Journal of Information Security and Applications 70 (2022) 103341

Contents lists available at ScienceDirect

Journal of Information Security and Applications

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

Adversarial machine learning in IoT from an insider point of view

Fatimah Aloraini [a,b,*], Amir Javed [a], Omer Rana [a], Pete Burnap [a]

[a] School of Computer Science and Informatics, Cardiff University, Cardiff, United Kingdom
[b] College of Sciences and Humanities, Shaqra University, Shaqra, Kingdom of Saudi Arabia

ARTICLE INFO

Keywords:
Adversarial machine learning
Insider
IoT
Cybersecurity
Machine learning
Deep learning

ABSTRACT

With the rapid progress and significant successes in various applications, machine learning has been considered a crucial component in the Internet of Things ecosystem. However, machine learning models have recently been vulnerable to carefully crafted perturbations, so-called adversarial attacks. A capable insider adversary can subvert the machine learning model at either the training or testing phase, causing them to behave differently. The vulnerability of machine learning to adversarial attacks becomes one of the significant risks. Therefore, there is a need to secure machine learning models enabling the safe adoption in malicious insider cases. This paper reviews and organizes the body of knowledge in adversarial attacks and defense presented in IoT literature from an insider adversary point of view. We proposed a taxonomy of adversarial methods against machine learning models that an insider can exploit. Under the taxonomy, we discuss how these methods can be applied in real-life IoT applications. Finally, we explore defensive methods against adversarial attacks. We believe this can draw a comprehensive overview of the scattered research works to raise awareness of the existing insider threats landscape and encourages others to safeguard machine learning models against insider threats in the IoT ecosystem.

1. Introduction

Internet of Things (IoT), as defined by the International Organization for Standardization (ISO) and the International Electrotechnical Commission (IEC), is *"an infrastructure of interconnected entities, people, systems and information resources together with services which process and react to information from the physical world and from the virtual world"* [1]. In simple terms, the IoT extends the current Internet by providing connectivity between physical things and cyberspace.

The IoT has rapidly grown in prominence in the last decade. IoT applications range from consumer-oriented, such as smart homes, to enterprise-oriented IoT domains, such as Industrial IoT (IIoT). According to the McKinsey Global Institute [2], around the world, an estimated 127 new IoT devices connect to the Internet every second. In 2020, for the first time, the number of IoT connections represented 54% of the active connections and hence exceeded the number of non-IoT connections (smartphones, laptops, and computers) [3].

The IoT concept comprises a broad ecosystem of interconnected things, including sensors/devices, communication services, data analytics, and a user interface. Machine Learning (ML) based data analytics is one of the valuable components in the IoT ecosystem for several reasons: First, ML plays a vital role in learning from the vast amount of data generated by IoT devices, allowing meaningful insights to be drawn. Second, it provides embedded intelligence in the IoT application that is leveraged to cope with various IoT problems [4,5]. Finally, IoT devices encompass the notion of actuating; thus, there is a need to deploy a decision-making technique [6].

A well-known application of ML models in the IoT environments is cybersecurity. ML, including deep learning (DL), has proven its success in protecting cyber- environments from cyber-threats. Among these various threats are insider threats. ML-based approaches have been proposed for insider attack detection, including those that consider behavior analysis [7–9]. Insider threats are widely perceived to be significant and often even considered to be more damaging than outsider threats [10,11] . The protection against adversarial insiders is usually more challenging than others for several reasons [12]. First and foremost, insiders tend to access sensitive resources because they are trusted. Second, there are more attacks venues available to the insider adversaries than the external adversaries. Finally, insider threats are more difficult to detect.

Despite the advantages of ML in the context of insider threats detection, recent studies, such as [13,14], have shown that ML models can be vulnerable to a novel class of attacks, so-called adversarial attacks, posing severe security threats to the systems that deploy them. Deploying ML models without considering their vulnerability to adversarial attacks can cause them to be the weakest link in the entire chain of

* Corresponding author at: School of Computer Science and Informatics, Cardiff University, Cardiff, United Kingdom.
E-mail address: alorainif@cardiff.ac.uk (F. Aloraini).

https://doi.org/10.1016/j.jisa.2022.103341

Available online 1 October 2022

IoT security [15]. A capable insider can subvert the ML-based model at either the training or testing phase, causing them to behave differently with carefully crafted input perturbations. For example, suppose an organization asks its employees to submit an ID photo for its ML-based facial recognition control system; an insider adversary can provide a poisoned photo that gives the adversary control of the face recognition system.

According to Gartner's report [16], by 2022, 30% of cyberattacks will involve adversarial attacks such as data poisoning, model theft, or adversarial perturbation. As a result, a robust ML models against insider threats are required, where a robust model is defined as a ML-based model that can withstand adversarial attacks. However, to build these models, one must understand how the insider adversary can manipulate the ML models' input points. That leads us to the following question *how can ML-based systems in IoT environments adapt when an insider adversary actively manipulates the system's inputs?*

Research on Adversarial Machine Learning (AML)*"the study of effective machine learning techniques against an adversarial opponent"* [13], has only begun, and many complex obstacles remain unaddressed. Thus, we aim to analyze the insider threats against ML models deployed in IoT environments.

