the attack itself. There is no guarantee that every device will have a patch and be protected from these attacks coming from any networked point [38,39]. The four-way handshake procedure, which is a crucial part of the IEEE 802.11 protocol has a serious weakness that allows any attacker to decode a user's communication without eavesdropping on the handshake or knowing the encryption key, according to [40]. This flaw results from the pairwise transient key (PTK) installation process' use of a particular message counter. It is vital to look at how keystreams are used in the encryption process to comprehend the decryption process. The plaintext and keystream are merged using the XOR (exclusive OR) technique to create the encrypted message that is sent from the client to the access point (AP). The PTK, which is derived using the AES (advanced encryption standard), is scrambled with several other factors to create the keystream. The vulnerability, though, only exists in the XOR operation's last phase. The logic flow of this step is connected to a fundamental mathematical feature that is exploited by the Krack vulnerability. Equation (1) shows how the plaintext (P) and keystream (KS), as shown in the paper, are combined to create the cipher text (E). The Krack hack uses this defect in the XOR method to decrypt the encrypted communications, putting the security of wireless networks using the IEEE 802.11 standard at risk.

$$E = P \oplus KS \tag{1}$$

An attacker could use two captured encrypted packets to decrypt them. Since the keystreams are identical, XORing the two ciphertexts results in the keystreams being canceled and leaving two plaintexts.

$$E1 = P1 \oplus KS1 \tag{2}$$

$$E2 = P2 \oplus KS2 \tag{3}$$

$$KS1 = KS2 = KS \tag{4}$$

Then:

$$E1 \oplus E2 = (P1 \oplus KS) \oplus (P2 \oplus KS) = P1 \oplus P2 \tag{5}$$

If the attacker were to accurately estimate or know $P1$, they could decrypt $P2$. The well-known first message that the AP or client sends after connecting can be used for this. The key WPA2 stream was designed to stop this exploitation, but Krack researchers have found a way around it. Most of the keystream is made up of the static variables PTK, GTK, flags, MAC addresses, and counters. The only variable that alters when communications are encrypted is the packet number. Because every encrypted communication will have a different packet number and unique keystream, XOR cancellation is not conceivable [41].

3.1.2. Kr00k Attack

Some Wi-Fi traffic that has been encrypted with WPA2 can be decrypted by a vulnerability called Kr00k. The security company ESET discovered this vulnerability in 2019. According to ESET, this loophole affects more than a billion devices. Devices with Wi-Fi chips that have not yet received a patch from Broadcom or Cypress are vulnerable to Kr00k. Most modern Wi-Fi-enabled devices, including smartphones, tablets, laptops, and Internet

of Things (IoT) devices, use these Wi-Fi chips [37]. Table 1 highlights the main differences between Krack and Kr00k attacks.

Table 1. Comparing Krack and Kr00k.

Krack	**Kr00k**
Krack is a series of attacks, exploited by attackers	Kr00k is a vulnerability in WPA2.
The basic idea of Krack is that the attacker can use the keystream to know the plain and the cipher text. The encryption employed to secure data packets transmitted over a Wi-Fi connection is impacted by Kr00k. Typically, a unique key determined by the user's Wi-Fi password is used to encrypt these packets. Researchers from ESET claim that during the "disassociation" process, this key is reset for Broadcom and Cypress Wi-Fi chips to an all-zero value.	
Exploited during the 4-way handshake	Exploited after a disassociation.
Because it exploits implementation flaws in the WPA2 protocol itself, it affects most Wi-Fi-capable devices. Identified Broadcom and Cypress components used in mobile phones, tablets, laptops, and IoT devices.	

3.2. Preprocessing Steps

As we mentioned before, the dataset consists of 36,913,503 instances, 30,387,099 of normal traffic, and 6,526,404 malicious ones.

There are 49,990 instances for the Krack attack and 186,173 instances for the Kr00k attack.

We will implement our experiment in two phases; the first phase consists of two classes (Krack, normal) and (Kr00k, normal). While the second phase consists of multi-class (Krack, Kr00k, and normal).

To highlight the importance of the preprocessing of the dataset before using it in the proposed model, we have used the chosen sample without any preprocessing and feature selection techniques. The first sample consists of 106,971 kr00k traffic and 106,791 normal ones, while the second sample consists of 33,180 Krack traffic and 34,000 normal ones, with 254 features for both samples, as shown in Table 2.

Table 2. Training and testing samples in AWID3 dataset.

Samples	**Attack**	**Normal**
First Sample	106,971 Kr00k traffic	128,093
Second Sample	33,180 Krack traffic	34,000

We chose the following machine learning algorithms:

(1) Decision tree: the process for building a decision tree and the most used criteria for splitting the data [42]:

- Calculate an impurity measure for the entire dataset (e.g., Gini impurity or entropy).
- For each feature, calculate the impurity measure of splitting the data based on the values of that feature.
- Choose the feature that produces the lowest impurity measure after splitting the data.
- Split the data based on the chosen feature and repeat the process for each resulting subset of data until a stopping criterion is met (e.g., a maximum depth is reached or the number of samples in a leaf node is below a certain threshold).

