Computer Communications 192 (2022) 316–331

Contents lists available at ScienceDirect

Computer Communications

journal homepage: www.elsevier.com/locate/comcom

# Machine learning security attacks and defense approaches for emerging cyber physical applications: A comprehensive survey

Jaskaran Singh [a], Mohammad Wazid [a], Ashok Kumar Das [b,c,*], Vinay Chamola [d], Mohsen Guizani [e,1]

[a] Department of Computer Science and Engineering, Graphic Era Deemed to be University, Dehradun 248002, India
[b] Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India
[c] Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA 23435, USA
[d] Department of Electrical and Electronics Engineering & Anuradha and Prashanth Palakurthi Centre for Artificial Intelligence Research (APPCAIR), BITS-Pilani, Pilani Campus, 333 031, India
[e] Machine Learning Department, Mohamed bin Zayed University of Artificial Intelligence (MBZUAI), 51133 Abu Dhabi, UAE

ARTICLE INFO

*Keywords:*
Cyber physical systems (CPS)
Machine learning (ML) security
Intrusion detection
Authentication
Privacy and security

ABSTRACT

The cyber physical systems integrate the sensing, computation, control and networking processes into physical objects and infrastructure, which are connected through the Internet to execute a common task. Cyber physical systems can be applied in various applications, like healthcare, transportation, industrial production, environment and sustainability, and security and surveillance. However, the tight coupling of cyber systems with physical systems introduce challenges in addressing stability, security, efficiency and reliability. The machine learning (ML) security is the inclusion of cyber security mechanism to provide protection to the machine learning models against various cyber attacks. The ML models work through the traditional training and testing approaches. However, execution of such kind of approaches may not function effectively in case if a system is connected to the Internet. As online hackers can exploit deployed security mechanisms and poison the data. This data is then taken as the input by the ML models. In this article, we provide the details of various machine learning security attacks in cyber physical systems. We then discuss some defense mechanisms to protect against these attacks. We also present a threat model of ML security mechanisms deployed in cyber systems. Furthermore, we discuss various issues and challenges of ML security mechanisms deployed in cyber systems. Finally, we provide a detailed comparative study on performance of the ML models under the influence of various ML attacks in cyber physical systems.

## 1. Introduction

Cyber physical systems (CPS) are systems that collaborate computational entities (i.e., sensors and actuators) in connection with the physical world and the associated processes. This further facilitates the data-accessing and data-processing services available on the Internet. CPS can be used in various types of applications (i.e., smart home, smart healthcare and transportation systems, etc.) [1,2]. Though CPS can be deployed in various applications at the same time, they have many Challenges related to security and privacy because various attacks (i.e., malware injection, impersonation, man-in-the-middle (MiTM), leakage of secret data, unauthorized data updates, etc.) are possible [3,4]. Sometimes we use the machine learning (ML) models in the CPS-based applications to draw useful outcomes from the collected data of the sensors. Therefore, the role of ML models is very important here, and their predictions and outcomes should be accurate. However, the existence of various attacks may affect the performance of the ML models, and thus, they may produce wrong outcomes [5,6]. ML is a field of computing that utilizes computational algorithms to train machines to be able to perform various tasks that further automate the workload without explicitly intervening at each step and limiting human interaction with the process [7]. ML has its applications in various domains (for example, medicine, agriculture, and natural disaster prediction and management) [8–10]. Moreover, it can be integrated and utilized in various domains, like the Internet of Things (IoT), cyber physical systems, cyber security, computer vision, image processing, robotics, natural language processing, and many more.

Since the presented work is related to ML security, therefore it is essential to explain the phases of ML tasks. Fig. 1 depicts various phases

* Corresponding author at: Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad 500 032, India.
*E-mail addresses:* jaskaran.jsk2001@gmail.com (J. Singh), wazidkec2005@gmail.com (M. Wazid), ashok.das@iiit.ac.in (A.K. Das), vinay.chamola@pilani.bits-pilani.ac.in (V. Chamola), mguizani@ieee.org (M. Guizani).
[1] *Fellow, IEEE.*

https://doi.org/10.1016/j.comcom.2022.06.012
Received 7 November 2021; Received in revised form 24 March 2022; Accepted 12 June 2022
Available online 18 June 2022
0140-3664/© 2022 Elsevier B.V. All rights reserved.

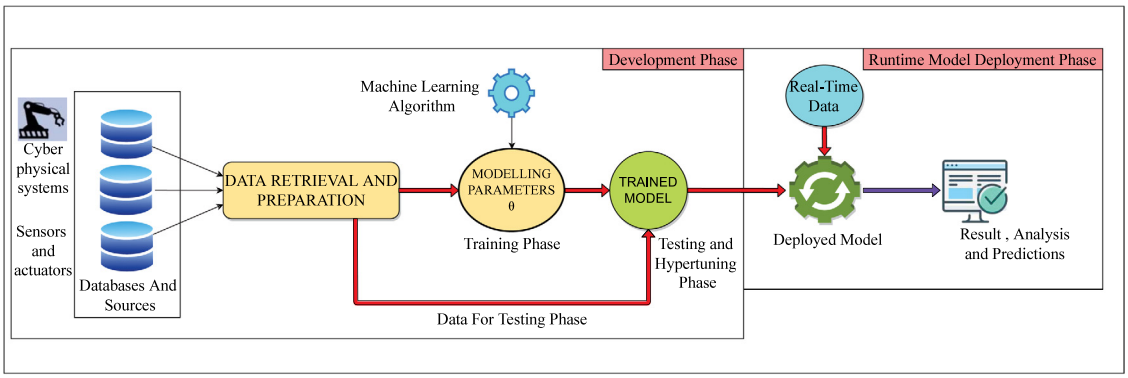


**Fig. 1.** Various phases of machine learning tasks in the cyber physical systems.

of ML tasks in the cyber physical systems. A basic ML task can be divided into two phases: (1) training phase and (2) deployment phase (testing phase) [11]. Their details are given below.

- **Training phase**: The task starts with accumulation of data from reputable and authorized sources (i.e., sensors). The often humongous data is analyzed and prepared for training. It utilizes different techniques, like cleaning, augmenting, and segmenting. It basically involves converting irregularities and missing values in the dataset into consistent data that can be processed further. Next, the dataset and problem statement is analyzed. The class labels and features of the dataset are understood, and correlation (degree of relationship) between different features is visualized. After that, the data is split into two parts for training and testing purposes of the model keeping in mind the correlation between features and the target prediction required the selection of suitable ML algorithm [12,13]. The algorithm is the intuition behind the model, and it provides the prediction output of the input data based on the value of the feature through a mathematical formula. The training data is given to the ML model to train it with the feature values, and then the possible pattern is made. This pattern is calculated, and the parameters for prediction are calculated. Now, the testing phase starts, and the calculated parameters are used on the test data in order to carry out the new predictions. The accuracy score of the ML algorithm is calculated using the test data to find the prediction capability of the algorithm. Hypertuning is performed on the best algorithm by tweaking its formula to get the best iteration of the ML algorithm. Once this hyper tuning is performed and the model gives a satisfactory accurate prediction, the trained model finally deployed [11,14].
- **Deployment phase:** The deployed model after hyper tuning is supplied with the real-time data. The trained model will provide prediction output on input new data. The model may use the Application Programming Interface (API) to interact with the users where we can feed the data through it and obtain the predictions based on training done under the training phase [15]. The results of predictions and findings are then summarized and presented in the illustrated manner for future analysis and decision-making purpose [11,16].

Information security is the methodology of protecting information and sensitive data from security risks (i.e., unauthorized access and usage, modification, inspection, and deletion of the information). Information security in the cyber physical systems is provided on the basis of CIA Triad, which comprises techniques like confidentiality, integrity, and availability [17]. Confidentiality (or privacy) involves restricting access to the information. Its usage is much needed in order to protect information from being accessed or modified by malicious entities. This can be achieved by utilizing encryption techniques, including public-key cryptography and security tokens. Moreover, integrity (or data integrity) involves maintaining the trustworthiness and dependability of the information. It is practiced to retain the usability of data and prevail it to be usable for other tasks. It could be achieved by using the techniques like version and access controlling, hashing and compliance checks, and keeping data checksums. Furthermore, availability is the practice of accessibility of information for retrieval and usage by authorized entities. It is required to maintain the information consistently through the maintaining systems which hold them. It could be achieved through server monitoring, redundancy, resolving software issues, and maintaining contingency protocols to deal with Denial-of-Service (DoS) or distributed DoS (DDoS) attacks [18]. These are the basic characteristics that we cover under information security. However, we need to take care of other important properties like authentication, access control, authorization, forward secrecy, backward secrecy, data freshness, etc., [19].

In the following, we provide the details of machine learning (ML) security in cyber physical systems. ML security is the inclusion of cyber security techniques to safeguard the integrity and privacy of an ML model from cyber attacks [20]. It utilizes the various defense mechanisms to prevent the subjection of the model from attacks that further prevent sensitive information from getting breached. It also stops any disaster-related to the prediction of wrong outcomes. With the vast extent of ML being used and fueling software that affects lives of billions, these days protecting our vital data, and for smooth deployment of services that directly or indirectly utilize ML security is becoming an important field of study [21]. Protection against malicious activities is a vital aspect of the ML task [14,22]. Securing our ML process is very important. For instance, when we work with sensitive data, the correct training of data is essentially required. With the substantial growth of technologies and development in Big data, securing all such types of data and protecting ML tasks are the ever occurring tasks, which need to be resolved with utmost priority [16].

### 1.1. Motivation

ML security operates with the help of various cyber security mechanisms, which are deployed there to safeguard the integrity and privacy of an ML model against the various threats and attacks [23]. It uses different safeguarding schemes to prevent the subjection of the model from attacks. This prevents sensitive information from getting breached and also prevents the system from producing bad outcomes (predictions) [11,22]. The primary motivation behind the survey article is to summarize the research work and case studies done in the field of ML security in the cyber physical systems [2]. Such a communication environment is being used in various domains (i.e., healthcare, security and surveillance, retailing, industrial automation, control and support, intelligent transportation system, etc.,) [3,4]. The correct prediction and privacy of users' data are essentially required. In machine learning, we use a model, which is called as ML model, and is used for the purpose of prediction of some phenomena. During the literature survey,

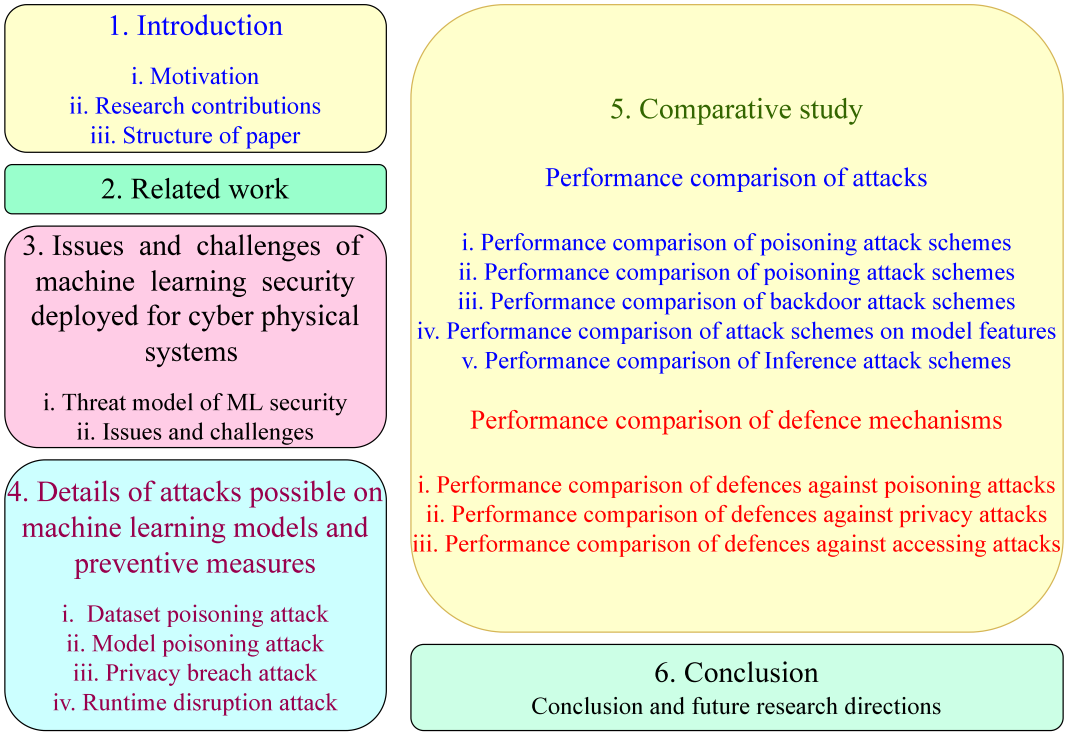


**Fig. 2.** Roadmap of the paper.

it has been identified that the ML models are vulnerable to various types of attacks (i.e., dataset poisoning attack, model poisoning attack, privacy breach, runtime disruption attack, and membership inference attacks). Due to the enormous use of ML, it becomes essential to protect its models against the various possible attacks [16]. Therefore, we focus on various attacks that are associated with the ML models. The different mechanisms of these attacks have also been discussed, along with some possible solutions to prevent them. This research work will be helpful for the researchers to make machine learning more secure and robust.