Our paper aims to review and organize the body of knowledge in the AML and IoT literature from an insider point of view. We review and taxonomize the adversarial attacks that can be exploited by the insiders, present applications of these methods in real-life domains, and discuss what defense methods have been proposed so far. Our review focuses on supervised ML-based systems associated with IoT. We hope this work will raise awareness of the existing adversarial insider threats landscape and encourages others to safeguard ML-based systems from malicious insiders in IoT ecosystems.

The structure of the paper is as follows: Section 1 gives the introduction and motivation. Section 2 discusses the related works, limitations, and contributions. Section 3 describes the research methodology. Section 4 gives a background of the primary concepts used in the paper. Section 5 discusses adversarial insider threats in IoT. Section 6 presents real-life applications of adversarial insider threats. Section 7 depicts adversarial insider attacks taxonomy. Sections 8 and 9 discuss two use cases. Section 10 elaborates on current countermeasures against adversarial insider threats. Section 11 concludes the paper.

2. Related work

In the field of cybersecurity, the concept of AML and its impact on the performance of ML-based systems has raised substantial concerns in the academic and industrial communities. Our survey lies at the junction of AML and the IoT from insider threats point of view. Thus, we briefly summarize the related comprehensive surveys and compare our work with them. The surveys surrounding the AML literature can be categorized into three environments: traditional IT systems, Cyber–physical Systems (CPSs), and the IoT.

Duddu [15] and Wang et al. [17] discuss the research works on traditional systems under adversarial conditions. Both surveys mainly focus on attack methods and defense strategies that are presented in the literature. Moreover, Duddu and Wang et al. review the privacy-preserving methods which are used to protect the sensitive data, data used to build the ML-based systems, against information leakage attacks. Biggio and Roli [18] provide a detailed review of adversarial machine learning evolution over the last ten years. From earlier to more recent years, Biggio and Roli review the adversarial machine learning literature in the context of computer vision and cybersecurity. Their review aims to provide a deep understanding of the security properties of deep and non-deep learning, respectively. Pitropakis et al. [19] provided a detailed taxonomy of evasion and poisoning attacks against traditional systems that deploy ML models. The proposed taxonomy can be broadly classified into two main phases: the attack preparation phase and the attack manifestation phase. The authors organize the literature knowledge according to different applications, including visual recognition, spam filtering, and intrusion detection. The survey aims to motivate the creation of a defense taxonomy, which is not investigated. Martins et al. [20] review the literature on intrusion and malware detection systems that apply adversarial machine learning concepts. They explore the adversarial attacks and defensive solutions and discuss the application of these techniques to intrusion and malware detection scenarios. Apruzzese et al. [21] proposed a taxonomy to model real capabilities and circumstances required by an adversary to launch a successful adversarial attack against ML-based Network Intrusion Detection Systems (NIDSs). The taxonomy is based on five elements on which the adversary has power over the target system: training data, feature set, detection model, oracle feedback, and manipulation depth. The authors aim to guide researchers in devising threat models that reflect realistic research on adversarial attacks against ML-based NIDSs. The above-mentioned surveys review the AML literature on traditional IT environments rather than IoT environments, and hence, the IoT literature is not covered. Conventional methods may not always be an option for IoT environments with a sheer amount of data and limited computing and storage [5,22]. Thus, exploring the applications of AML methods in IoT literature can reveal in what ways the strategies of adversarial attack and defense in traditional IT environments and IoT environments may differ.

Li et al. [23] discuss the AML literature on sensor-based CPSs. The paper focuses on CPSs applications beyond computer vision, including surveillance sensor data, audio data, and textual data. The authors first describe the general workflow of adversarial attacks in CSPs. Then, they cover the existing works of recent adversarial attacks against CPSs and potential defenses that can be performed in CPSs. The work presented by Li et al. based solely on CPSs and cannot be hence generalized for all IoT environments. In other words, although the CPSs do overlap with IoT environments, differences do exist, and they tend to be considered two different paradigms [24,25]. In addition, the surveyed papers are limited to only three application domains.

Yulei [26] focuses on robust learning of ML-based systems in IoT environments. Their survey explores the existing research that discusses the IoT-related data issues and their impact on the learning process during the training phase. The robustness of ML-based systems is discussed from two perspectives. First, when the training data has noises. Second, when IoT devices are compromised, and adversarial examples exist in the training set. The discussion of AML in the IoT environments is only a part of the survey, focusing only on the learning phase. There are other key stage where adversarial attacks can happen, such as inference phase, which highlight the need for a comprehensive survey where more potential attack points can be covered. In addition, the concept of AML is looked at through a different lens, i.e., measuring the reliability of the ML model rather than attack and defense methods and adversarial examples' generation methods.