The equations for calculating impurity measures depend on the specific criterion being used. For example, the Gini impurity measure for a set of samples S with C classes is:

$$G(s) = 1 - \sum_{i=1}^{c} P_i^2 \quad (6)$$

where p_i is the proportion of samples in S that belong to class i. The entropy impurity measure for the same set of samples S is:

$$H(s) = -\sum_{i=1}^{c} p_i \, log_2 \, p_i \tag{7}$$

where p_i is the same as above.

These impurity measures are used to evaluate the quality of each split and to choose the feature that produces the lowest impurity measure.

(2) Ensemble classifiers: combine multiple individual classifiers into a single ensemble classifier to improve the overall predictive performance. There are different types of ensemble classifiers, such as bagging, boosting, and stacking, and the equations used for each type can vary.

(3) SVM: Support Vector Machine (SVM) is a popular machine learning algorithm for classification, regression, and outlier detection. The main idea behind SVM is to find a hyperplane that separates the data into different classes with the largest margin possible. The equations used in SVM [43] are as follows:

$$f(x) = sign \, (\sum_{i=1}^{N} \alpha_i \, y_i \, K(x, x_i) + b) \tag{8}$$

where $f\,(x)$ is the predicted class label, α_i is the Lagrange multiplier for the i-th training sample, y_i is the class label of the i-th training sample (either +1 or -1), $K(x, x_i)$ is the kernel function that maps the input features x and x_i to a higher-dimensional space, and b is the bias term.

(4) Kernel: A kernel function is a function that maps the input data into a higher-dimensional space, where it is easier to find a separating hyperplane. The equation of linear Kernal [44] is as follows:

$$K\left(x_i \, , \, x_j\right) = x_i^T \, x_j \tag{9}$$

where x_i and x_j are the input features of the i-th and j-th training samples, respectively.

(5) KNN: K-Nearest Neighbors (KNN) is a simple, yet effective machine learning algorithm used for classification and regression tasks. The basic idea behind KNN is to find the K-nearest training samples to a given test sample based on a distance metric, and then use the labels of the K-nearest neighbors to predict the label of the test sample. The equation of KNN can be represented as follows [42]:

$$d\,(xi,\ xj) = \sqrt{\sum_{k=1}^{p} \left(x_{ik} - x_{jk}\right)^2} \tag{10}$$

where p is the number of features in each sample.

(6) Neural Network: Neural networks are a powerful class of machine learning algorithms that are inspired by the structure and function of the human brain. A neural network consists of multiple layers of interconnected processing units called neurons, and the input data are processed through the network in a forward pass, with the output of each layer serving as the input to the next layer.

After applying different ML algorithms, using the cross-validation $K = 10$, the accuracy results were very low, which is to be expected. The results are presented in Table 3.

Table 3. The performance of the learning algorithms without preprocessing.

Algorithm	Accuracy/Kr00k Attack	Accuracy/Krack Attack
Decision tree	22.10%	50%
Ensemble classifier	44.20%	50%
SVM	12.50%	failed
Kernel	12.50%	failed

The accuracy results proved the importance of preprocessing steps since it is a constructive and essential step for obtaining the correct data required to build a classifier, as shown in several types of research such as [44–48]. Data preprocessing, which aims to convert the raw data into a simpler and more efficient format for subsequent processing steps, is a crucial step in the knowledge discovery process because quality decisions must be based on quality data. Thus, the preprocessing procedures were carried out on the AWID3 dataset. The AWID3 consists of 13 CSV files with 36,913,503 instances, 30,387,099 of normal traffic, and 6,526,404 malicious ones. That has been studied and is well understood.

3.2.1. Detecting Krack Attacks

According to the importance of preprocessing steps as we mentioned before, the preprocessing procedure for the Krack dataset sample was as follows:

1. Deleting the constant and empty features.
2. Ignoring features that have more than 60% missing values.
3. Replace missing values with NaN.

The remaining dataset consists of 67 features and 67,180 instances, 33,180 Krack traffic and 34,000 normal traffic.

After preprocessing the data, we applied different ML algorithms. Table 4 shows the performance of the learning algorithms after preprocessing.

Table 4. The performance of the learning algorithms after preprocessing—Krack attack.

Algorithm	Accuracy	True Positive Rate	False Negative Rate
Decision tree	95.1%	90%	10%
Ensemble classifier	50.6%	50.6%	49.4%
KNN	73.8%	48.3%	51.7%
SVM	50.6%	50.6%	49.4%

For the same previous sample, we have used feature selection techniques to reduce the computing time and enhance the accuracy of the detection model. We chose the ANOVA FS technique, as shown in Figure 3, which is a widely used statistical approach for comparing different independent means.

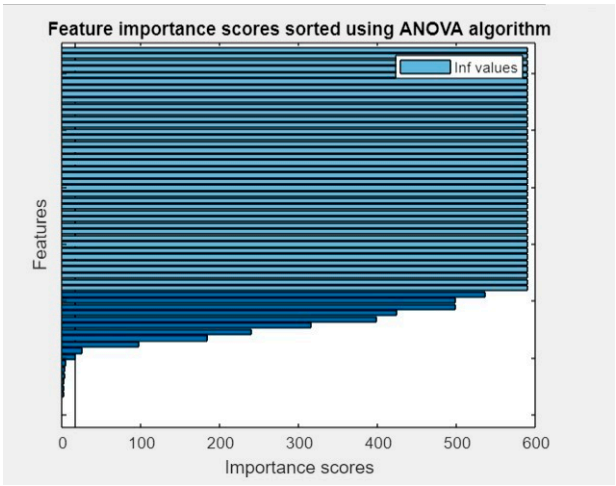

Figure 3. Feature importance scores stored using the ANOVA algorithm.