### *1.2. Research contributions*

The research contributions of this work are summarized below.

- We present a threat model of ML security in the cyber physical systems, in which we provide the details of all threats associated with the ML models.
- We then discuss various issues and challenges of ML security in the cyber physical systems.
- Next, we discuss the mechanisms of various attacks related to the ML security.
- We also discuss some possible solutions that can be used to protect the ML security.
- Furthermore, we provide a comparative study on performance of the ML models under the influence of various attacks that can be also deployed for the cyber physical systems.

### *1.3. Paper outline*

The remaining part of the paper is organized as follows. The existing surveys and their limitations are stated in Section 2. A summary of various research works conducted in the field of ML security is also given in Section 2. Various issues and challenges of ML security deployed for the cyber physical systems are given in Section 3. The associated threat model of ML security deployed for the cyber physical systems is also discussed in Section 3. The details of various attacks that are possible on ML models and their preventive measures are given in Section 4. The details of comparative study on performance of the ML models under the influence of various ML attacks are then given in Section 5. Section 6 provides some concluding remarks and future research directions. In addition, Fig. 2 also represents the overall outline of the article.

In this section, we discussed machine learning, information security, and machine learning security. Then we discussed the motivation behind the presented work. We have also highlighted the research contributions of the paper. In the next section, we will discuss the existing research works related to ML security.

## 2. Related work

The details of some existing surveys related to the ML attacks and their defense approaches are discussed below.

Chen et al. [24] provided a survey on adversarial attacks in reinforcement learning under the Artificial Intelligence (AI) security. Further, they provided a brief introduction on the most representative defense technologies against existing adversarial attacks. However, they did not provide the details of potential attacks (i.e., various privacy breaches). Apart from that a comparative study on existing solutions for potential ML attacks was not provided.

Berman et al. [25] provided a literature review of deep learning (DL) methods, which could be used for cyber security. They provided a discussion on various DL methods, like deep autoencoders, restricted Boltzmann machines, recurrent neural networks and generative adversarial networks. Next, they discussed about the deployment of DL methods for cyber security applications. They also covered various attacks, such as malware, spam, insider threats, network intrusions and false data injection in their survey. Moreover, a discussion on the performance parameters, such as accuracy, false positive rate and F1-score were given.

Dasgupta et al. [26] discussed the recent research works on ML in cyber security. They described the mechanisms of cyber attacks along with some defense mechanisms. They also highlighted the security characteristics of deep learning methods. Some future research directions were also provided at the end.

**Table 1**
Comparison of exiting surveys.

| Survey | Contributions | Advantages | Limitations |
|---|---|---|---|
| Chen et al. [24] | • Provided a survey on adversarial attacks in reinforcement learning under AI security.<br>• Brief introduction on the most representative defense technologies against existing adversarial attacks was given. | • Coverage of adversarial attacks in reinforcement learning. | • Did not provide the details of potential attacks i.e., various privacy breaches.<br>• Moreover, comparative study of existing solutions for potential ML attacks wan not given.<br>• They did not provide any discussion on the threat model of the domain. |
| Berman et al. [25] | • Provided a literature review of deep learning (DL) methods for cyber security and covered various attacks.<br>• Discussed various DL methods i.e., deep autoencoders, restricted Boltzmann machines, recurrent neural networks and generative adversarial networks.<br>• Performance parameters i.e., accuracy, false positive rate, F1-score, etc., were highlighted. | • Coverage of DL methods and their deployment mechanism in the domain of cyber security. | • Did not provide the details of potential attacks i.e., various privacy breaches.<br>• Moreover, comparative study of existing solutions for potential ML attacks wan not given.<br>• Moreover, they did not discuss the threat model. |
| Dasgupta et al. [26] | • Provided recent research works on ML in cyber security.<br>• Described the mechanisms of cyber attacks and the corresponding defenses.<br>• Some future research directions were given. | • Described the mechanisms of cyber attacks along with their corresponding defense mechanisms.<br>• Some future research directions were given. | • They did not discuss the threat model of the domain. |
| Rosenberg et al. [27] | • Provided the review of adversarial attacks associated with ML techniques.<br>• Categorized the adversarial attack methods on the basis of their occurrence, attacker's goals and capabilities.<br>• Categorized associated defense methods in the cyber security.<br>• Highlighted some future research directions. | • Categorization of defense mechanisms.<br>• Provided some future research directions. | • They did not highlight the various threats and the associated threat model of the domain. |
| Proposed work | • Provided the details of various machine learning security attacks in cyber physical systems.<br>• Discussed some defense mechanisms to protect against these attacks.<br>• Presented the threat model of ML security mechanisms deployed in cyber systems.<br>• Discussed various issues and challenges of ML security mechanisms deployed in cyber systems.<br>• Provided a detailed comparative study on performance of the ML models under the influence of various ML attacks in cyber physical systems. | • Coverage of various ML attacks<br>• Explanation of the working mechanism of various ML attacks and analysis of their impact on the various performance parameters.<br>• Discuss the devastating effects of various ML attacks under the threat model. | • Future research directions were provided.<br>• Did not provide any discussion on the zero day attacks. |

Rosenberg et al. [27] provided a review of adversarial attacks associated with the ML techniques. They categorized the adversarial attack methods on the basis of their occurrence, an attacker's goals and the respective capabilities. They also categorized associated defense methods in the cyber security and provided some future research directions.

The summary of the existing surveys is then given in Table 1.

The details of other related works are given below.

ML security and its privacy is still a moderately explored domain of research. The groundwork for securing machine learning tasks from different types of attacks were grounded by Barreno et al. [28]. Barreno et al. [29] provided a discussion on a framework for expressing an analytical view of different classes of attacks on machine learning systems. Papernot [30] discussed the importance and need for securing integrity and privacy of machine learning and the associated threats. There was a discussion on frameworks and work done to achieve the privacy of machine learning tasks.

Xue et al. [31] discussed various security mechanisms used to protect some potential attacks. They also proposed a new technique to countermeasures the security related problems. Furthermore, they discussed about various attacks, which are possible on neural network models and non-neural network models. Liu et al. [32] elaborated the threats on machine learning model during training and deployment. They also described a taxonomy on the various defenses available and scope of research that could be carried.

Spring et al. [33] demonstrated the prevalent vulnerabilities in machine learning systems and key arising problems in machine learning algorithms from a security point of view. Managing these vulnerabilities is of vital importance. They discussed the existing practices

that could be used to counter these problems by subdividing management into six areas, like (a) discovery, (b) report intake, (c) analysis, (d) coordination, (e) disclosure, and (f) response. Auernhammer et al. [34] highlighted the attacks on machine learning task and had further classified attacks on the basis of their deployment stage.

Evtimov et al. [35] proposed an adversarial data generator attack algorithm and tested it on road sign classification in which generated adversarial examples caused a very high rate of miscalculation in prediction for the classifier. Papernot et al. [36] discussed about the various security and privacy vulnerabilities. It was further explained the way of adversarial attacks implementation via white-box and black-box knowledge of the model.

Guo et al. [37] and Yang et al. [38] demonstrated the different domains of utilizing machine learning models, which were susceptible to attacks. They demonstrated the procedure of machine learning based network flow and intrusion detection models. Song and Shmatikov [39] and Carlini et al. [40] focused on attacks on machine learning based classifiers and natural language text generation models. These models were widely used in chat-bots and predictive keyboards, which "memorize" the provided inputs for processing. This could further hampered the privacy attacks on the user's data.

Ateniese et al. [41] demonstrated the procedure of implementation of ML to build an attack model to interact with another classifier and extract vital information about the training dataset. Liu et al. [42] talked about the security issues in the domain of deep learning and summarized all the attacks that can occur in development and deployment phases.

Rigaki and Garcia [43] summarized the previous research works on the privacy attacks on machine learning models and classified them intuitively based on the model's knowledge (i.e., white-box or black-box). Gu et al. [44] discussed the various real-life test case studies incorporating machine learning. They also simulated the attacks on the models as per the identified vulnerabilities.

Tramer et al. [45] provided the details of model extraction attacks. They focused on the conducted research works on ML models with the help of prediction API's (or queries to the model). It demonstrated that the integrity of an deployed ML model could be hampered. It also elaborated on how a model can be extracted using just given class labels. Fredrikson et al. [46] introduced another category of attacks called as model inversion attacks. They had implemented them to reconstruct the training data and its vital features and membership records through hampering with the model parameters and the intuition and inner working of the model (preferably by API calls to the model). Hidano et al. [46,47] implemented the attack without the requirement of any non-sensitive attributes of the model. They highlighted the inversion attack without degrading accuracy of the model.

Al-Rubaie and Chang discussed [48] the privacy concerns in ML systems. In addition, they reviewed the attacks based on some possible approaches, like homomorphic encryption and garbled circuits which can build the privacy-preserving ML models. Furthermore, the countermeasures, like restriction and authentication on APIs to ML are discussed.

In this section, we provided the details of existing surveys related to the ML attacks, their defense approaches, advantages, and limitations. Moreover, a Table 1 for the comparison of the existing surveys of the domain was provided. In the next section, we discuss the issues and challenges of machine learning security deployed for cyber physical systems along with the details of the threat model of ML security.

## 3. Issues and challenges of machine learning security deployed for cyber physical systems

In this section, we discuss the various issues and challenges of machine learning security deployed for the cyber physical systems. We also present the threat model of ML security, in which we provide the details of all threats associated with the ML models. The details are given below.

### 3.1. Threat model of ML security

The threat model of ML security provide the details of various threats and attacks and their consequences. Usually the data, which is being utilized for the learning and testing purpose in ML receives through the open (insecure) channel. Therefore, the existing adversary $\mathcal{A}$ can interrupt the normal flow of ML task in many ways. As various attacks i.e., replay, man in the middle (MiTM) attack, impersonation, malware injection, flooding attacks, denial of service (DoS) attacks, distributed denial of service (DDoS) attacks, false data insertion, unauthorized data updates are possible. According to Dolev–Yao (DY) model, the communicating entities, communicate through the insecure channel, therefore, the exchanged information can be leaked or altered or deleted [49]. If a ML model learns through the data, from which some information was deleted or altered then there are the high chance that this ML model will produce the wrong outcomes (results). Hence there are the high chances that normal procedure of ML flow may get disturbed. Thus we need some security solutions to provide protection against these threats and attacks [5].

### 3.2. Issues and challenges of machine learning security deployed for cyber physical systems

Some of the potential issues and challenges of machine learning security are given below [5].