Only a few published surveys have considered the AML literature in the IoT setting and have not primarily focused on insider threats against ML-based systems. As a result, a comprehensive survey that explores the susceptibility of the ML-based systems in IoT for insider threats and the corresponding countermeasures is still lacking, but it is highly desired. It will facilitate understanding the existing attack landscape and offer a new perspective of breaking down the insider threats in IoT literature into various characteristics to pave the way for follow-up works. Moreover, it will provide a guideline of what defense methods have been proposed so far in the literature and serve as a stepping stone toward improving defense mechanisms against insider threats within the field of AML in IoT environments.

Based on this conclusion, we conduct a comprehensive survey on adversarial insider threats and the potential defenses in IoT environments. To the best of our knowledge, this is the first survey that discusses insider threats against ML-based systems in the context of IoT. The main contributions can be summarized as follows:

- Analyze and taxonomize adversarial attack methods against ML-based systems in IoT from an insider point of view.
- Discuss how these methods can be applied in real-life IoT use cases.
- Provide an overview of defensive strategies that can withstand these attacks in IoT.

3. Methodology

For this review, we defined a search methodology for selecting the relevant literature to ensure the coverage of the most relevant studies in the intersection of AML and IoT literature from an insider point of view. The method of Kitchenham and Brereton [27] was generally followed (not all the steps) as a guideline for the searching and selecting process. The rationale for selecting this methodology was that it follows a uniform protocol comprising structured steps, which is efficient in analyzing published papers in a particular research area. Moreover, hundreds of citations indicate its successful adoption in literature searching related studies.

The search process as divided into the two following steps: automatic search and manual search. These steps are described below.

3.1. Automatic search

Automatic search is considered an initial search step. The key objective of the step is to have a broad overview of the available literature without any bias favoring a specific publisher. It was performed using a search engine, Google Scholar, with a defined list of keywords. The root search words stemmed from the research's main dimensions: They were "adversarial" and "IoT". To ensure that we would not miss papers that discuss specific areas and applications, the keyword "adversarial" was expanded to its variations, for example, poisoning and evasion keywords [13,28]. Similarly, the keyword "IoT" was expanded to well-known IoT applications, such as smart grid, wearable devices, and IIoT.

It is worth noting that the automatic search process can lead to many thousands of results such as 22,100 results for "Adversarial" AND "IoT" query. Some of these papers are loosely related which makes the analysis process difficult. To reduce the scope, the advanced search in Google Scholar is used to limit the appearance of the keywords to be in the title only.

3.2. Manual search

The manual search was conducted using the list of publishers resulting from the automatic search and by applying forward and backward snowballing. Libraries such as Scopus, IEEE Xplore, Springer, ACM, and ScienceDirect and a list of journals and conferences were searched manually to look for relevant literature. The resulting papers were collocated with the papers from the automatic search. A filtering process has been undertaken based on the paper's abstract to limit the scope to the relevant papers. Then, forward snowballing (cited-by search) and backward snowballing (paper's reference list search) have been applied.

The filtered results from automatic and manual searches represent the final bodies of the studied literature. These papers will be passed through a stricter relevance criterion, where the paper is fully read and analyzed. The paper will be included if inclusion criteria are met. The inclusion criteria for a research paper were: written in English, have a full version (not only a poster or abstract), discuss adversarial attack methods or defensive solutions that deployed in IoT ecosystem in a white-box setting, insider threat setting. Fig. 1 illustrates the fundamental steps included in our search methodology.

4. Background

This section outlines the preliminary knowledge of the main concepts used in this work.

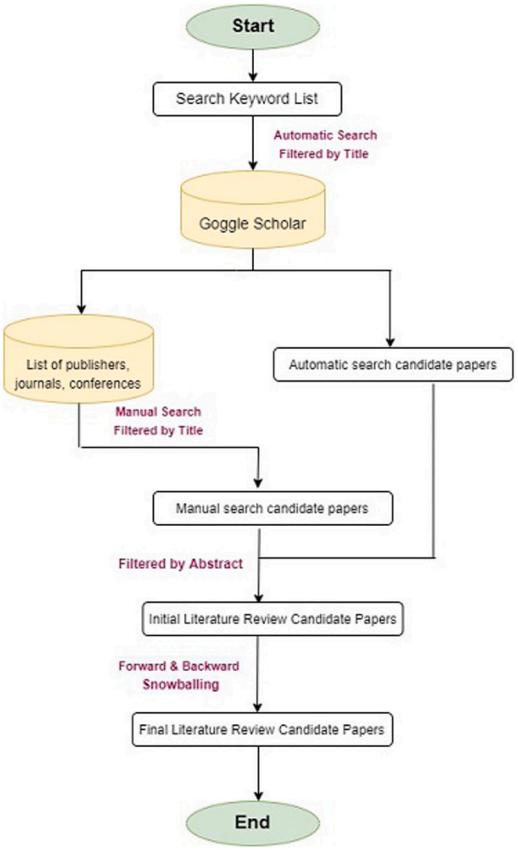

Fig. 1. Research methodology.