The features are ranked in the ANOVA method by calculating the variance ratio between and within groups [49]. The accuracy results after applying ANOVA feature selection (FS = 15) are shown in Table 5.

Table 5. The performance of the learning algorithms after feature selection—Krack attack.

Algorithm	Accuracy	True Positive Rate	False Negative Rate
Decision tree	93.2%	86.6%	13.4%
Naive Bayes	95%	97.7%	2.3%
Ensemble classifier	99.1%	98.2%	1.8%
KNN	68.2%	35.8%	64.2%
SVM	73.8%	46.9%	53.1%

The results proved the necessity of processing the dataset since an efficient result depends on efficient data, and furthermore, the results showed improvement in accuracy results when we used feature selection techniques, as Figure 4 shows.

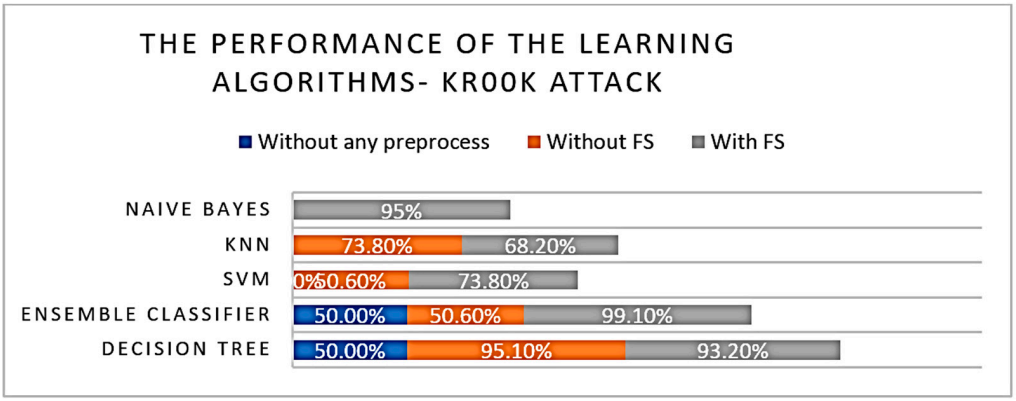

Figure 4. The performance of the learning algorithms—Krack attack.

The best accuracy results that we obtained were from the ensemble classifier with 99.1% in addition to a 1.8% false negative rate, followed by naive Bayes with a 95% accuracy result and 2.3% false negative rate.

3.2.2. Detecting Kr00k Attack

This sample consists of 235,064 instances; 106,971 kr00k traffic and 128,093 normal ones, and the preprocessing procedure for the Krack dataset sample was as follows:

1. Deleting the constant and empty features.
2. Ignoring features with more than 60% missing values, the remaining features are 63.
3. Replace missing values with NaN.

The remaining dataset consists of 63 features and 235,064 instances.

After preprocessing the data, we applied different ML algorithms. Table 6 shows the performance of the learning algorithms after preprocessing.

Table 6. The performance of the learning algorithms after preprocessing—Kr00k attack.

Algorithm	Accuracy	True Positive Rate	False Negative Rate
Decision tree	81.8%	60%	40%
Ensemble classifier	70%	45.3%	54.7%
KNN	53.3%	53.3%	46.7%
SVM	44.8%	40.7%	59.3%

For the same previous sample, we have used ANOVA feature selection techniques to reduce the computing time and enhance the accuracy of the detection model as shown in Figure 5, which is a widely used statistical approach for comparing different independent means.

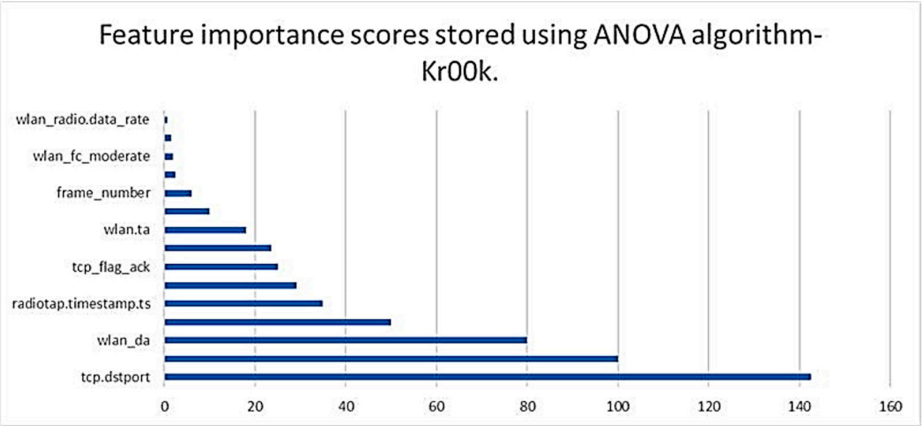

Figure 5. Feature importance scores stored using ANOVA algorithm—Kr00k.