- **Security of deployed mechanism:** Most of the ML related tasks are executed through some ML models. If somehow it gets compromised then the performance of the entire system will be affected. Therefore, security of the ML models is very essential. The security of ML models can be ensured through various mechanism i.e., proper use of authentication schemes (like device to device authentication, user to device authentication), proper use of access control schemes (like, certificate based access control, certificate less access control) and intrusion detection schemes [50].
- **Accuracy of deployed ML model:** It is always desirable to get the high value of accuracy for some ML tasks. However, in some of the cases, we get very less value of accuracy. That happens because, we have not pre-processed the data in the proper manner or the selection of ML algorithm is not correct. Therefore, to get the high value of accuracy, we have to be very careful and selective. We should select the ML algorithm wisely and as per the scenario and available datasets [51].
- **Failure of deployed security mechanisms:** Though we deploy various kind of security mechanisms to protect the ML models and tasks. Still hackers get some chance to break into the system due to the existence of zero day vulnerabilities and their exploitation (zero day attack). Under the execution of such kinds of attack the working of ML models may get affected (sometimes stopped). Therefore, we need to tackle the zero day attacks carefully and go for the proper deployment and use of intrusion detection approaches. To provide more security we can go for the combination of intrusion detection schemes i.e., hybrid anomaly detection (for example, combination of signature based detection and anomaly based detection).
- **Interoperability:** ML security environment is the collection of various ML algorithm, security algorithm (i.e., encryption, digital signature, integrity). These algorithms have their own limitations and constraints. Therefore, we may have issues related to interoperability. Hence we should do the selection of these algorithms wisely, i.e., which security algorithm should be used with which ML algorithm, etc.
- **Obsolescence:** ML security environment is the collection of various ML algorithm, security algorithm and communication related algorithms as discussed earlier. These algorithms have their own limitations and some of them become obsolete when the time

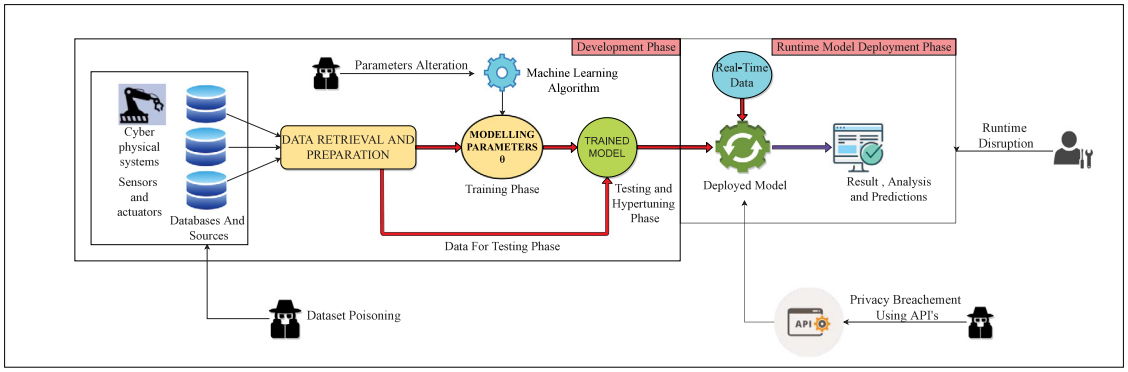


**Fig. 3.** Attack points in the machine learning workflow.

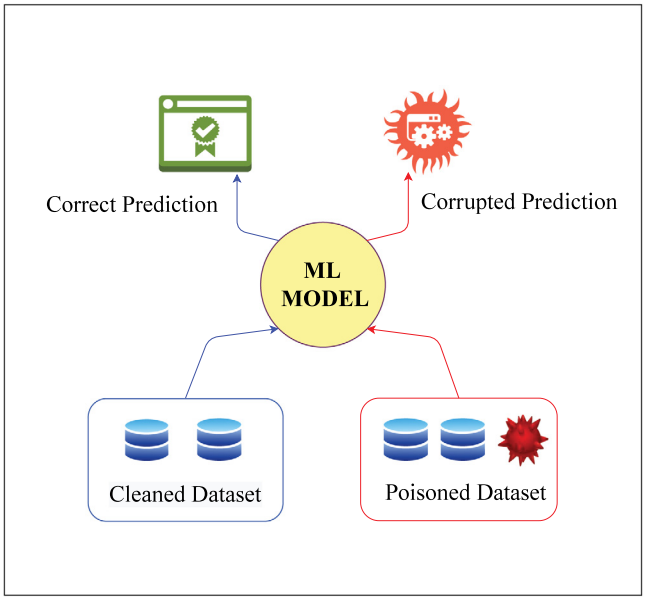


**Fig. 4.** Scenario of dataset poisoning attack.

passes. That raises issues related to the obsolescence. Hence steps of ML tasks should be updated accordingly and the tools and technologies, which become obsolete should not be used in the ML tasks.

In this section, we discussed the issues and challenges of machine learning security deployed for cyber physical systems. Issues and challenges like the security of deployed mechanisms, accuracy, failure of deployed security mechanisms, etc., were there. In the next section, we will discuss the various attacks possible on the ML models. Moreover, we will provide the details of some preventing methods of ML attacks.

## 4. Details of attacks possible on machine learning models and preventive measures

In this section, we provide the details of various attacks, which are possible on the machine learning models. These attacks can be broadly classified into four categories, i.e., dataset poisoning attacks (both real time and training data), model poisoning attacks, privacy breach and model inversion attacks and run-time disruption attacks [5,6]. Various notations and their meanings provided in Table 2 are used in this paper.

Fig. 3 visually depicts the attack points in the machine learning workflow. These attack points can be subjugated or interfered with to create disruption and cause privacy breach. It can further hamper the accuracy of the model. We can categorize attacks on the ML models into four broad categories, which are summarized below.

- **Dataset poisoning attack:** In this attacker inserts adversarial examples in dataset to cause the attacking model to produce incorrect predictions.
- **Model poisoning attack:** These types of attack focus on corrupting models by interfering with their internal workings and modifying the parameters.
- **Privacy breach attack:** These attacks work on exposing sensitive data of users and retrieving valuable information of the model.
- **Runtime disruption attack:** In this, intruder compromises the ML workflow to prevent efficient and accurate prediction by attacking the model in its execution.

The details of these attacks are given below.

### 4.1. Dataset poisoning attack

In this attack, the attacker utilizes the various techniques to infiltrate the training and testing data to disturb the normal functioning of a machine learning task. The attacker can utilize adversarial examples and can attack the data containing server or data lake from where raw data and photos have to be taken. The compromising of the data sources can lead to deployment of data, which can possibly alter the functioning of the ML model. It again changes the output of the classifier, which is very devastating in nature, i.e., for example, system is showing no illness, however, the patient suffers from the severe illness.

Fig. 4 denotes the result and repercussion of data-set poisoning attacks on a machine learning model. It denotes the inclusion of poisoned data (in this case an adversarial example) the same classifier which would predict correct and viable information, which can be corrupted to provide malicious results.

Adversarial examples [52] are a bothersome problem in machine learning. Even in testing and training phase it has been found out that they wrecked a havoc in the prediction capability and precision of an machine learning classifier [44]. To get a clear understanding and counter such attacks, we have to look at the problem from the attacker's perspective.

#### 4.1.1. Mechanism of dataset poisoning attack

The mechanism of dataset poisoning attack is elaborated in this section. The dataset poisoning attack can be launched by the attacker to make some changes in the dataset. The attacker can utilize the steps of a sophisticated SQL injection attack. This further makes the dataset corrupted. The ML model becomes poisoned as a result of the corrupted dataset and is unable to perform accurate analysis and prediction. The following steps can be used to explain the mechanism of a dataset poisoning attack.

**Table 2**
Notations used in the paper.

| Notation | Meaning |
|---|---|
| $\mathcal{A}$ | An adversary |
| $MLM_i$ | $i$th ML model |
| $num_{MLM}$ | Number of deployed ML models |
| $ACC_{MLM_i}$ | Accuracy of $MLM_i$ |
| $ACC^{\theta}_{MLM_i}$ | Threshold value of accuracy of $MLM_i$ |
| $A_{AE}$ | Adversarial examples from $\mathcal{A}$ |
| $CPr$ | Correct prediction |
| $WPr$ | Wrong prediction |
| $HP_{MLM_i}$ | Hyper parameters of $MLM_i$ |
| $DS_{MLM_i}$ | Dataset supplied to $MLM_i$ |
| $DMLM_i$ | Dummy model of $MLM_i$ |
| $PI_{U_i}$ | Private information (data) of user $U_i$ |

- **DPA1:** First, the attacker analyses the machine learning environment and goes to the data lake and warehouse where the training data is stored or sourced from.
- **DPA2:** Then points of breaching are identified, and normal input data flow is altered and is supplied with adversarial examples.
- **DPA3:** The adversarial examples that are supplied are made so as to train the model with incorrectly labeled and deceiving data supplemented with SQL injection.
- **DPA4:** Special focus is given by the attacker to prevent the poisoning examples from becoming outliers for the data and subsequently removing them as part of data cleaning.
- **DPA5:** With a sufficient ratio of the adversarial example, dataset poisoning is achieved, the accuracy of the model decreases, and the model becomes poisoned as it provides the wrong prediction to the labeled input data.

The mechanism of dataset poisoning attack is also summarized in Algorithm 1.

**Algorithm 1** Mechanism of dataset poisoning attack

```
for deployed ML models MLM_i, ∀i = 1, 2 ··· num_MLM do
    A does the analyses of ML environment
    A identifies points of breaching in MLM_i
    A supplies A_AE to MLM_i
    Training of MLM_i with A_AE
    A prevents A_AE from becoming outliers
    if A achieves dataset poising then
        MLM_i is under influence of dataset poison attack
        MLM_i provides WPr
        if ACC_MLM_i < ACC^θ_MLM_i then
            Break
        end if
        Continue
    end if
end for
```

Dataset poisoning attack can occur both on training and real data. It can happen through three ways—black box attack, white box attack and backdoor attack. The details are provided below.

- **Black box attack:** The procedure of an attack on "black box" model is given in Fig. 5. Here black box model defines to an model. The computational task that generally occurs in such a complex manner that the features, parameters and internal working of attacked machine learning model are not accessible to the attacker. However, attacker does the analysis of feedback or prediction result based on the adversarial examples supplied [37]. Black box attacks have been attempted by the researchers on ML models in recent times [53] through poisoning a model with adversarial examples without knowing the features of the model

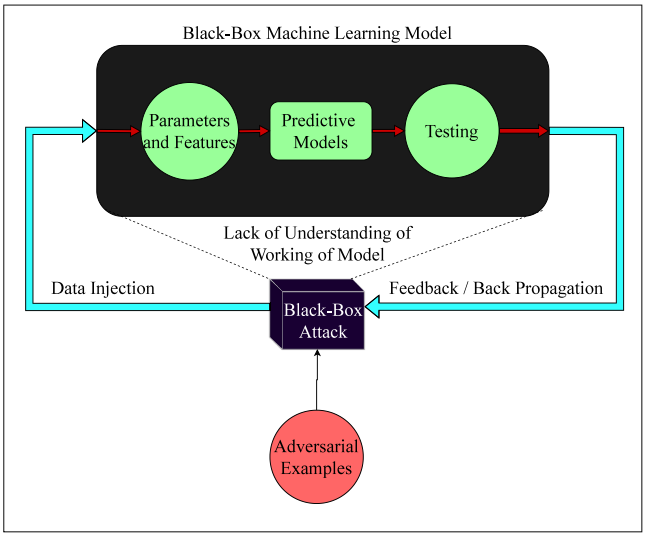


**Fig. 5.** Scenario of black box attack.

at a great 84.24% misclassification rate of the injected adversarial examples. After that the calibration increased it to misclassification of 96.19% of the adversarial inputs to model hosted at Amazon Web Services. Xinyun [54] used poisoning strategies to create backdoor instances to misled the system to clasify them as target label. The backdoor attack comes under this category. We have discussed the backdoor attack below in this section. Jagielski et al. [55] proposed a black-box poisoning attack based on linear regression and visualized it on a range of datasets. They also showcased a defense mechanism based on the regression parameters and is able to isolate most of the poisoning point on dataset.
- **White box attack:** In this attack, attacker ($\mathcal{A}$) has access to the model's parameters, whereas in black box attack, $\mathcal{A}$ has no access to these parameters.
- **Backdoor attacks:** Backdoor attacks are also generally implemented on ML tasks (preferably on CNN models). These attacks [56] are mostly on training set data and modifies the prediction through getting access to the model and its dataset. They do this by inserting a trigger in supplied adversarial data which when "activates" cause the model to misclassify the input with the backdoor induced model. Barni et al. [57] presented a backdoor attack on CNN model which corrupts the training data stealthily without any label poisoning such that it does not get identified by the user and successfully corrupts the prediction. Lorenz et al. [58] proposed a backdoor poisoning attack against network certifiers by identifying new attack vectors that come with the use of certifiers (used for improving robustness of the model) in machine learning models. A scheme of defenses have been advocated for backdoor attacks. Chen et al. [59] has done the mitigation of backdoor attack induced using activation cluster methodology. It was applicable only on deep neural networks. Weber et al. [60] used deterministic test-time augmentation mechanism to check for any backdoor attack. Liu et al. [61] combined existing fine tuning and pruning defense to develop an efficient defense technique, which removed any backdoor instance rather than locating it. Aladag et al. [62] and Steinhardt et al. [63] have taken different approaches to counter dataset poisoning attacks efficiently. The approaches are summarized below.