4.1. AML overview

AML is a research field that studies the design of a robust ML model that can withstand adversarial opponents [13,15]. It discusses the various adversarial threats and their defensive strategies related to the use of ML models in artificial intelligent-based systems [28]. Moreover, it studies the capabilities and limitations of adversaries under a successful attack scenario [13]. In 2006, Barreno et al. presented an initial attempt to classify attacks against ML-based systems and also provided an overview of a variety of defenses against these attacks [29]. The work presented by Barreno et al. was improved in 2011 by Huang et al. [13] where the concept of AML was formally introduced. Moreover, a taxonomy of AML attacks was proposed. It is worth noting that most of the papers presented in the AML and IoT literature adopted Huang et al. taxonomy [13].

According to Huang et al. taxonomy, an adversarial attack can be studied based on four primary dimensions. Firstly, the **influence dimension** reflects the adversary's capability over the target system. It can be either causative (can tamper with the training dataset) or exploratory (does not influence the training dataset but can manipulate the model during the test or inference phase). Secondly, the **specificity dimension** draws the broadness of the adversarial attack manipulations. These can be targeted attacks (generate adversarial samples to be misclassified into a targeted label/group of labels) or indiscriminate attacks (generate adversarial samples to be misclassified into any label other than its original label). Thirdly, the **knowledge dimension** describes the adversary's level of knowledge about the target system (i.e., training data, feature set, and model configuration). The adversary can have full (white-box), partial (grey-box), or no (black-box) knowledge about the target system. Lastly, the **adversary goal dimension** refers to which security property the adversary aims

Table 1
Dimensions of adversarial attacks analysis.

Influence	Causative attack	Alter the training process through influence over the training data
	Exploratory attack	Evade the detection after the deployment
Specificity	Targeted attack	Focus on a single or small set of target samples
	Indiscriminate	Involve a very general class of points.
Impact	Confidence reduction	Reduce the ML model prediction confidence
	Misclassification	Mislead the ML model response in any way possible
Knowledge	White-box	Adversary has full knowledge about the target system
	Grey-box	Adversary has partial knowledge about the target system
	Black-box	Adversary has zero knowledge about the target system
Goal	Integrity violation	Result in malicious points being classified as normal
	Availability violation	Result in many errors that the system being unusable
	Privacy violation	Result in obtaining private information from the ML model

to violate. These can be integrity violation (where the adversary aims to violate the ML-model integrity by classifying adversarial samples as benign), availability violation (where the adversary aims to increase misclassification errors rendering the system useless), and privacy violation (where the adversary aims to obtain private information by probing the ML-based system). Papernot et al. [30] presented an additional dimension: the **impact dimension** of an adversarial attack refers to the effect that results from a successful adversarial attack. These can be confidence reduction impact (which happens when the adversary successfully manipulates the training data in a way that can corrupt the decision process of the ML-based system) or miss-classifications impact (which happens when the adversary successfully fools the ML-based system and hence, adversarial sample is misclassified).

In order to facilitate understanding of the upcoming sections the five dimensions are summarized in Table 1. The first three dimensions describe the attack characteristics, while the last two dimensions describe the adversary's goal and level of knowledge.

4.2. IoT overview

As discussed in Section 1, IoT describes a network of physical objects, so-called things. Currently, there is no universal consensus on the IoT architecture. However, based on the reviewed literature, the standard IoT architecture has four layers, namely, the perception, network, middleware and application layer [31–35]. Fig. 2 depicts an outline of the typical IoT architecture. A detailed description of each layer is given below.

- **Perception layer:** it is the bottom layer of IoT architecture. It is also known as the physical or sensing layer because it primarily deals with physical IoT sensors and actuators. The primary function of this layer is to collect data from the real world through various sensors.
- **Network layer:** it serves as a communication channel to transmit data collected in the perception layer either to other connected things or to a computational unit for processing and vice versa. It uses a variety of communication protocols, such as Wi-Fi and IPv6.
- **Middleware layer:** it is located between the network and the application layers. It is also known as the processing layer. The key function of this layer is to provide a diverse set of services to the lower layers. It stores, analyzes, and processes vast amounts of data that comes from the network layer. The middleware layer is the candidate target of adversarial attacks in the IoT ecosystem because it is where the ML model usually resides.
- **Application layer:** it is the highest layer within the IoT architecture. The main goal of the application layer is to provide different application services and user interface (UI) to the end-users. It defines various applications for IoT deployment, such as environmental monitoring, smart grid, and smart healthcare.

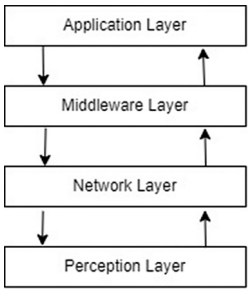

Fig. 2. Layers of IoT architecture.