The accuracy results after applying ANOVA feature selection (FS = 15), are shown in Table 7. After applying the ML algorithms on the chosen sample three times—without any process for the dataset, with preprocessing, and with FS—we can realize that the preprocessing step is a critical and essential step in data mining to obtain accurate results, especially in dealing with data that suffer from high dimensionality imbalance and overfitting of the data. The accuracy results for the mentioned steps are presented in Figure 6. The best accuracy that we obtained was for neural network and SVM with 96.7%. We can conclude from Figure 6 how the accuracy is affected by FS and prepossessing the dataset before applying any ML algorithms to it.

Table 7. The performance of the learning algorithms after feature selection—Kr00k attack.

Algorithm	Accuracy
Decision tree	93.3%
Ensemble classifier	83.3%
KNN	86.7%
SVM	96.67%
Neural Network	96.7%
Kernel	83.3%

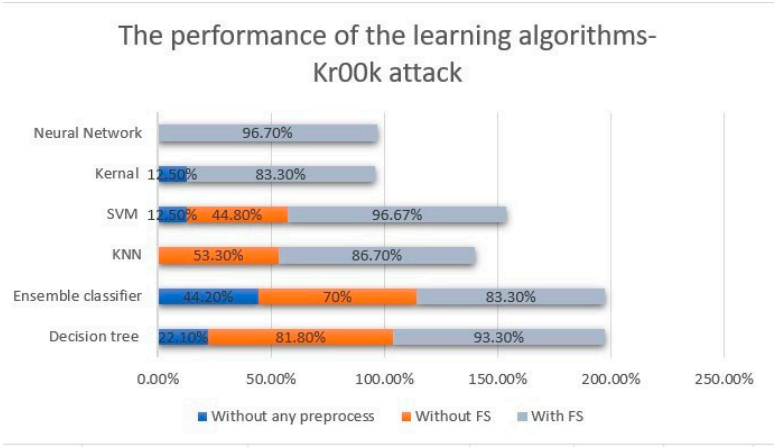

Figure 6. The performance of the learning algorithms—Kr00k attack.

3.2.3. Multiclass Detection

In this phase, we will use a sample consisting of three classes (Krack, Kr00k, and Nominal), 15,000 instances, and 254 features. Due to the importance of preprocessing as we noted in the previous subsections, we have applied preprocessing steps in the chosen sample, removing the empty features and the features with constant values, in addition to replacing all the empty cells in the remaining features with NaN. Then, we applied ML algorithms using the classification linear application on MATLAB. The performance of the ML algorithms is presented in Table 8. The table presents the accuracy results using FS with NOVA algorithm techniques and without using FS.

Table 8. The performance of the learning algorithms—multi-class.

Algorithm	Accuracy without FS	Accuracy FS
Decision tree	57.1%	88.3%
Ensemble classifier	44.7%	90.7%
KNN	67.4%	73.3%
SVM	57.1%	69.5%
Kernel	53.1%	60.4%

The accuracy results for the mentioned steps are presented in Figure 7.

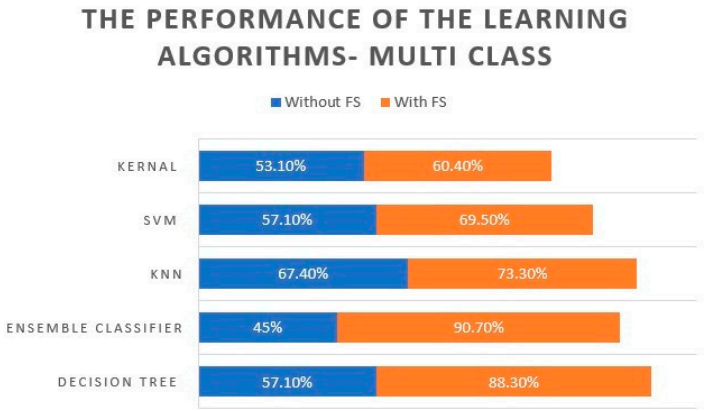

Figure 7. The performance of the learning algorithms—multiclass.

The best accuracy that we achieved without using FS was for KNN 67.4%, while when we applied the ANOVA FS, the performance increased in all the used algorithms. The best accuracy was achieved by the ensemble classifier with 90.7% and for the decision tree with 88.3%.

4. Results and Discussion

We implemented our experiments in three phases; for the first phase, we used a sample of AWID3 dataset that includes a nominal two classes (Krack and Normal); the best accuracy that we achieved was for the decision tree with 95.1% without using FS, while the best accuracy after using ANOVA FS was 99.1% for the ensemble classifier. For the second phase, we used a sample with two nominal classes too (Kr00k and Normal); the best accuracy that we achieved was for the decision tree as well, with 81.8% without using FS. The accuracy increased to 96.7% for the neural network after applying ANOVA FS techniques. In the last phase, we used a multi-classes classifier, where the label includes three classes (Krack, Kr00k, and Nominal), and the accuracy results were determined to be low. However, after applying the FS techniques, the accuracy increased to 90.7% and 88.3% for the ensemble classifier and decision tree, respectively. Figure 8 summarizes the results for the three mentioned phases.