#### 4.1.2. Mechanism to counter dataset poisoning attack

The mechanism to counter dataset poisoning attack is given below.

- **MCDPA1:** The developer can utilized various cryptographic techniques i.e., symmetric and asymmetric key cryptographic algorithms for the detection purpose. The utilization of some outliers in dataset can be done to detect any injection of malicious data.
- **MCDPA2:** We train an outlier detector in parallel to the ML model, which helps in filtering out any data it deems poisoned, as it does not comply with how normally it should have been predicted at the deployment.
- **MCDPA3:** Some specific data and there proposed prediction can be "tokenised" and maintained separately during training and can be compared during the deployment to get awareness of any occurring attack.
- **MCDPA4:** Also a red flag can be issued if the difference in accuracy within the training and deployment phase is not within a acceptable limit due to higher chance of data poisoning attack at the deployment.
- **MCDPA5:** We can also implement artificial neural network based-generative model parallel to the ML model with pre-computed accuracy score to cross verify the poisoning of the data.

### 4.2. Model poisoning attack

Parameter alteration is a method utilized by hackers to generate faulty output by interfering with the classifier and altering the parameters through which the classifier prepares ML model. The person can switch sensitivity limits, rate of accession, cause under-fitting, and over-fitting by accessing the classifier file via an unprotected ML task. The algorithm poisoning is achieved through interacting with unprotected classifier file. In model poisoning attacks following procedure is executed.

#### 4.2.1. Procedure of model poisoning attack

Model poisoning attacks is executed with the help of following steps.

- **MPA1:** First of all the attacker interacts with the machine learning environment and find redundancy in the classifier.
- **MPA2:** The attacker can change or modify the training algorithm to generate wrong output and publish deceiving results.
- **MPA3:** Further the attacker can interact with the hyper-parameters and cause the model to over-fit or under-fit and create problems in the testing phase.
- **MPA4:** As the parameters of the algorithm is hard-coded and can be dynamically modified unprotected uses of the algorithm can speed up the Attacks.

The mechanism of model poisoning attack is also summarized in Algorithm 2.

**Algorithm 2** Mechanism of model poisoning attack

**for** deployed $MLM_i$, $\forall i = 1, 2 \cdots num_{MLM}$ **do**
  $\mathcal{A}$ does the analyses of ML environment
  $\mathcal{A}$ tries to find out redundancy in $MLM_i$'s classifier
  $\mathcal{A}$ modifies training process of $MLM_i$
  $\mathcal{A}$ interacts with $HP_{MLM_i}$
  **if** it is so **then**
    $MLM_i$ becomes over-fit or under-fit
    **if** $ACC_{MLM_i} < ACC^{\theta}_{MLM_i}$ **then**
      Break
    **end if**
    Continue
  **end if**
**end for**

Various attacks schemes and countermeasures for model poisoning have been discussed by researchers. Most of the attacks on model's working follow same intuition utilized in dataset poisoning and fine tuning it further. Shen et al. [64] demonstrated a poisoning attack "Tensorclog", which decreased the accuracy of deployed model via interfering with the predicting parameters of model. With the utilization of adversarial examples, it modifies the weights or degree of correlation between variables of model, which causes gradient vanishing in the model. A model hyperparameter stealing attack has been formalized [65] which steals hyperparameter value of different models with great success of some regression models. It successfully steals loss function, learning rate and model features through computing gradient of functions. Zhang et al. [66] demonstrated a model reconstructing scheme, which rather implemented conventional procedure to reconstruct private training data utilized generative models on publicly available data and invert it causing a privacy issue. We have discussed in Section 4.3.

#### 4.2.2. Mechanism to counter model poisoning attack

Following mechanism can be used to counter the model poisoning attack.

- **MCMPA1:** When the user creates a ML classifier he can vectorize the constraints and predictive labels to embed its own $ID$ using hashing.
- **MCMPA2:** During the hyper-parameters tuning phase the hash $ID$ of the classifier can be updated by the Administrator.
- **MCMPA3:** Finally during the deployment phase the hash $ID$ can be matched with the deployed model's hash $ID$ to check for any discrepancies in both. Further this identify if the model has been altered.
- **MCMPA4:** Blockchain can be implemented on the classifier to retain its hash value from possible attacks in a secured network.

### 4.3. Privacy breach

Sensitive user's data and model's internal working can be compromised through a variety of methods. Unprotected files and lack of encrypted pipelines during training and deployment phases of ML task can leak the data and enables an unauthorized user to interfere with the model. Papernot et al. [36] discussed the various privacy risks in ML workflow and their repercussions in exposing confidentially of users after deployment of model. They also discussed some privacy preserving schemes to protect the privacy of model and usage of noise generation to provide differential privacy by randomizing model's Behavior. Sensitive data of users can also be compromised from a model which has been trained by utilizing exposed API's. It further compromises the model's working by infecting the dataset with malicious input and by getting output from API call to reverse engineer the process. It exposes the inner working of the ML model.

For this attack 6 we formulate that most of the prediction tasks are behind the protected framework. So the intruder utilizes underlying API and publishes the leaked data sources. The attacker tends to create an dummy ML model, which is identical to the targeted model. Utilizing API calls the intruder sends edge-case data and receives prediction output by the target model. With this data the attacker tries to form an intuition for his model by referring data sourced, which have been linked to model or tries to get access to the data. This information is then compared with prediction result. The published results are used by the developer to create a working ML model, which behaves like the target model. Further with the dummy model the attacker can perform various attacks like membership inference attacks. A brief overview of membership inference attack is discussed at the end of this section.

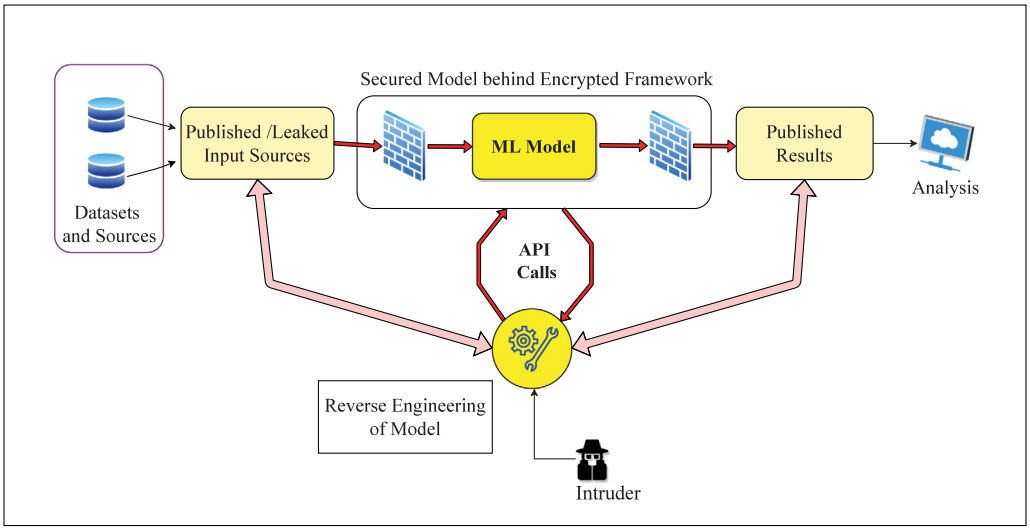


**Fig. 6.** Scenario of privacy breach on a ML model through API calls.

*4.3.1. Procedure of exposing sensitive data through API calls and reverse engineering*

This procedure of exposing sensitive data through API calls and reverse engineering (privacy breach attack) is given below.

- **PBA1:** Initial step is to understand the problem statement of ML task.
- **PBA2:** Then required input will be scrapped from provided input sources or dataset leaked due to poor or null authorization of the server and published output can also be studied and analyzed. Then its predictions can be uncovered.
- **PBA3:** Try to find intuition of model using the data recovered in Step *PBA2*.
- **PBA4:** Using the intuition try to reverse engineer the process and create a dummy model of ML model to use with predicted approximate parameters.
- **PBA5:** With built dummy model use it on published input resources to retrieve private information and data from the users.
- **PBA6:** Optimization steps would involve utilizing API calls on the running original model and retrieving specific outputs, which are based on specific input arguments for optimizing the edge cases in the new dummy model. Best practice is not to use common parameters, which may have been provoked by the user to as a part of prevention technique as it may nullify the attack.

Cryptography approaches, notably homomorphic encryption based defense mechanism have been used [67] to protect the privacy of the model and its data. Also Abadi et al. reduced privacy loss with the inclusion of techniques [68] to train the model with differential privacy. In cases of protected privacy in a federate scenario Mohassel and Zhang [69] implemented privacy schemes on a distributed frame of reference more specifically two-server model where data owners distribute their private data among two non-colluding servers to train the ML model on the combined data using secure two-party computation. The mechanism of exposing sensitive data through API calls and reverse engineering (privacy breach attack) is also summarized in Algorithm 3.

Privacy breach attack can also be conducted through the steps of membership inference attacks. The details of membership inference attacks are given below.

*4.3.2. Membership inference attacks*

The scenario of membership inference attacks is provided in Fig. 7. Membership attack on ML model [70] utilizes techniques and attacks discussed till now (underlying API's) to inject some personal data in the model work space and determines the utilization of that specific data

**Algorithm 3** Mechanism of privacy breach attack

1: **for** deployed $MLM_i$, $\forall i = 1, 2 \cdots num_{MLM}$ **do**
2: $\mathcal{A}$ understands the problem statement
3: $\mathcal{A}$ uncovers $DS_{MLM_i}$
4: $\mathcal{A}$ does the analysis of $DS_{MLM_i}$
5: $\mathcal{A}$ finds out intuition of model for $DS_{MLM_i}$
6: $\mathcal{A}$ creates $DMLM_i$
7: $\mathcal{A}$ uses $DMLM_i$ with $DS_{MLM_i}$
8: $DS_{MLM_i}$ produces $PI_{U_i}$
9: $\mathcal{A}$ verifies $PI_{U_i}$
10: **if** $PI_{U_i}$s are not desirable **then**
11: $\mathcal{A}$ optimizes the process
12: Repeat the above steps from Step 2
13: **end if**
14: **end for**

during training of model. With this membership information several privacy risk can be issued due to lack of confidentiality of membership on one's record in the model. This leakage of membership data can cause some serious privacy concern, which are discussed in [71,72]. There is the great research potential in the development of some prevention methods.

Various attack schemes have been demonstrated by researchers which have been formalized below. Stacey et al. [73] demonstrated membership inference attacks on a variety of data sets and compiled their findings in their paper [73]. They have further inculcated topic of federated learning and visualized multiple participation vulnerabilities. Milad *at al.* [74] discussed quantified membership interface attacks in a white box as well as in a federated scenario. Sablayrolles *at el.* [75] utilized Bayesian learning and noise-injected training data to find out that there are no distinguishable effect on adversary knowledge of model working to achieve membership attacks on the ML models. The paper also discusses some simple defense schemes like cropping, rounding, which can decrease the risk of inference. Researchers have found the way to address it by utilizing noise induced adversarial examples with in a black box setting [76]. Through inducing noise in the dataset, they have successfully thwarted the attempts of membership interference attacks. Another fast defense mechanism is provided later [77] which works on the principle of training a autoencoder. It takes the confidence scores of the target model as input and "concentrate" them on the clear patterns. Its defense is improvised in defending a particular attack via the adversarial learning. In the other method [40] the differential

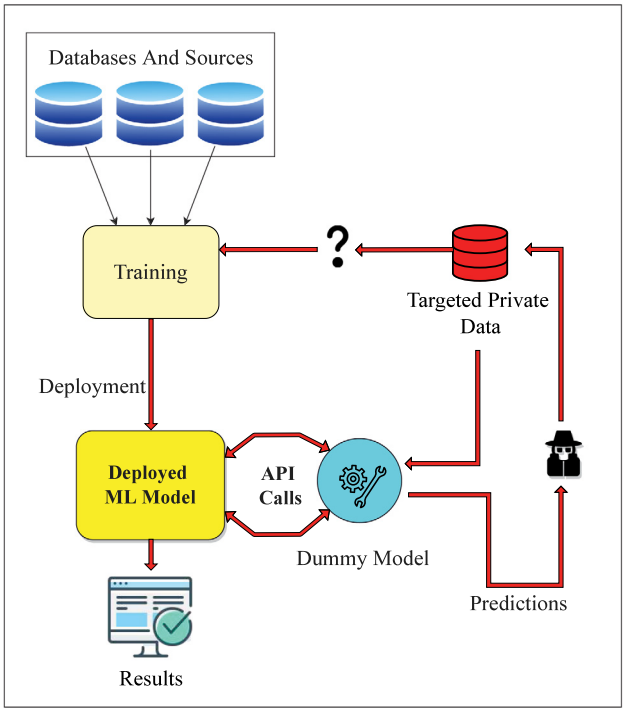


**Fig. 7.** Scenario of membership inference attacks.

privacy algorithms have been provided. They utilize them on the generative models, which can reduce membership inference attack by reducing memorization by model. Fig. 7 describes the process of a membership inference attack on ML model.