5. AML and insider threats in IoT

Undoubtedly and as discussed in Section 1, ML has been considered a promising approach for the IoT. It has been adopted in a plethora of applications, representing one of the most widely used computational paradigms in the IoT architecture, specifically in the middleware layer. Given the proven vulnerability of ML, *does adding the ML approach to an IoT ecosystem introduce a vulnerability to adversarial threats that a malicious insider can exploit?* In fact, the vulnerabilities of ML to adversarial threats stem from an assumption made when the ML methods were developed. It was assumed that both training and testing data have identical statistical characteristics, and ML models will be trained and used under a benign environment [17]. Nevertheless, this is not always the case, especially in the IoT ecosystem. To demonstrate, IoT architecture comes with unique design challenges compared to traditional systems that may not fully meet the above-mentioned assumption [4,34,36]. Firstly, IoT devices generate a sheer volume of real-time data in the perception layer from various data types and are expected to be the most significant source of new data in the future [4]. Secondly, IoT deals with heterogeneous communications in the network layer where various devices with different capabilities communicate within the network. Thirdly, most IoT device manufacturers are trying to be 'first to market' by introducing new smart applications, which in turn may compromise some of the security requirements [37]. Finally, IoT networks expand rapidly with more devices being added continually, resulting in a massive scale deployment [5,6]. As a result, these challenges can contribute to multiple vulnerabilities in each layer of the IoT architecture. Surveys on vulnerabilities of IoT layers can be found in [32,33,35]. These vulnerabilities can facilitate access to ML-based systems residing in the IoT ecosystem and hence can make the adversarial perturbations easier compared to perturbating the ML-based system residing in traditional systems, which are more secure against malicious access. Moreover, IoT applications tend to focus more on the open-loop platform, i.e., interconnecting all the things in the physical world. While CPS applications tend to focus more on the closed-loop platform, i.e., sensing information and feedback loop to control the physical world [38]. The open loop platform can result

in larger adversarial attack entry points in IoT compared to CSP and traditional systems.

In addition, the original assumption does not consider that, in some cases, the ML model training environment may not be benign too. In other words, data collection in some IoT applications can be done unsupervised, such as in [39], giving the adversary direct access to the training data. Therefore, a capable insider with an advantage over a malicious outsider, such as being familiar with the security protocols and systems, knowing the ML model configuration, or/and the training data set, can exploit vulnerabilities presented in different IoT layers to access and consequently manipulate the ML model. For example, a malicious insider can exploit some of the vulnerabilities presented in the perception layer to manipulate a small fraction of the training data and thus mislead the ML learning process (causative attack).

We conclude that due to the inherently insecure nature of IoT architecture layers and the mutual communication between the middleware layer, where the ML model is usually located, and other layers, deploying ML models within the IoT ecosystem without considering their susceptibility to adversarial attacks introduces a vulnerability that a malicious insider can exploit. Examples of adversarial attacks that can be exploited by an insider are discussed in Sections 8 and 9.

In what follows, we review the AML literature in IoT from insider threats point of view. The filtered papers resulted from Section 3 will be discussed. This includes adversarial attacks methods and their applications, use cases, and defensive solutions.

5.1. Threat model

To define the scope of our review, we consider an insider threat scenario where a malicious insider attempts to harm an organization without being detected. The organization deploys an IoT platform, and only authorized personnel can access it. The malicious insider aims to lunch an adversarial attack on the ML model located on the middleware layer of the IoT platform.

Inspired by the adversarial dimensions presented in Table 1, we re-defined the scenario mentioned above according to the adversarial taxonomy dimensions. We assume the adversary knowledge is white-box or grey-box; that is the adversary can have complete or partial knowledge about the target ML model, including data sets or/and ML model configuration, a malicious insider scenario. Moreover, the adversary can aim to manipulate the target ML model's integrity, availability, or privacy. With regard to the dimensions related to the attack characteristics, namely influence, specificity, and impact, we consider all the possibilities presented in the explored literature because they are all applicable in the case of insider threat scenarios. For example, the malicious insider can lunch causative (training) or/and exploratory (inference) attack and the attack can be either targeted or indiscriminate. The adversarial dimensions of the reviewed IoT papers are presented in Table 2.

5.2. Adversarial attacks methods in IoT

It is worth mentioning that adversarial attack methods are data-nature dependent. Methods used in some domains, like image processing, cannot be applied directly to other domains, like voice recognition. Thus, the question is, *what are the adversarial methods that can be used by a malicious insider to lunch a successful attack against ML based systems in IoT?*

Based on the defined threat model, we reviewed the IoT literature to elaborate on the various adversarial attack methods against the ML based systems. The adversarial attack methods can be organized into two main categories based on the phase where an attack can be launched: training attack methods and inference attack methods [13].