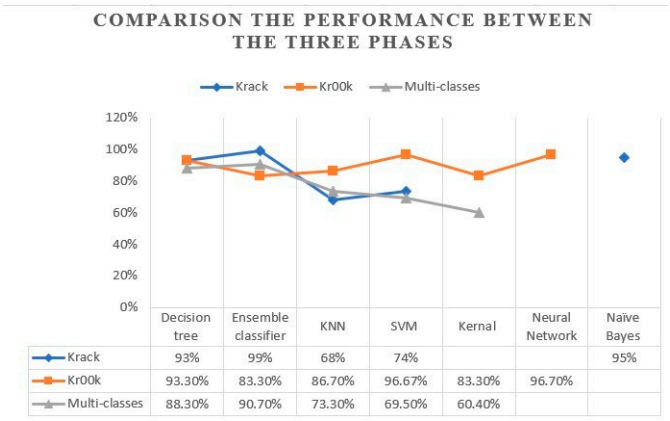

Figure 8. Comparison of the performance between the three phases.

When looking at the results, it becomes clear that the decision tree and ensemble classifier have a high performance in three experiments.

It is worth mentioning that the wireless dataset including the AWID3 dataset has many challenges, such as:

- High dimensionality: This refers to a high number of features in the dataset. So, it is important to transform the data from a high-dimensional space into a low-dimensional

space so that the low-dimensional representation retains some meaningful properties of the original data, ideally close to its intrinsic dimension [50]. We solved this problem using the ANOVA FS techniques, where the performance of the algorithms shows clear improvement in accuracy results when we reduce the high dimensionality for the dataset.

- Overfitting of the dataset: This occurs when a statistical model fits exactly against its training data [51]. When performing the ML model to the data without solving the overfitting problem, the accuracy results will be almost 100% or 99.99% which is not a reliable performance. We solved this problem in the preprocessing step by getting rid of the features that copy the label, in addition to the importance of FS in solving this problem.
- Unbalanced data: Unbalanced refers to a classification data set with skewed class proportions. We solved this problem by taking almost the same number of instances for attack and benign in the three experiments.

Comparing Our Findings with Previous Studies

The authors in [38] used a state-machine architecture to find Krack attacks by monitoring numerous wireless channels. To specifically identify the Krack symptoms at various points of a handshake session, they undertook deep packet inspection and created a grouping method to group Wi-Fi handshake packets. They used supervised machine learning models based on gradient boosting, and their accuracy was around 93.39% with a false positive rate of 5.08%.

In [52], the authors proposed a framework for unsupervised classification and data mining of tweets about cyber vulnerabilities; this vulnerability included the Kr00K attack, which allows unauthorized decryption in Wi-Fi chips. The best accuracy that they achieved was 88.52%

Chatzoglou et al. applied deep learning and machine learning techniques on the AWID3 benchmark dataset [53], in order to answer questions about the competence of 802.11-specific and non-802.11 features when used separately and in tandem in detecting application layer attacks and to know which network protocol features are the most informative to the machine learning model for detecting application layer attacks; the performance of the detection model achieved 96.7% accuracy.

Due to the increased urgency for unrestricted network data access to improve cyber-AI efficiency in unfamiliar threat scenarios, the authors in [54] proposed an automated network scanning and data-mining technique through open-source service discovery tools for deep reinforcement learning-based cognitive network intrusion detection systems. They obtained the lowest false alarm rate and a 98.68% accuracy.

In [55], the authors investigated how to map machine learning algorithms to programmable network devices. Furthermore, state-of-the-art and newly proposed in-network ML algorithms are evaluated and compared in terms of functionality, resources, scalability, and throughput. They used six datasets, including KDD99 and AWID3, for intrusion detection purposes. Their accuracy ranged from 97.47% for decision trees to 49.37% for KNN.

Table 9 summarizes the above-mentioned studies.

Table 9. Comparison of our findings with other studies.

Reference	Year	Description	Accuracy
[38]	2022	They monitored numerous wireless channels to detect Krack attacks	93.0%
[52]	2021	Proposed a framework for cyber vulnerabilities, including Kr00k	88.52%
[53]	2022	Proposed a model to detect application layer attacks	96.7%
[54]	2021	Proposed an automated network scanning and data-mining technique for Network IDS	98.68%
Our work		We focused on the IEEE 802.11 vulnerabilities, by proposing a model to detect Krack and Kr00k attacks using ML techniques.	Phase 1: 99% Phase 2: 96.7% phase 3: 90.7%

5. Conclusions

Worldwide internet usage has significantly increased because of the extensive adoption of Wi-Fi connectivity, but there has also been a commensurate growth in cybercrimes that target the weaknesses of wireless systems. In this research, we concentrated on the recently discovered attacks known as Krack and Kr00k that were found in the IEEE 802.11 standard. The protocols for constructing wireless local area networks, including the media access control (MAC) and physical layer protocols, are specified in this standard, which is a part of the larger IEEE 802 collection of local area network technical standards.

We developed an intrusion detection system (IDS) model utilizing MATLAB's classification linear program to address the problems of these attacks. The AWID3 dataset, which is regarded as one of the most recent and well-liked wireless datasets accessible, was used to thoroughly test our suggested model. With the help of this research, we were able to successfully handle problems like excessive dimensionality and data imbalance that are frequently found in wireless datasets.

We detected these assaults with remarkable accuracy using our ML-based methodology. For example, our ensemble classifier showed an astounding 99% accuracy in recognizing Krack attacks, while our neural network classifier had the best accuracy in detecting Kr00K attacks at 96.7%.