The procedure of a membership inference attack is described below.

- First, the intruder sources the data whose membership status has to be found.
- Then, normally a dummy model is created through reverse engineering the target model through methods discussed till now.
- The targeted data record is given to the dummy model for prediction.
- An intuition is developed by the intruder through comparing target model's published prediction and retrieved prediction of the instance from the dummy model.
- Though this intuition the intruder finds out if the specific data instance was in the dataset used for training of the model.
- With these findings membership status of the data instance is breached.

### 4.4. Runtime disruption attack

Runtime disruption attack is used by $\mathcal{A}$ to end or delay the ML task. $\mathcal{A}$ generally targets the server during the deployment phase and tries to remotely disrupt the ML process. It takes advantage of the susceptible areas of the environment and breaks in and disrupts the normal functioning of the task, which results in wastage of resources and the time of the users. During the deployment of trained model many issues can be taken place in the ML's generalized workflow [78]. In addition to scaling, executing and integrating there is an issue during deployment and predicting phase to the model and its worklet environment. $\mathcal{A}$ identifies the points and penetrates the run time server, which utilizes various techniques like phishing, denial-of- service (DoS) attack and SQL injection attack. That further causes problems in the workflow of the server carrying the ML task [79]. These concerns are of grave importance, especially when the model is deployed in government or healthcare sectors as minor disruption or privacy breach can have preposterous implications [80]. This attack can be countered by decentralizing the ML work space. Alaasam et al. [81] discussed the virtualization process for the big data, which can be further integrated into ML tasks. Furthermore, blockchain based methodology can be implemented to further bifurcate and implementing distributed ML to protect the integrity and privacy of the users and the models [82].

A mechanism to prevent runtime disruption attack is given below.

- **MPRTA1:** This procedure implements the techniques of parallel computing to safeguard a ML task from the attacks, which occur during deployment of trained model on real-data that are made to cause runtime disruption.
- **MPRTA2:** It consist of one master node which contains the trained model and the instruction. It then divides and gives them to its sub units to perform specif task. These units further categorize into racks and multiple slave nodes, which basically perform the real task in the ML workflow.
- **MPRTA3:** The slave nodes would be responsible for the collection of data. They process it and provides output on the basis of instructions received by the master node.
- **MPRTA4:** The multiple machine working on same unit of data, therefore, the ML task will be self sufficient to continue the task without any denial or error.
- **MPRTA5:** To avoid redundancy and integrity of training and deploying of model the master node accumulates multiple results and compile it to complete task even in the case of attack.

Fig. 8 depicts the decentralization of ML workload to secure the process from any case of disruption attack on the deployed model. It visualizes how the suggested model could have sub-models with there own private rack and nodes to generate rack-awareness algorithm to keep on continuing the ML task without any disruption. In Fig. 8, the architectural ML model is divided into multiple machines which perform the prediction task separately. Each ML model is subdivided into sub-model which are performed by different machine located in a different cluster. Each cluster of sub-ML model utilizes master or root node concept to do the task with retaining availability of system to continue the task. Each sub model has a system of root nodes, which allocate specific independent task to a system of computing machine called root node.

As referenced in Fig. 8 each root node computes prediction task of multiple racks, implying one node does the computing of more than one sub-model. This ensures that when there is the disruption attack at rack (root node) level, then other rack (root node) can do the computing for them. The prediction task still executes with the increment in the computational time. This summarizes how rack-awareness algorithm is utilized to retain the integrity of the ML task in case of attack on the server.

In this section, we discussed the various attacks possible on the ML models, such as dataset poisoning attack, model poisoning attack, privacy breach, etc. Moreover, we provided the details of some methods which could be used to mitigate these attacks. In the next section, we provide the comparison of the impact of discussed attacks along with the details of their defense mechanisms.

## 5. Comparative study

In this section, we compare the impact of discussed attacks and the performance of their defense mechanisms. The attacks can be classified into four categories, i.e., poisoning attacks, privacy, accessing and inference attacks.

The comparative study of various attacks is given in Tables 3, 4, 5 and 6. The comparative study of various defenses mechanisms is given in Tables 7, 8 and 9.

Adversarial and dataset poisoning is one of the oldest field in ML security. Papernot et al. [53] proposed a attack which misclassifies the

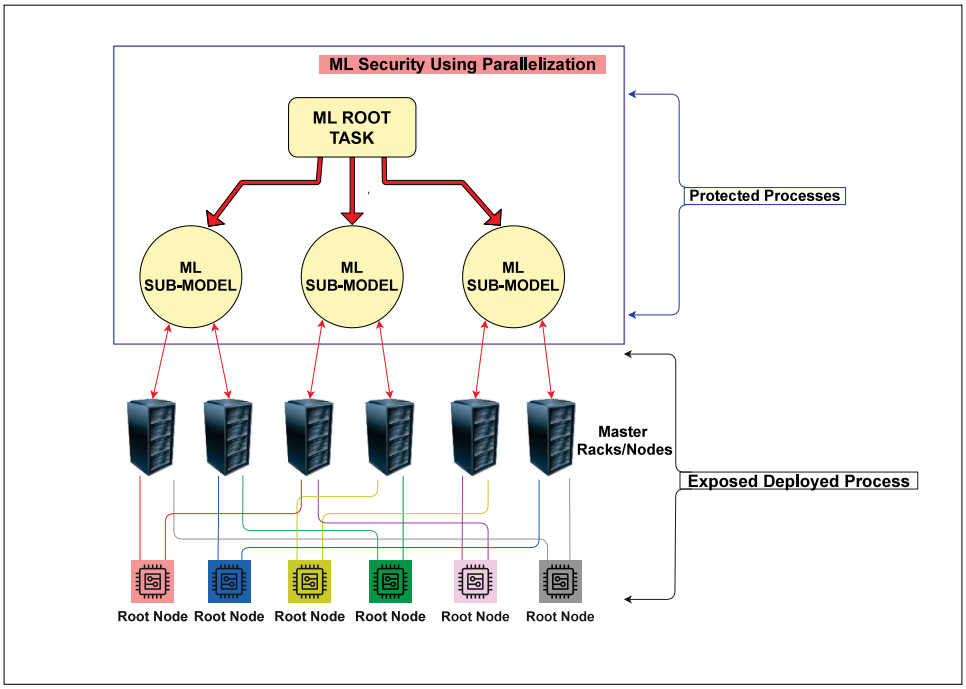


**Fig. 8.** Parallelization for securing ML task from runtime disruption.

**Table 3**
Performance comparison of poisoning attack schemes.

| Scheme and Year | Approach used | Attack setting | Misclassification rate | Remarks |
|---|---|---|---|---|
| Papernot et al. (2017)[53] | Black box based attack that crafts adversarial examples without knowledge of the model | Black-Box | 84.24% further increased to 96.19% | N/A |
| Kurakin et al. (2016) [52] | Impact of real life physical changes on conversion of clean data to adversarial data | White-Box | 98.0% | Paper visualizes how minute features like contrast, brightness can misclassify the model. |
| Evtimov et al. (2017) [35] | Algorithm to generate robust adversarial examples under different physical conditions | Black-Box | 84.8% | Showcased an evaluation methodology to study the repercussion of poisoning attacks in real world scenarios. |
| Chen et al. (2017) [54] | Backdoor targeted attack through data poisoning | Black Box | 97.90% | Injection of little as 5 poisoned samples could achieve > 99% attack accuracy on model. |
| Jagielski et al. (2018) [55] | Optimization framework for poisoning attack on linear regression models | Model specific | N/A | The attack is not model generic, only works on model, datasets based on linear regression. |

Note: N/A: not available.

**Table 4**
Performance comparison of backdoor attack schemes.

| Scheme and Year | Approach used | Error rate | Advantages | Shortcomings |
|---|---|---|---|---|
| Gu et al. (2019) [56] | Poisons the model and attach a trigger mechanism which gets activated by specific input data | 99.44% | Attacks stealthily and poisons without hampering overall performance of the model | Has to be induced in training of the model, cannot be attacked at inference time. |
| Barni et al. (2019) [57] | Corruption of model stealthily by avoiding poisoning labels of the corrupted samples through only corrupting the target class | Average attack rate 90% | Attacks stealthily by avoiding label poisoning which backdoor attacks used to misclassify the input | The stealthiness of attack was decreased by increasing the strength of attack, so model was only successful to an extent if label poisoning has to be done. |
| Lorenz et al. (2021) [58] | Backdoor attack through identifying attack vectors used by network certifiers | 90% decrease in accuracy | In ideal situations indirect attack (can only inject poisoned data) accuracy score had decreased up to 8.8% | Fine-tuning the model caused inverse effect by decreasing overall robustness. |

Note: N/A: not available.

adversarial examples through injecting poisoned data. It achieved accuracy of 96.19% on MNIST dataset using Amazon web services. Kurakin et al. [52] showcased how physical features like brightness and contrast of data can poison an image with maximum incorrect prediction of

**Table 5**
Performance comparison of attack schemes on model features.

| Scheme and Year | Approach used | Attack setting | Attack success rate | Remarks |
|---|---|---|---|---|
| Hidano et al. (2017) [47] | Model inversion attack which reveals sensitive attributes without the requirement of knowledge about non-sensitive attributes of the target user using output of the model | White-Box | Highest success rate of attack was 0.741 | They generalized the inversion attack by Fredrikson [46] by structuring the amount of auxiliary information at attack time. |
| Wang and Gong (2018) [65] | Estimation of model's hyperparameter through computing gradient of objective functions | White-Box | Estimation errors less than $[10^{-4}]$ | Can only estimate loss function and regularization terms. Need to extract model specific parameters like learning rate, mini-batch size. |
| Zhang et al. (2020) [66] | Utilization of public information about the model to reconstruct it effectively | White-Box and Black Box | Attack accuracy 76%(black-box) 80% (white-box) | Attack scheme performs upto 75% more than existing attack scheme [46]. |

**Table 6**
Performance comparison of inference attack schemes.

| Scheme and Year | Approach used | Attack setting | Attack rate | Remarks |
|---|---|---|---|---|
| Sablayrolles et al. (2019) [75] | Used Bayes strategies to compare membership inference attack on white-box and black-box models | White-Box and Black-box | 90.8% Attack accuracy | Deeming of conclusion that membership inference attack only depends on loss function. |
| Milad et al. (2019) [74] | Exploiting stochastic gradient descent vulnerabilities to leak membership inference | White-box and federated based | Attack accuracy 74.3% (white box) and 82.1% (federated scenario) | There was formulation of efficient adversary setting when user utilizes federated learning in ML. |
| Stacey et al. (2019) [73] | Membership inference attacks on a wide variety of ML models and utilization of federate learning | Black-Box | 95.74% accuracy | Membership attack is demonstrated as model type independent. |

**Table 7**
Performance comparison of defenses against poisoning attacks.

| Scheme and Year | Approach used | Accuracy rate | Advantages | Shortcomings |
|---|---|---|---|---|
| Aladag et al. (2019) [62] | Improvisation of robustness to poisoning attacks using auto generative model | N/A | Efficient attack to identify malicious data | Model has to learn the manipulated data during training, model's diversity has to be improved. |
| Jagielski et al. (2018) [55] | Estimation of parameters and using a reduced loss function to isolate any points deemed poisoned | 0.768 | Robustness against poisoned data with similar distribution value used in training | Works only on regression based models. |
| Gupta et al. (2020) [83] | Isolation of poisoned point from testing set through comparison with nearest neighbors | 0.918 | Works extremely well against clean-label poisoning attacks by reducing > 99% poisoned data. | Could be only used for clean-label data poisoning attacks. |
| Chen et al. (2021) [84] | Utilizing generative adversarial networks to reconstruct a clean model and distinguish its factors from poisoned one | 0.931 | Very effective in defending against various type of poisoning attacks | N/A |

Note: N/A: not available.