- **Training attack (causative attack):** an adversarial attack conducted during the training phase. It assumes that the adversary compromises the learning process and can manipulate the training data set. The adversary can either aim to degrade the ML model's overall performance, resulting in a denial of services, or only target specific training samples [28].
- **Inference attack (exploratory attack):** an adversarial attack conducted during the testing or inference phase. The adversary aims to construct an Adversarial Example (AE) by adding some crafted perturbation to a benign data point. The constructed AE can then force the ML model to misclassify it to any class other than its original class. The miss-classification can result in bypassing a cybersecurity defense to deliver an exploit or other form of cyber-attacks without detection.

5.2.1. Training attack methods

Below, we describe the training attack methods that have been discussed in IoT literature:

- **Label flipping method:** it is a subclass of the causative attack methods, in which the adversary can influence the labels assigned to a fraction of the training dataset [40]. There are two different methods for poisoning the training dataset through label flipping: random and targeted label flips [40]. In the random flipping, the adversary randomly selects a fraction from the training samples and flips their labels, whereas in the targeted label flipping, given a number of allowed label flips the adversary's main objective is to find the combination of label flips that maximizes the classification error on the testing data.
- **Gradient ascent method:** Biggio et al. constructed a causative attack that significantly decreases the Support Vector Machine's (SVM) classification accuracy [41]. The attack's goal is to find, using the gradient ascent method, a specially crafted data point whose addition to the training data set can manipulate the optimal solution reached by the SVM.

5.2.2. Inference attack methods

The Inference attack methods presented in IoT literature can be further categorized into three main subcategories based on the method used to generate the AEs which are gradient-based, optimization-based, and graph-based attack methods. First, the gradient-based attack methods:

- **Fast Gradient Sign Method (FGSM):** It is the simplest and the most widely used gradient-based attack [42]. It generates AEs by performing a one-step gradient update of the model loss function with respect to every input in the dataset. Then, the gradient sign is computed to indicate the perturbation direction. The signed gradient adjusts the original input x by adding a specified perturbation. The strength of FGSM depends on perturbation size. If perturbation size is too small, then y(x′) might not differ from y(x). However,it is crucial to consider the actual range (x_{min}, x_{max}) of each feature in the dataset.
- **Basic Iteration Method (BIM):** It is an iterative version of FGSM attack [43]. The AEs are created by applying FGSM iteratively with a small step size instead of a single large step. Although BIM is slower and consume a higher computation cost compered to FGSM, it generates a stronger attacks with subtle perturbations [43].
- **Momentum Iterative Method (MIM):** it introduces an additional momentum term to BIM; momentum is a technique to accelerate and stabilize stochastic gradient descent algorithm [44]. By incorporating the momentum term into the iterative process for AEs generation, the MIM can stabilize update directions and escape from poor local maxima during the iterations. Therefore, the MIM method results in more transferable AEs compared to BIM [44].

- **Projected Gradient Descent (PGD):** This is a variant of the FGSM like the BIM [45]. The main difference from BIM resides in the fact that PGD initializes the search for an AE at a random point, then runs several iterations of the BIM attack to find the AE [46]. Due to the randomization, PGD regarded as the strongest first-order attack compared to FSGM and BIM [45,46].
- **Jacobian Saliency Map Attack (JSMA:)** it was proposed for crafting AEs that can fool the Deep Neural Networks (DNNs) [30]. Compared to the previous methods, JSMA reduces the perturbations by controlling the number of perturbated features rather than the perturbation size itself. In doing so, JSMA uses saliency map to select features that should be modified; feature that will not increase the probability of the target label or will not decrease the probabilities over all other labels is rejected. The perturbation reduction comes with significant computational cost [30].
- **DeepFool:** it is a gradient-based attack proposed by [47] that aims to efficiently compute the adversarial perturbations that fool DNNs in the untargeted setting. When compared to FGSM and JSMA, DeepFool produced less perturbation [48]. Moreover, DeepFool reduce the size of the perturabtion rather than the number of selected features, as JSMA does [48].

Second, the optimization based inference attack methods can be summarized as follows:

- **L-BFGS Method:** it generates AEs based on L-BFGS optimization algorithm. The L-BFGS attack has two key differences from the FSGM method [49]. First, it is optimized for the L_2 distance metric rather than L_∞ used in FSGM, and second, it is designed to produce very close AEs instead of being fast.
- **Carlini & Wagner (C&W) Method:** Carlini and Wagner proposed a targeted attack to defeat *defensive distillation*, an adversarial defense method, by extending L-BFGS method [49]. The authors discussed three kind of attacks based on different distance metrics: C&W L_0, C&W L_2, and C&W L_∞ attacks.
- **Elastic Net (EAD) method:** The EAD method proposed by [50] as a modification of the C&W attack and aimed to generalize the C&W attack by exploring L1 based adversarial attacks. The EAD targets the DNN systems and is based on elastic-net regularization to craft L1 oriented AEs different from existing attack methods.