This study highlights how important it is to comprehend the distinctive characteristics of wireless datasets and how they affect the effectiveness of detection models. We intend to address the issues posed by wireless datasets in the next work to improve wireless IDS performance even more. We work to contribute to the creation of stronger and more efficient wireless intrusion detection systems by constantly enhancing and modifying our methods.

Author Contributions: Conceptualization, Z.S. and E.A.E.; Formal analysis, Z.S. and E.A.E.; Investigation, Z.S. and E.A.E.; Methodology, Z.S. and E.A.E.; Software, Z.S. and E.A.E.; Supervision, Z.S.; Validation, Z.S. and E.A.E.; Data curation, Z.S. and E.A.E.; Writing—original draft, Z.S. and E.A.E.; Writing—review and editing, Z.S. and E.A.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available in a publicly accessible repository. The data presented in this study are openly available [https://icsdweb.aegean.gr/awid/awid3], accessed on 12 October 2022.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Alraih, S.; Shayea, I.; Behjati, M.; Nordin, R.; Abdullah, N.F.; Abu-Samah, A.; Nandi, D. Revolution or Evolution? Technical Requirements and Considerations towards 6G Mobile Communications. *Sensors* **2022**, *22*, 762. [CrossRef]

2. Chettri, L.; Bera, R. A Comprehensive Survey on Internet of Things (IoT) Toward 5G Wireless Systems. *IEEE Internet Things J.* **2019**, *7*, 16–32. [CrossRef]
3. Ahn, V.T.H.; Ma, M. A Secure Authentication Protocol with Performance Enhancements for 4G LTE/LTE-A Wireless Networks. In Proceedings of the 2021 3rd International Electronics Communication Conference (IECC), Ho Chi Minh City, Vietnam, 8–10 July 2021; pp. 28–36.
4. Prabha, P.A.; Arjun, N.; Gogul, J.; Prasanth, S.D. Two-Way Economical Smart Device Control and Power Consumption Prediction System. In Proceedings of the International Conference on Recent Trends in Computing, Ghaziabad, India, 4–5 June 2021; Springer: Singapore, 2022; pp. 415–429.
5. Liyanage, M.; Braeken, A.; Jurcut, A.D.; Ylianttila, M.; Gurtov, A. Secure communication channel architecture for Software Defined Mobile Networks. *Comput. Netw.* **2017**, *114*, 32–50. [CrossRef]
6. Gurtov, A.; Liyanage, M.; Ylianttila, M. *Software Defined Mobile Networks (SDMN): Beyond LTE Network Architecture*; John Wiley & Sons: Hoboken, NJ, USA, 2015.
7. Park, J.H.; Rathore, S.; Singh, S.K.; Salim, M.M.; Azzaoui, A.; Kim, T.W.; Pan, Y.; Park, J.H. A comprehensive survey on core technologies and services for 5g security: Taxonomies, issues, and solutions. *Hum.-Centric Comput. Inf. Sci.* **2021**, *11*, 3.
8. Gupta, S.; Parne, B.L.; Chaudhari, N.S. Security vulnerabilities in handover authentication mechanism of 5g network. In Proceedings of the 2018 First International Conference on Secure Cyber Computing and Communication (ICSCCC), Jalandhar, India, 15–17 December 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 369–374.
9. Borgaonkar, R.; Tøndel, I.A.; Degefa, M.Z.; Jaatun, M.G. Improving smart grid security through 5G enabled IoT and edge computing. *Concurr. Comput. Pract. Exp.* **2021**, *33*, e6466. [CrossRef]
10. Gonzalez, A.J.; Grønsund, P.; Dimitriadis, A.; Reshytnik, D. Information security in a 5g facility: An implementation experience. In Proceedings of the 2021 Joint European Conference on Networks and Communications & 6G Summit (EuCNC/6G Summit), Porto, Portugal, 8–11 June 2021; IEEE: Piscataway, NJ, USA, 2021; pp. 425–430.