98.0% of adversarial data. These ideas are further extended in [35] to generate the adversarial examples. Chen *at el.* [54] proposed a backdoor targeted attack through data poisoning and achieved 97.90% attack success rate. Through injecting five poisoned samples more than 99% attack accuracy was achieved. Defense in dataset poisoning has been tested and compared by [84]. We have reviewed multiple papers in our paper and did a comparative study. Jagielski et al. [55] introduced an attack called r-attack and also introduced a defense technique named "TRIM", which is an extremely fast defense. Gupta et al. [83] proposed an attack deep-KNN which isolated any point using KNN algorithm also performs satisfactorily defense against the attack with 91% accuracy score [85]. A defense technique De-pois which worked on all attacks was analysed [55,84–86]. It was a robust attack in specific poisoning defense with providing upto 93% accuracy score on the 93% attack [87].

Backdoor attacks have been implemented by Gu [56] via inducing a trigger in dataset with mislabeling of 0.56% (mislabeling more than 99%). Backdoor attack with stealth [57] gives an average attack rate of 90% and avoids label poisoning. Lorenz et al. [58] conducted backdoor attack on black-box and white-box with upto 90% decrease in accuracy. It causes decrement of accuracy to 8.8% in case of indirect attack (via injecting poisoned data). Defenses against it includes the mechanism of Weber et al. [60] that uses deterministic test-time augmentation mechanism. It is a uniform mechanism and can increase robustness of a model and provide the efficient defense with an accuracy of 98.6%. Later on Liu et al. [61] combines existing fine tuning and pruning defense to develop an efficient defense technique which has an accuracy of 97.7%.

Privacy attacks and defense was a vital part of our study. Membership inference attack have been discussed by Sablayrolles et al. [75] to compare membership inference attack on white-box and black-box models with 90.8% attack accuracy without augmentation defenses like cropping and rounding off. Further 79.5% attack accuracy with data augmentation. Milad et al. [74] utilized the vulnerabilities of stochastic gradient descent to leak membership inference data with 74.3% attack accuracy further improved it to 82.1% in the federated scenario. Finally, Stacey et al. [73] conducted membership inference attacks on a wide variety of ML models and achieved 95.74% accuracy against "Purchases-100 dataset" through the decision tree model.

**Table 8**
Performance comparison of defenses against privacy attacks.

| Scheme and Year | Approach used | Effectiveness | Advantages | Shortcomings |
|---|---|---|---|---|
| Le Trieu et al. (2018) [67] | Preserving privacy using additively homomorphic encryption | 97% defense accuracy | It protects the privacy by protecting the gradients | Like other Deep learning based defense, it is computational intensive. |
| Mohassel and Zhang (2017) [69] | SecureML | 98.62% defense accuracy | Scalable and can perform complex operation on multiple devices with privacy protection | Complex to distribute the workload without an efficient algorithm. |
| Abadi et al. (2016) [68] | Usage of differential privacy technique during training to protect privacy | Decrease in accuracy of 1.3% as opposed to | Protects the privacy of the model from attacker | Through minuscule, affects the accuracy of the model. |
| Jia et al. (2019)[76] | Utilizing vulnerability of attack from noise induced adversarial examples | 50.8% | No computational resources are required,as only noise is added | Can work only on black-box inference attacks, could be extended to white-box inference attacks. |
| Yang et al. (2020) [77] | Utilizing autoencoder to take confidence score and isolate them | reduction of inference attack accuracy by 15% | Purifier can be further specialized in defending a particular attack | Due to complex training of purifier model, heavy task. |

**Table 9**
Performance comparison of defenses against accessing attacks.

| Scheme and Year | Approach used | Accuracy effect | Advantages | Shortcomings |
|---|---|---|---|---|
| Chen et al. (2018) [59] | Activation cluster based methodology for removing backdoors from data | 99.97% | Defense is robust to complex poisoning attacks in which the classes are multimodal | There is a high computational overload. |
| Liu et al. (2018) [61] | Combining fine tuning and pruning defense to develop an efficient defense technique | 98.6% | Defense remove backdoor induced trigger automatically rather than locating | The defense is not applicable and studied on sequential processing models like natural language processing. |
| Weber et al. (2020) [60] | Model deterministic test-time augmentation mechanism to check for any backdoor attack | 97.7% | Proposed a unified framework to certify model robustness | Defense is computationally high due to training of a large number of models on the smoothed datasets. |

Various inference defenses have been implemented with utilization of noise induced adversarial examples. Yang et al. [77] proposed a defense named MemGaurd which successfully defended against black-box inference attacks and achieved 50.8% inference attack accuracy on "Texas100 dataset". Yang et al. [77] worked on the principle of training a autoencoder and provided an accuracy that reduced membership inference attack accuracy by 15%. It again increased the model inversion error by a factor of four at a minute drop of 0.4% classification accuracy drop. Its defense was improvised via adversarial learning.

Some other privacy risk have also been induced. Hidano et al. [47] conducted a model inversion attack, which revealed the sensitive attributes without the knowledge of non-sensitive attributes at the success rate of 74.1%. Wang and Gong et al. [65] stole the model's hyperparameter through computing gradient of objective function with estimation errors less than $[10^{-4}]$ in some regression models. Zhang et al. [66] reconstructed model using publicly available information with accuracy of 76% (highest attack accuracy on CelebA dataset with black-box knowledge) and 80% (highest attack accuracy on MNIST dataset with white-box knowledge). It showcases it performs upto 75% more than the existing attack scheme [46]. Homomorphic encryption mechanism have been used to protect the privacy [67]. Abadi et al. [68] reduced privacy loss with the inclusion of techniques to train the model with differential privacy. The resultant model does the protection of privacy of the model. However, it decreases the accuracy of overall model by 1.3% in case of MNIST dataset with $(8, [10^{-5}])$ differential privacy. Mohassel and Zhang [69] implemented privacy schemes on a distributed frame of reference more specifically the two-server model. In this model the data owners distribute their private data among two non-colluding servers, which train the ML model on the combined data via the secure two-party computation and achieves 98.62% accuracy on MNIST dataset.

In this section, we provided the comparison of the impact of discussed attacks along with the details of their defense mechanisms, such as "poisoning attack mitigation schemes", "backdoor attack mitigation schemes", "inference attack mitigation schemes", etc. In the next section, we conclude the work along with a discussion of some important future research directions.

## 6. Conclusion and future research directions

We provided the details of various Machine learning security attacks (i.e., dataset poisoning attack, model poisoning attack, privacy breach, membership inference attack, runtime disruption attack), which are possible on the machine learning models deployed in the cyber physical systems. We also discuss some of the defense mechanisms which can be deployed to protect against these attacks. We then presented the threat model of ML security, in which we provided the details of all threats associated with the ML models deployed for the cyber physical systems. Further, we discussed the issues and challenges (like the security of deployed mechanism, accuracy, failure of deployed security mechanisms, etc.) of ML security deployed for the cyber physical systems. Finally, we provided a comparative study of the performance of the ML models under the influence of various ML attacks along with the performance of various defense mechanisms.

The following identified future research directions could be helpful to the researchers:

- **Uses of deep learning algorithms:** The study of deep learning algorithms with respect to generative adversarial networks (GANs) have been a topic of much research interest in the AI field, and it shows a promising research aspect. The research field has intended to focus primarily on the enhancement of pre-existing neural models, and more work should be undertaken on

how GANs could increase their robustness from a cyber security perspective point of view.

- **Undercover impact of federated learning:** Although federated learning has been introduced to a vast extent with the incorporation of big data processing, very little work has been proposed in its effectiveness to counter an attack during the deployment of the ML tasks. A rigorous effort must be made to propose algorithms that can utilize decentralization to offer runtime protection of any ML tasks.
- **Computational factor:** Numerous works have been discussed upon incorporating cost and resources due to the use of computationally high tasks in protecting the integrity of the ML tasks. However, significant work is still required to propose some lightweight schemes to preserve the privacy of the ML processes.
- **Lack of standardization of evaluation parameters:** The studies on the schemes on various attacks and their defenses in ML systems have distinct performance parameters. They do not have a general correlation of performance parameters (i.e., accuracy, F1-score, detection rate, and false-positive rate). Hence, there should be some standardized evaluation parameters. As a result, some standard criteria need to be introduced and followed in categorizing and demonstrating techniques in a structured manner.
- **Handing of heterogeneous data:** Data in the cyber physical system comes from a variety of sources, each with its own set of qualities and characteristics. As a result, the ML model has a difficult time dealing with it. We will need to put in more effort in this procedure, particularly in the data preprocessing. These issues also exist in ML security, which should be handled carefully by the other researchers working in the same domain.
- **Full proof security:** Researchers in ML security always attempt to build a security scheme that can mitigate the numerous possible threats associated with the ML models. However, there are situations when methods are insufficient to prevent attacks on the ML model. As a result, rigorous testing, analysis, and validation should be performed prior to the deployment of security schemes in order to discover vulnerabilities in those schemes. This could be an important research direction in the future.

## Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

## Acknowledgments

The authors thank the anonymous reviewers and the associate editor for their valuable feedback on the paper, which helped us to improve its quality and presentation.