Finally, the graph-based inference attack method can be summarized as follows:

- **Graph Embedding and Augmentation Attack (GEA) method:** GEA is an AE's generation method proposed by [51] to manipulate the graphical representation of an IoT software, namely Control Flow Graph (CFG). The GEA generates an adversarial IoT software through combining the original graph with an adversarial target graph using shared entry and exit nodes while maintaining the practicality and functionality of the attacked sample. The GEA targets a DL-based model that is trained over CFG features. In contrast to the above-mentioned adversarial methods, GEA manipulates the CFG rather than the features extracted from the CFG.

6. Applications of adversarial insider attack methods in IoT

In the previous sections, namely Sections 5.2.1 and 5.2.2, we reviewed the adversarial attack methods that a malicious insider can use against a target ML model in IoT ecosystem. This section explores different application domains of these methods. Indeed, this section mainly focuses on two questions: first, *based on the reviewed literature, what domains in IoT applications can be susceptible to adversarial insider attack methods?* Second, *how these attacks are generated in those domains?* Table 2 summarizes the reviewed applications based on adversarial dimensions mentioned in Table 1.

The applications of adversarial insider attack methods during the training phase are presented first, followed by the inference phase applications.

6.1. Training attack methods applications

6.1.1. Environmental sensing

Baracaldo et al. [59] further enhanced in [39] showed that the poisoning attack could be successful in a scenario where several IoT devices contribute data points used to train ML-based system. For example, an environmental government regulator installs IoT sensors around each factory to regulate factory emissions. A polluting factory (insider) tampers the collected data in ways that will violate the integrity of the trained ML. To craft poisonous data, they used two different methods: [41,58]. Both methods target the SVM. The first method focuses on a label flipping attack [58], while the second method uses the gradient ascent attack [41]. The adversary's ultimate goal is to reduce the accuracy of the poisoned SVM model.

6.2. Inference attack methods applications

6.2.1. Smart home

Anthi et al. [52] have explored the vulnerability of ML-based Intrusion Detection System (IDS) to adversarial inference attack, an exploratory attack. They proposed a rule-based approach to generate indiscriminate AEs that target a range of pre-trained supervised ML models, namely Decision Tree(DT), Random Forest (RF), Naive Bayes (NB), and SVM, used to detect Denial of Service (DoS) attack in an IoT smart home network. It was assumed that the adversary successfully retrieved the password for the central access point within the smart home network; the adversary may have a pre-existing relationship with the victim. Subsequently, the adversary can scan the network and lunch different attacks. For adversarial inference attacks, the adversary mainly focuses on identifying the most important features that best discriminate between the malicious and benign packets. Then, manually perturbing the values of these features forces the IDS to misclassify the incoming packet. The experimental results showed that all ML models' performance was affected, decreasing a maximum of 47.2% when the adversarial packets were present.

6.2.2. Healthcare

Away from the defensive DL-based systems, Rahman et al. [54] showed that the DL-based diagnostic model that relies on medical IoT could be vulnerable to adversarial inference attacks, exploratory attacks. They tested six DNN-based COVID-19 applications against different adversarial methods, including FGSM, MI-FGSM, Deepfool, L-BFGS, C&W, BIM, Foolbox, PGD, and JSMA. Given that the adversary has complete knowledge of each DNN model, open source libraries, including PyTorch, Tensorflow, and Keras, were used to design the targeted and indiscriminate AEs that aim to violate the integrity of the DNN-based model. It was concluded that DL-based systems that do not consider defensive measures against adversarial perturbations remain vulnerable to adversarial inference attacks.

6.2.3. Malware detection

Abusnaina et al. [51] studied the robustness of malware detection systems against adversarial inference attacks, particularly those trained over CFG features. To do so, first, the CFG of benign and malicious samples were generated, and the CFG-based features were extracted. Second, a Convolutional Neural Network (CNN)-based model that distinguishes IoT malware from IoT benign software was built. Finally, using an imbalanced dataset with 276 and 2281 begin and malicious samples, respectively, two different approaches are designed to generate targeted and indiscriminate AEs. The first approach is based on off-the-shelf adversarial methods like FSGM, PGD, MIM, JSMA, C&W, DeepFool, and ElasticNet. The second approach is GEA which manipulates the CFG itself instead of the extracted features. The adversary's ultimate goal is to violate the integrity of the ML-based malware detection model; in such a manner a malware can be delivered to the target IoT system successfully. The findings showed

Table 2
Applications of adversarial insider attack methods in IoT.