11. Kim, H. 5G core network security issues and attack classification from network protocol perspective. *J. Internet Serv. Inf. Secur.* **2020**, *10*, 1–15.
12. Mohan, J.P.; Sugunaraj, N.; Ranganathan, P. Cyber security threats for 5g networks. In Proceedings of the 2022 IEEE International Conference on Electro Information Technology (eIT), Mankato, MN, USA, 19–22 May 2022; IEEE: Piscataway, NJ, USA, 2022; pp. 446–454.
13. Tsiknas, K.; Taketzis, D.; Demertzis, K.; Skianis, C. Cyber Threats to Industrial IoT: A Survey on Attacks and Countermeasures. *IoT* **2021**, *2*, 163–186. [CrossRef]
14. Muthuramalingam, S.; Thangavel, M.; Sridhar, S. A review on digital sphere threats and vulnerabilities. In *Combating Security Breaches and Criminal Activity in the Digital Sphere*; IGI Global: Hershey, PA, USA, 2016; pp. 1–21.
15. Klaine, P.V.; Imran, M.A.; Onireti, O.; Souza, R.D. A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks. *IEEE Commun. Surv. Tutor.* **2017**, *19*, 2392–2431. [CrossRef]
16. Klautau, A.; Batista, P.; Gonza, N.; Wang, Y.; Heath, R.W. 5G mimo data for machine learning: Application to beam-selection using deep learning. In Proceedings of the 2018 Information Theory and Applications Workshop (ITA), San Diego, CA, USA, 11–16 February 2018; IEEE: Piscataway, NJ, USA, 2018; pp. 1–9.
17. Kafle, V.P.; Fukushima, Y.; Martinez-Julia, P.; Miyazawa, T. Consideration on Automation of 5G Network Slicing with Machine Learning. In Proceedings of the 2018 ITU Kaleidoscope: Machine Learning for a 5G Future, Santa Fe, Argentina, 26–28 November 2018.
18. Sofi, I.B.; Gupta, A. A survey on energy efficient 5G green network with a planned multi-tier architecture. *J. Netw. Comput. Appl.* **2018**, *118*, 1–28. [CrossRef]
19. Ioannou, I.; Christophorou, C.; Vassiliou, V.; Pitsillides, A. A distributed AI/ML framework for D2D Transmission Mode Selection in 5G and beyond. *Comput. Netw.* **2022**, *210*, 108964. [CrossRef]
20. Nassef, O.; Sun, W.; Purmehdi, H.; Tatipamula, M.; Mahmoodi, T. A survey: Distributed Machine Learning for 5G and beyond. *Comput. Netw.* **2022**, *207*, 108820. [CrossRef]
21. Babu, K.V.; Das, S.; Sree, G.N.J.; Patel, S.K.; Saradhi, M.P.; Tagore, M. Design and development of miniaturized MIMO antenna using parasitic elements and Machine learning (Ml) technique for lower sub 6 GHz 5G applications. *AEU-Int. J. Electron. Commun.* **2022**, *153*, 154281. [CrossRef]
22. Yang, L.; Li, J.; Yin, L.; Sun, Z.; Zhao, Y.; Li, Z. Real-Time Intrusion Detection in Wireless Network: A Deep Learning-Based Intelligent Mechanism. *IEEE Access* **2020**, *8*, 170128–170139. [CrossRef]
23. Berisha, V.; Krantsevich, C.; Hahn, P.R.; Hahn, S.; Dasarathy, G.; Turaga, P.; Liss, J. Digital medicine and the curse of dimensionality. *NPJ Digit. Med.* **2021**, *4*, 153. [CrossRef]
24. Lee, S.J.; Yoo, P.D.; Asyhari, A.T.; Jhi, Y.; Chermak, L.; Yeun, C.Y.; Taha, K. IMPACT: Impersonation attack detection via edge computing using deep autoencoder and feature abstraction. *IEEE Access* **2020**, *8*, 65520–65529.
25. Kolias, C.; Kambourakis, G.; Stavrou, A.; Gritzalis, S. Intrusion Detection in 802.11 Networks: Empirical Evaluation of Threats and a Public Dataset. *IEEE Commun. Surv. Tutor.* **2015**, *18*, 184–208. [CrossRef]
26. Chatzoglou, E.; Kambourakis, G.; Kolias, C. Empirical evaluation of attacks against IEEE 802.11 enterprise networks: The awid3 dataset. *IEEE Access* **2021**, *9*, 34188–34205. [CrossRef]