## References

[1] Ziaur Rahman, Ibrahim Khalil, Xun Yi, Mohammed Atiquzzaman, Blockchain-based security framework for a critical industry 4.0 cyber-physical system, IEEE Commun. Mag. 59 (5) (2021) 128–134.
[2] Aakarsh Rao, Nadir Carreon, Roman Lysecky, Jerzy Rozenblit, Probabilistic threat detection for risk management in cyber-physical medical systems, IEEE Softw. 35 (1) (2018) 38–43.
[3] Mojtaba Kordestani, Mehrdad Saif, Observer-based attack detection and mitigation for cyberphysical systems: A review, IEEE Syst. Man Cybern. Mag. 7 (2) (2021) 35–60.
[4] Jairo Giraldo, Esha Sarkar, Alvaro A. Cardenas, Michail Maniatakos, Murat Kantarcioglu, Security and privacy in cyber-physical systems: A survey of surveys, IEEE Des. Test 34 (4) (2017) 7–17.
[5] Abdulmalik Humayed, Jingqiang Lin, Fengjun Li, Bo Luo, Cyber-physical systems security-A survey, IEEE Internet Things J. 4 (6) (2017) 1802–1831.
[6] Ziaur Rahman, Ibrahim Khalil, Xun Yi, Mohammed Atiquzzaman, Blockchain-based security framework for a critical industry 4.0 cyber-physical system, IEEE Commun. Mag. 59 (5) (2021) 128–134.
[7] Viraaji Mothukuri, Prachi Khare, Reza M. Parizi, Seyedamin Pouriyeh, Ali Dehghantanha, Gautam Srivastava, Federated-learning-based anomaly detection for IoT security attacks, IEEE Internet Things J. 9 (4) (2022) 2545–2554.
[8] K. Shailaja, B. Seetharamulu, M. A. Jabbar, Machine learning in healthcare: A review, in: Second International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2018, pp. 910–914.
[9] Andreas Kamilaris, Francesc X. Prenafeta-Boldú, Deep learning in agriculture: A survey, Comput. Electron. Agric. 147 (2018) 70–90.
[10] Vinay Chamola, Vikas Hassija, Sakshi Gupta, Adit Goyal, Mohsen Guizani, Biplab Sikdar, Disaster and pandemic management using machine learning: A survey, IEEE Internet Things J. PP (2020).
[11] Zhengping Luo, Shangqing Zhao, Zhuo Lu, Jie Xu, Yalin E. Sagduyu, When attackers meet AI: Learning-empowered attacks in cooperative spectrum sensing, IEEE Trans. Mob. Comput. 21 (5) (2022) 1892–1908.
[12] Afeez Ajani Afuwape, Ying Xu, Joseph Henry Anajemba, Gautam Srivastava, Performance evaluation of secured network traffic classification using a machine learning approach, Comput. Stand. Interfaces 78 (2021) 103545.
[13] Rabia Khan, Pardeep Kumar, Dushantha Nalin K. Jayakody, Madhusanka Liyanage, A survey on security and privacy of 5G technologies: Potential solutions, recent advancements, and future directions, IEEE Commun. Surv. Tutor. 22 (1) (2020) 196–248.
[14] Gan Sun, Yang Cong, Jiahua Dong, Qiang Wang, Lingjuan Lyu, Ji Liu, Data poisoning attacks on federated machine learning, IEEE Internet Things J. (2021).
[15] Geetanjali Rathee, Naveen Jaglan, Sahil Garg, Bong Jun Choi, Dushantha Nalin K. Jayakody, Handoff security using artificial neural networks in cognitive radio networks, IEEE Internet Things Mag. 3 (4) (2020) 20–28.
[16] Huadi Zheng, Qingqing Ye, Haibo Hu, Chengfang Fang, Jie Shi, Protecting decision boundary of machine learning model with differentially private perturbation, IEEE Trans. Dependable Secure Comput. 19 (3) (2022) 2007–2022.
[17] Prabhat Kumar, Randhir Kumar, Gautam Srivastava, Govind P. Gupta, Rakesh Tripathi, Thippa Reddy Gadekallu, Neal N. Xiong, PPSF: A privacy-preserving and secure framework using blockchain-based machine-learning for IoT-driven smart cities, IEEE Trans. Netw. Sci. Eng. 8 (3) (2021) 2326–2341.
[18] S. Pundir, M. Wazid, D.P. Singh, A.K. Das, J.J.P.C. Rodrigues, Y. Park, Intrusion detection protocols in wireless sensor networks integrated to internet of things deployment: Survey and future challenges, IEEE Access 8 (2020) 3343–3363.
[19] Mohammad Wazid, Ashok Kumar Das, Sachin Shetty, Prosanta Gope, Joel J.P.C. Rodrigues, Security in 5G-enabled internet of things communication: Issues, challenges, and future research roadmap, IEEE Access 9 (2021) 4466–4489.
[20] Yichen Hou, Sahil Garg, Lin Hui, Dushantha Nalin K. Jayakody, Rui Jin, M. Shamim Hossain, A data security enhanced access control mechanism in mobile edge computing, IEEE Access 8 (2020) 136119–136130.
[21] Uttam Ghosh, Pushpita Chatterjee, Deepak Tosh, Sachin Shetty, Kaiqi Xiong, Charles Kamhoua, An SDN based framework for guaranteeing security and performance in information-centric cloud networks, in: IEEE 10th International Conference on Cloud Computing (CLOUD), Honolulu, USA, 2017, pp. 749–752.
[22] Dominik Sisejkovic, Farhad Merchant, Lennart M. Reimann, Rainer Leupers, Deceptive logic locking for hardware integrity protection against machine learning attacks, IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst. 41 (6) (2022) 1716–1729.
[23] Sujeet More, Jimmy Singla, Sahil Verma, Kavita, Uttam Ghosh, Joel J.P.C. Rodrigues, A.S.M. Sanwar Hosen, In-Ho Ra, Security assured CNN-based model for reconstruction of medical images on the internet of healthcare things, IEEE Access 8 (2020) 126333–126346.
[24] Tong Chen, Jiqiang Liu, Yingxiao Xiang, Wenjia Niu, Endong Tong, Zhen Han, Adversarial attack and defense in reinforcement learning-from AI security view, Cybersecurity 2 (1) (2019) 11.
[25] Daniel S. Berman, Anna L. Buczak, Jeffrey S. Chavis, Cherita L. Corbett, A survey of deep learning methods for cyber security, Information 10 (4) (2019).
[26] Dipankar Dasgupta, Zahid Akhtar, Sajib Sen, Machine learning in cybersecurity: a comprehensive survey, J. Def. Model. Simul. (2020).
[27] Ishai Rosenberg, Asaf Shabtai, Yuval Elovici, Lior Rokach, Adversarial machine learning attacks and defense methods in the cyber security domain, ACM Comput. Surv. 54 (5) (2021).
[28] M. Barreno, Blaine Nelson, R. Sears, A. Joseph, J. Tygar, Can machine learning be secure? in: ACM Symposium on Information, Computer and Communications Security (ASIACCS)'06, Taipei Taiwan, 2006, pp. 1–10.
[29] Marco Barreno, Blaine Nelson, Anthony D. Joseph, J.D. Tygar, The security of machine learning, Mach. Learn. 81 (2) (2010) 121–148.
[30] Nicolas Papernot, A marauder's map of security and privacy in machine learning, in: 11th ACM Workshop on Artificial Intelligence and Security, Toronto, Canada, 2018.
[31] Mingfu Xue, Chengxiang Yuan, Heyi Wu, Yushu Zhang, Weiqiang Liu, Machine learning security: Threats, countermeasures, and evaluations, IEEE Access 8 (2020) 74720–74742.
[32] Qiang Liu, Pan Li, Wentao Zhao, Wei Cai, Shui Yu, Victor C.M. Leung, A survey on security threats and defensive techniques of machine learning: A data driven view, IEEE Access 6 (2018) 12103–12117.
[33] Jonathan M. Spring, April Galyardt, Allen D. Householder, Nathan M. VanHoudnos, On managing vulnerabilities in AI/ML systems, 2020, CoRR, abs/2101.10865.

[34] Katja Auernhammer, Ramin Tavakoli Kolagari, Markus Zoppelt, Attacks on machine learning: Lurking danger for accountability, in: AAAI Workshop on Artificial Intelligence, Honolulu, Hawaii, USA, 2019.

[35] Ivan Evtimov, Kevin Eykholt, Earlence Fernandes, Tadayoshi Kohno, Bo Li, Atul Prakash, Amir Rahmati, Dawn Song, Robust physical-world attacks on machine learning models, 2017, CoRR, arXiv:1707.08945.

[36] Nicolas Papernot, Patrick McDaniel, Arunesh Sinha, Michael P. Wellman, SoK: Security and privacy in machine learning, in: IEEE European Symposium on Security and Privacy (Euro S & P), London, UK, 2018, pp. 399–414.

[37] Sensen Guo, Jinxiong Zhao, Xiaoyu Li, Junhong Duan, Dejun Mu, Xiao Jing, A black-box attack method against machine-learning-based anomaly network flow detection models, Secur. Commun. Netw. 2021 (2021) 5578335.

[38] Kaichen Yang, Jianqing Liu, Chi Zhang, Yuguang Fang, Adversarial examples against the deep learning based network intrusion detection systems, in: MILCOM 2018 - 2018 IEEE Military Communications Conference (MILCOM), Los Angeles, USA, 2018, pp. 559–564.

[39] Congzheng Song, Vitaly Shmatikov, The natural auditor: How to tell if someone used your words to train their model, 2018, ArXiv, abs/1811.00513.

[40] Nicholas Carlini, Chang Liu, Jernej Kos, Úlfar Erlingsson, Dawn Song, The secret sharer: Evaluating and testing unintended memorization in neural networks, in: Proceedings of the 28th USENIX Conference on Security Symposium, 2019, pp. 267–284.

[41] Giuseppe Ateniese, Giovanni Felici, Luigi V. Mancini, Angelo Spognardi, Antonio Villani, Domenico Vitali, Hacking smart machines with smarter ones: How to extract meaningful data from machine learning classifiers, Int. J. Secur. Netw. 10 (2015) 137–150.

[42] Ximeng Liu, Lehui Xie, Yaopeng Wang, Jian Zou, Jinbo Xiong, Zuobin Ying, Athanasios V. Vasilakos, Privacy and security issues in deep learning: A survey, IEEE Access 9 (2021) 4566–4593.

[43] Maria Rigaki, Sebastian Garcia, A survey of privacy attacks in machine learning, 2020, abs/2007.07646.

[44] Tianyu Gu, Brendan Dolan-Gavitt, Siddharth Garg, BadNets: Identifying vulnerabilities in the machine learning model supply chain, 2017, ArXiv, abs/1708.06733.

[45] Florian Tramer, Fan Zhang, Ari Juels, Michael K. Reiter, Thomas Ristenpart, Stealing machine learning models via prediction APIs, in: 25th USENIX Conference on Security Symposium, Vancouver, Canada, 2016, pp. 601–618.

[46] Matt Fredrikson, Somesh Jha, Thomas Ristenpart, Model inversion attacks that exploit confidence information and basic countermeasures, in: 22nd ACM SIGSAC Conference on Computer and Communications Security, Denver, USA, 2015 pp. 1322–1333.

[47] Seira Hidano, Takao Murakami, Shuichi Katsumata, Shinsaku Kiyomoto, Goichiro Hanaoka, Model inversion attacks for prediction systems: Without knowledge of non-sensitive attributes, in: 15th Annual Conference on Privacy, Security and Trust (PST), Calgary, Canada, 2017, pp. 115–11509.

[48] Mohammad Al-Rubaie, J. Morris Chang, Privacy-preserving machine learning: Threats and solutions, IEEE Secur. Priv. 17 (2) (2019) 49–58.

[49] D. Dolev, A.C. Yao, On the security of public key protocols, IEEE Trans. Inform. Theory 29 (2) (1983) 198–208.

[50] Uttam Ghosh, Xinshu Dong, Rui Tan, Zbigniew Kalbarczyk, David K.Y. Yau, Ravishankar K. Iyer, A simulation study on smart grid resilience under software-defined networking controller failures, in: Proceedings of the 2nd ACM International Workshop on Cyber-Physical System Security, Association for Computing Machinery, Xi'an, China, 2016, pp. 52–58.

[51] Uttam Ghosh, Pushpita Chatterjee, Sachin Shetty, A security framework for SDN-enabled smart power grids, in: IEEE 37th International Conference on Distributed Computing Systems Workshops (ICDCSW), Atlanta, USA, 2017, pp. 113–118.

[52] Alexey Kurakin, Ian J. Goodfellow, Samy Bengio, Adversarial examples in the physical world, 2016, CoRR, abs/1607.02533.

[53] Nicolas Papernot, Patrick D. McDaniel, Ian J. Goodfellow, Somesh Jha, Z. Berkay Celik, Ananthram Swami, Practical black-box attacks against deep learning systems using adversarial examples, in: Proceedings of the ACM Asia Conference on Computer and Communications Security, Abu Dhabi, UAE, 2017.

[54] Xinyun Chen, Chang Liu, Bo Li, Kimberly Lu, Dawn Song, Targeted backdoor attacks on deep learning systems using data poisoning, 2017, ArXiv, abs/1712.05526.

[55] Matthew Jagielski, Alina Oprea, B. Biggio, Chang Liu, C. Nita-Rotaru, Bo Li, Manipulating machine learning: Poisoning attacks and countermeasures for regression learning, in: IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 2018, pp. 19–35.

[56] Tianyu Gu, Kang Liu, Brendan Dolan-Gavitt, Siddharth Garg, BadNets: Evaluating backdooring attacks on deep neural networks, IEEE Access 7 (2019) 47230–47244.

[57] Mauro Barni, Kassem Kallas, Benedetta Tondi, A new backdoor attack in CNNs by training set corruption without label poisoning, in: IEEE International Conference on Image Processing (ICIP), 2019, pp. 101–105.

[58] Tobias Lorenz, Marta Kwiatkowska, Mario Fritz, Backdoor attacks on network certification via data poisoning, 2021, ArVix, abs/2108.11299.

[59] Bryant Chen, Wilka Carvalho, Nathalie Baracaldo, Heiko Ludwig, Benjamin Edwards, Taesung Lee, Ian Molloy, Biplav Srivastava, Detecting backdoor attacks on deep neural networks by activation clustering, in: SafeAI@AAAI, 2019.

[60] Maurice Weber, Xiaojun Xu, Bojan Karlas, Ce Zhang, Bo Li, RAB: Provable robustness against backdoor attacks, 2020, ArXiv, abs/2003.08904.

[61] K. Liu, Brendan Dolan-Gavitt, S. Garg, Fine-pruning: Defending against backdooring attacks on deep neural networks, in: RAID, 2018.

[62] Merve Aladag, Ferhat Ozgur Catak, Ensar Gul, Preventing data poisoning attacks by using generative models, in: 1st International Informatics and Software Engineering Conference (UBMYK), Ankara, Turkey, 2019, pp. 1–5.