27. Kolias, C.; Kolias, V.; Kambourakis, G. TermID: A distributed swarm intelligence-based approach for wireless intrusion detection. *Int. J. Inf. Secur.* **2017**, *16*, 401–416. [CrossRef]
28. Aminanto, M.E.; Choi, R.; Tanuwidjaja, H.C.; Yoo, P.D.; Kim, K. Deep Abstraction and Weighted Feature Selection for Wi-Fi Impersonation Detection. *IEEE Trans. Inf. Forensics Secur.* **2017**, *13*, 621–636. [CrossRef]
29. Diro, A.; Chilamkurti, N. Leveraging LSTM Networks for Attack Detection in Fog-to-Things Communications. *IEEE Commun. Mag.* **2018**, *56*, 124–130. [CrossRef]
30. Sethuraman, S.C.; Dhamodaran, S.; Vijayakumar, V. Intrusion detection system for detecting wireless attacks in IEEE 802.11 networks. *IET Netw.* **2019**, *8*, 219–232. [CrossRef]
31. Kasongo, S.M.; Sun, Y. A deep learning method with wrapper based feature extraction for wireless intrusion detection system. *Comput. Secur.* **2020**, *92*, 101752. [CrossRef]
32. Zhou, Y.; Cheng, G.; Jiang, S.; Dai, M. Building an efficient intrusion detection system based on feature selection and ensemble classifier. *Comput. Netw.* **2020**, *174*, 107247. [CrossRef]
33. Hacılar, H.; Aydın, Z.; Güngör, V.Ç. Intrusion Detection with Bayesian Optimization on Imbalance Wired Wireless and Software-Defined Networking Traffics. Available online: https://www.researchgate.net/publication/357833330_Intrusion_Detection_with_Bayesian_Optimization_on_Imbalance_Wired_Wireless_and_Software-Defined_Networking_Traffics (accessed on 1 July 2023).
34. Wilson, R.; Linekar, R. Towards Effective Wireless Intrusion Detection using AWID Dataset. Thesis, Rochester Institute of Technology, Rochester, NY, USA, 2021. Available online: https://scholarworks.rit.edu/theses/10700 (accessed on 1 July 2023).
35. Bhandari, S.; Kukreja, A.K.; Lazar, A.; Sim, A.; Wu, K. Feature selection improves tree-based classification for wireless intrusion detection. In Proceedings of the 3rd International Workshop on Systems and Network Telemetry and Analytics, Stockholm, Sweden, 23 June 2020; pp. 19–26.
36. Rahman, M.A.; Asyhari, A.T.; Leong, L.; Satrya, G.; Tao, M.H.; Zolkipli, M. Scalable machine learning-based intrusion detection system for iot-enabled smart cities. *Sustain. Cities Soc.* **2020**, *61*, 102324. [CrossRef]
37. Čermák, M.; Svorenčík, S.; Lipovský, R. Kr00k-cve-2019-15126–Serious Vulnerability Deep Inside Your Wi-Fi Encryption; ESET Research White Paper, Bratislava, Slovak Republic; 2020. Available online: https://web-assets.esetstatic.com/wls/2020/02/ESET_Kr00k.pdf (accessed on 1 July 2023).
38. Agrawal, A.; Chatterjee, U.; Maiti, R.R. Ktracker: Passively tracking krack using ml model. In Proceedings of the Twelveth ACM Conference on Data and Application Security and Privacy, Baltimore, MD, USA, 24–27 April 2022; pp. 364–366.
39. Fontes, R.D.R.; Rothenberg, C.E. On the Krack Attack: Reproducing Vulnerability and a Software-Defined Mitigation Approach, (2018). Available online: https://api.semanticscholar.org/CorpusID:51995777 (accessed on 1 July 2023).
40. Vanhoef, M.; Piessens, F. Key reinstallation attacks: Forcing nonce reuse in wpa2. In Proceedings of the 2017 ACM SIGSAC Conference on Computer and Communications Security, Dallas, TX, USA, 30 October–3 November 2017; pp. 1313–1328.
41. Kohlios, C.P.; Hayajneh, T. A Comprehensive Attack Flow Model and Security Analysis for Wi-Fi and WPA3. *Electronics* **2018**, *7*, 284. [CrossRef]
42. Hastie, T.; Tibshirani, R.; Friedman, J.H.; Friedman, J.H. *The Elements of Statistical Learning: Data Mining, Inference, and Prediction*; Springer: Berlin/Heidelberg, Germany, 2009; Volume 2.
43. Cortesc, V. Support vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
44. Shawe-Taylor, J.; Cristianini, N. *Kernel Methods for Pattern Analysis*; Cambridge University Press: Cambridge, UK, 2004.
45. Goodfellow, I.; Bengio, Y.; Courville, A. *Deep Learning*; MIT Press: Cambridge, MA, USA, 2016.
46. Alam, S.; Yao, N. The impact of preprocessing steps on the accuracy of machine learning algorithms in sentiment analysis. *Comput. Math. Organ. Theory* **2019**, *25*, 319–335. [CrossRef]
47. Go, A.; Bhayani, R.; Huang, L. *Twitter Sentiment Classification Using Distant Supervision*; CS224N Project Report; Stanford University: Stanford, CA, USA, 2009; Volume 1.
48. Kubik, C.; Knauer, S.M.; Groche, P. Smart sheet metal forming: Importance of data acquisition, preprocessing and transformation on the performance of a multiclass support vector machine for predicting wear states during blanking. *J. Intell. Manuf.* **2022**, *33*, 259–282. [CrossRef]
49. Nasiri, H.; Alavi, S.A. A Novel Framework Based on Deep Learning and ANOVA Feature Selection Method for Diagnosis of COVID-19 Cases from Chest X-Ray Images. *Comput. Intell. Neurosci.* **2022**, *2022*, 4694567. [CrossRef] [PubMed]
50. Zebari, R.; Abdulazeez, A.; Zeebaree, D.; Zebari, D.; Saeed, J. A Comprehensive Review of Dimensionality Reduction Techniques for Feature Selection and Feature Extraction. *J. Appl. Sci. Technol. Trends* **2020**, *1*, 56–70. [CrossRef]
51. Belkin, M.; Hsu, D.; Ma, S.; Mandal, S. Reconciling modern machine-learning practice and the classical bias–variance trade-off. *Proc. Natl. Acad. Sci. USA* **2019**, *116*, 15849–15854. [CrossRef] [PubMed]
52. Alperin, K.; Joback, E.; Shing, L.; Elkin, G. A framework for unsupervised classificiation and data mining of tweets about cyber vulnerabilities. *arXiv* **2021**, arXiv:2104.11695.
53. Chatzoglou, E.; Kambourakis, G.; Smiliotopoulos, C.; Kolias, C. Best of both worlds: Detecting application layer attacks through 802.11 and non-802.11 features. *Sensors* **2022**, *22*, 5633. [CrossRef] [PubMed]
54. Muhati, E.; Rawat, D.B. Asynchronous Advantage Actor-Critic (A3C) Learning for Cognitive Network Security. In Proceedings of the 2021 Third IEEE International Conference on Trust, Privacy and Security in Intelligent Systems and Applications (TPS-ISA), Virtual, 13–15 December 2021; pp. 106–113.

55. Zheng, C.; Zang, M.; Hong, X.; Bensoussane, R.; Vargaftik, S.; Ben-Itzhak, Y.; Zilberman, N. Automating in-network machine learning. *arXiv* **2022**, arXiv:2205.08824.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.