[63] Jacob Steinhardt, Pang Wei Koh, Percy Liang, Certified defenses for data poisoning attacks, in: Proceedings of the 31st International Conference on Neural Information Processing Systems, 2017, pp. 3520–3532, abs/1706.03691.

[64] Juncheng Shen, Xiaolei Zhu, De Ma, TensorClog: An imperceptible poisoning attack on deep neural network applications, IEEE Access 7 (2019) 41498–41506.

[65] Binghui Wang, Neil Zhenqiang Gong, Stealing hyperparameters in machine learning, in: IEEE Symposium on Security and Privacy (SP), 2018, pp. 36–52.

[66] Y. Zhang, R. Jia, H. Pei, W. Wang, B. Li, D. Song, The secret revealer: Generative model-inversion attacks against deep neural networks, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), Seattle, USA, 2020, pp. 250–258.

[67] Le Trieu Phong, Yoshinori Aono, Takuya Hayashi, Lihua Wang, Shiho Moriai, Privacy-preserving deep learning via additively homomorphic encryption, IEEE Trans. Inf. Forensics Secur. 13 (2018) 1333–1345.

[68] Martin Abadi, Andy Chu, Ian Goodfellow, H. Brendan McMahan, Ilya Mironov, Kunal Talwar, Li Zhang, Deep learning with differential privacy, in: Proceedings of the 2016 ACM SIGSAC Conference on Computer and Communications Security, ACM, Vienna Austria, 2016.

[69] Payman Mohassel, Yupeng Zhang, SecureML: A system for scalable privacy-preserving machine learning, in: IEEE Symposium on Security and Privacy (S&P), San Jose, USA, 2017, pp. 19–38.

[70] Reza Shokri, Marco Stronati, Vitaly Shmatikov, Membership inference attacks against machine learning models, in: Proceedings of the IEEE Symposium on Security and Privacy, 2016, abs/1610.05820.

[71] Gaoyang Liu, Chen Wang, Kai Peng, Haojun Huang, Yutong Li, Wenqing Cheng, SocInf: Membership inference attacks on social media health data with machine learning, IEEE Trans. Comput. Soc. Syst. 6 (5) (2019) 907–921.

[72] Apostolos Pyrgelis, Carmela Troncoso, Emiliano De Cristofaro, Knock knock, who's there? Membership inference on aggregate location data, in: Proceedings of the 25th Network and Distributed System Security Symposium, 2017, abs/1708.06145.

[73] Stacey Truex, Ling Liu, Mehmet Emre Gursoy, Lei Yu, Wenqi Wei, Demystifying membership inference attacks in machine learning as a service, IEEE Trans. Serv. Comput. (2019).

[74] Milad Nasr, Reza Shokri, Amir Houmansadr, Comprehensive privacy analysis of deep learning: Passive and active white-box inference attacks against centralized and federated learning, in: 2019 IEEE Symposium on Security and Privacy (SP), 2019.

[75] Alexandre Sablayrolles, Matthijs Douze, Yann Ollivier, Cordelia Schmid, Hervé Jégou, White-box vs black-box: Bayes optimal strategies for membership inference, 2019.

[76] Jinyuan Jia, Ahmed Salem, Michael Backes, Yang Zhang, Neil Zhenqiang Gong, MemGuard: Defending against black-box membership inference attacks via adversarial examples, pp. 259–274.

[77] Ziqi Yang, Bin Shao, Bohan Xuan, E. Chang, Fangfang Zhang, Defending model inversion and membership inference attacks via prediction purification, 2020, ArXiv, abs/2005.03915.

[78] Andrei Paleyes, Raoul-Gabriel Urma, N. Lawrence, Challenges in deploying machine learning: a survey of case studies, 2020, ArXiv, abs/2011.09926.

[79] Julian Jang-Jaccard, Surya Nepal, A survey of emerging threats in cybersecurity, J. Comput. System Sci. 80 (5) (2014) 973–993, Special Issue on Dependable and Secure Computing.

[80] Nader Sehatbakhsh, Ellie Daw, Onur Savas, Amin Hassanzadeh, Ian McCulloh, Security and privacy considerations for machine learning models deployed in the government and public sector (white paper), 2020, CoRR, abs/2010.05809.

[81] Ameer Alaasam, Gleb Radchenko, Andrei Tchernykh, Comparative analysis of virtualization methods in big data processing, Supercomput. Front. Innov.: Int. J. 61 (2019) 48–79.

[82] Vaidotas Drungilas, Evaldas Vaičiukynas, Mantas Jurgelaitis, Rita Butkienė, Lina Čeponienė, Towards blockchain-based federated machine learning: Smart contract for model inference, Appl. Sci. 11 (2021).

[83] Neal Gupta, W. Ronny Huang, Liam Fowl, Chen Zhu, Soheil Feizi, Tom Goldstein, John P. Dickerson, Strong baseline defenses against clean-label poisoning attacks, in: ECCV Workshop, 2020, pp. 55–70.

[84] Jian Chen, Xuxin Zhang, Rui Zhang, Chen Wang, Ling Liu, De-pois: An attack-agnostic defense against data poisoning attacks, 2021, abs/2105.03592.

[85] Chen Zhu, W. Ronny Huang, Ali Shafahi, Hengduo Li, Gavin Taylor, Christoph Studer, Tom Goldstein, Transferable clean-label poisoning attacks on deep neural nets, 2019.

[86] Han Xiao, Huang Xiao, Claudia Eckert, Adversarial label flips attack on support vector machines, in: Proceedings of the 20th European Conference on Artificial Intelligence, Montpellier, France, 2012, pp. 870–875.

[87] Battista Biggio, Blaine Nelson, Pavel Laskov, Support vector machines under adversarial label noise., J. Mach. Learn. Res. - Proc. Track 20 (2011) 97–112.

**Jaskaran Singh** is pursuing Bachelor of Technology in Computer Science and Engineering with specialization in Data Science in the Department of Computer Science and Engineering at Graphic Era Deemed to be University Dehradun, India. His area of research is machine learning, data science, machine learning security and intrusion detection system.

**Mohammad Wazid** received M.Tech. degree in Computer Network Engineering from Graphic Era deemed to be University, Dehradun, India and Ph.D. degree in Computer Science and Engineering from the International Institute of Information Technology, Hyderabad, India. He is currently working as an Associate Professor in the Department of Computer Science and Engineering, Graphic Era deemed to be University, Dehradun, India. He is also the head of Cybersecurity and IoT research group at Graphic Era deemed to be University, Dehradun, India. Prior to this, he was working as an assistant professor at the Department of Computer Science and Engineering, Manipal Institute of Technology, MAHE, Manipal, India. He was also a postdoctoral researcher at cyber security and networks lab, Innopolis University, Innopolis, Russia. His current research interests include information security, remote user authentication, Internet of things (IoT), cloud/fog/edge computing and blockchain. He has published more than 100 papers in international journals and conferences in the above areas. Some of his research findings are published in top cited journals, such as the IEEE Transactions on Dependable and Secure Computing, IEEE Transactions on Smart Grid, IEEE Internet of Things Journal, IEEE Transactions on Industrial Informatics, IEEE Journal of Biomedical and Health Informatics (formerly IEEE Transactions on Information Technology in Biomedicine), IEEE Consumer Electronics Magazine, IEEE Access, Future Generation Computer Systems (Elsevier), Computers & Electrical Engineering (Elsevier), Computer Methods and Programs in Biomedicine (Elsevier), Security and Communication Networks (Wiley) and Journal of Network and Computer Applications (Elsevier). He has also served as a Program Committee Member in many international conferences. He was a recipient of the University Gold Medal and the Young Scientist Award by UCOST, Department of Science and Technology, Government of Uttarakhand, India. He has received Dr. A. P. J. Abdul Kalam award for his innovative research works. He has also received ICT Express (Elsevier) Journal Best Reviewer award for the year of 2019.

**Ashok Kumar Das** received a Ph.D. degree in computer science and engineering, an M.Tech. degree in computer science and data processing, and an M.Sc. degree in mathematics from IIT Kharagpur, India. He is currently an Associate Professor with the Center for Security, Theory and Algorithmic Research, International Institute of Information Technology, Hyderabad, India. He is also working as a visiting faculty with the Virginia Modeling, Analysis and Simulation Center, Old Dominion University, Suffolk, VA 23435, USA. His current research interests include cryptography, system and network security, security in vehicular ad hoc networks, smart grids, smart homes, Internet of Things (IoT), Internet of Drones, Internet of Vehicles, Cyber–Physical Systems (CPS) and cloud computing, intrusion detection, blockchain and AI/ML security. He has authored over 305 papers in international journals and conferences in the above areas, including over 265 reputed journal papers. He was a recipient of the Institute Silver Medal from IIT Kharagpur. He is on the editorial board of IEEE Systems Journal, Journal of Network and Computer Applications (Elsevier), Computer Communications (Elsevier), Journal of Cloud Computing (Springer), Cyber Security and Applications (Elsevier), IET Communications, KSII Transactions on Internet and Information Systems, and International Journal of Internet Technology and Secured Transactions (Inderscience), and has served as a Program Committee Member in many international conferences. He also severed as one of the Technical Program Committee Chairs of the first International Congress on Blockchain and Applications (BLOCKCHAIN'19), Avila, Spain, June 2019, International Conference on Applied Soft Computing and Communication Networks (ACN'20), October 2020, Chennai, India, and second International Congress on Blockchain and Applications (BLOCKCHAIN'20), L'Aquila, Italy, October 2020. His Google Scholar h-index is 66 and i10-index is 193 with over 12,600 citations. He is a senior member of the IEEE.

**Vinay Chamola** is currently Assistant Professor in Dept. of Electrical and Electronics Engg., BITS-Pilani, Pilani campus. Vinay received his B.E. degree in Electrical & Electronics Engineering and Master's degree in communication engineering from Birla Institute of Technology & Science (BITS), Pilani, India in 2010 and 2013 respectively. He received his Ph.D. degree in Electrical and Computer Engineering from the National University of Singapore, Singapore, in 2016. From June to Aug. 2015, he was a visiting researcher at the Autonomous Networks Research Group (ANRG) at the University of Southern California (USC), USA. After his PhD, he worked as a postdoctoral researcher at the National University of Singapore in the area of Internet of Things. His research interests include IoT security, Blockchain, 5G resource management, Drones, VANETs and BCI. He is an Associate Editor of various journals including Ad Hoc Networks, IEEE Internet of Things Magazine, IEEE Networking letters, IET Networks, and IET Quantum Communications.

**Mohsen Guizani** (Fellow, IEEE) received the B.S. (Hons.) and M.S. degrees in electrical engineering and the M.S. and Ph.D. degrees in computer engineering from Syracuse University, Syracuse, NY, USA, in 1984, 1986, 1987, and 1990, respectively. He is currently a Professor with the Department of Computer Science and Engineering, Qatar University, Qatar. Previously, he has served in different academic and administrative positions for the University of Idaho, Western Michigan University, the University of West Florida, the University of Missouri-Kansas City, the University of Colorado-Boulder, and Syracuse University. He is the author of nine books and more than 1100 publications in refereed journals and conferences. He guest edited a number of special issues in IEEE journals and magazines. His research interests include wireless communications and mobile computing, computer networks, mobile cloud computing, security, and smart grid. He is also a Senior Member of ACM. Throughout his career, he received three teaching awards and four research awards. He was a recipient of the 2017 IEEE Communications Society Wireless Technical Committee (WTC) Recognition Award, the 2018 AdHoc Technical Committee Recognition Award for his contribution to outstanding research in wireless communications and Ad-Hoc Sensor networks, and the 2019 IEEE Communications and Information Security Technical Recognition (CISTC) Award for outstanding contributions to the technological advancement of security. He was the Chair of the IEEE Communications Society Wireless Technical Committee and the Chair of the TAOS Technical Committee. He has served as a member, the chair, and the general chair of a number of international conferences. He is also the Editor-in-Chief of the IEEE Network. He serves on the editorial boards for several international technical journals. He also serves the Founder and the Editor-in-Chief for Wireless Communications and Mobile Computing journal (Wiley). He has also served as the IEEE Computer Society Distinguished Speaker. He is also the IEEE ComSoc Distinguished Lecturer. He is an Associate Editor of various journals including Ad Hoc Networks, IEEE Internet of Things Magazine, IEEE Networking letters, IET Networks, and IET Quantum Communications.