Ref.	Domain	AE Method	Influence	Specificity	Impact	Knowledge	Goal	Attacked ML	Dataset
[52]	Smart home	Rule-based	Exploratory	indiscriminate	Misclassify	White-box	Integrity	DT, RF, NB, SVM	[53]
[54]	Healthcare	Gradient-based, L-BFGS, C&W,	Exploratory	Targeted and indiscriminate	Misclassify		Integrity	DNN	[54]
[51]	Malware	Gradient and optimization based, GEA	Exploratory	Targeted and indiscriminate	Misclassify		Integrity	CNN	[55]
[56]	Malware	Software transplantation	Exploratory	Targeted	Misclassify		Integrity	SVM	[57]
[39]	Environmental sensing	Label flipping & gradient ascent	Causative	Targeted	Confidence reduction		Integrity	SVM	[41,58]

that the first approach achieved 100% misclassification rate while the second approach misclassified all malware samples, six samples, as benign ones. Similarly, Pierazzi et al. [56] propose a novel problem-space adversarial attack, where the adversarial manipulations target the software object rather than the features vectors, to generate an Android malware. The attack aims to evade the static analysis detection at test time (exploratory attack) without relying on code obfuscation as it may increase the suspiciousness of the generated malware. They assume the attacker has perfect knowledge about the target system, which can be an insider scenario. They use the automated software transplantation concept to generate a targeted (malware classified as benign) adversarial sample. It extracts slices of bytecode from benign donor Android software and injects them into actual malware until the detection system misclassifies it as benign. To do so, they defined a set of constraints on the software transplantations process, including available transformations, methods to preserve generated malware semantics, its robustness to software analysis techniques, and its plausibility (i.e., resembles an actual, functioning Android software). Using Android applications collected from AndroZoo [57], Pierazzi et al. showed that it is practically feasible to generate Android malware in problem-space that were able to evade the SVM-based Android malware classifier DREBIN [60] and its hardened variant, Sec-SVM [61].

7. Taxonomy of adversarial insider attacks in IoT

Based on the reviewed literature and inspired by concepts presented by Huang et al. [13] and Apruzzes et al. [21], we propose a taxonomy to model adversarial insider attacks in IoT ecosystems as depicted in Fig. 3. The taxonomy aims to position adversarial attacks in the context of the IoT ecosystem and raise awareness of potential adversarial insider threats. It breaks down the adversarial attack into various characteristics to facilitate the evaluation of different attack scenarios. For example, one scenario can be an exploratory attack with full knowledge power, while the other can be an exploratory attack with partial knowledge power. Eventually, the identified attack scenarios can serve as a stepping stone for cybersecurity analysts toward what defense methods should be implemented.

As shown in Fig. 3, the taxonomy is built upon three main bases: the adversary (malicious insider), adversarial attack characteristics, and layers of the IoT architecture, each of which is expanded to different characteristics. Using the taxonomy, the first step to build an attack scenario is to examine the different layers of the IoT ecosystem. Then, with reference to Fig. 3, from left to right, the flow of a potential attack scenario can be built by answering the following questions: *Is the layer inside the organization boundary?* if yes, then we move to the other questions; otherwise, we skip the layer because it is out of our taxonomy scope. *Does the layer interact with the ML-based system?* if yes, then we can identify the potential scenarios as follows:

- *What power a malicious insider can have at this stage?*
- *What goal a malicious insider can achieve at this stage?*
- *Based on the insider power, What is/are the candidate adversarial attack(s) at this stage?*
- *What IoT layer can be used by a malicious insider as an entry point to manipulate the ML-based system at this stage?*
- *Can a malicious insider lunch targeted or/and indiscriminate adversarial attack at this stage?*
- *What is/are the potential impact(s) of successful candidate attack(s) identified in question three?*

Given a determined insider's goal and power, the attack characteristics can be determined. For example, selecting "write to training data set" power results in a possible training phase or so-called causative attack.

The concept of adversary power was introduced by Apruzzes et al. [21] to model realistic adversary capabilities. They defined the adversary power based on five elements on which the adversary has power over the target system: training data, feature set, detection model, oracle, and manipulation depth. Depending on the element, the power level can be defined. In the training data set element, the adversary power can be read, write, or has no access to the data set. Regarding the feature set element, which is features used by the ML-based model to perform its decision, and the detection model element, which is a trained model used to perform detection, the adversary power is knowledge about these elements and it can be full, partial, or zero knowledge. In the oracle element, the adversary power is feedback, i.e., the possibility of obtaining feedback from the ML-based model by observing the relationship between changes in inputs and outputs. The feedback can facilitate launching the intended adversarial attack and can be limited, unlimited, or absent. Finally, the manipulation depth element refers to the nature of adversarial perturbation. The adversarial perturbation can be at problem space, perturbing raw data input, or at feature space, perturbing input data after it has been transformed into features representation. Generally, the more power the adversary has, the more likely the attack will be successful; however, it can be unrealistic.

In modeling adversarial insider attacks on the IoT ecosystem, insiders tend to have more control "power" on these elements than malicious outsiders. The reason is that insiders already have access to the organization's systems, networks, and data, and with vulnerable IoT devices being around, they could take advantage of them to harm the organization's ML systems [62,63].

In what follows, we present two use cases to illustrate how the proposed taxonomy can be used in real-life scenarios. The first use case is considered a representative use case of the industrial IoT sector, and the second use case represents an example of everyday organizational activities.

8. First use case: Chatty factory

This section elaborates on how the proposed taxonomy can be applied to a real-world case. Our main objective is to increase awareness of potential insider threats against the ML-based system by showing how our taxonomy can be used to determine potential insider adversarial threats.

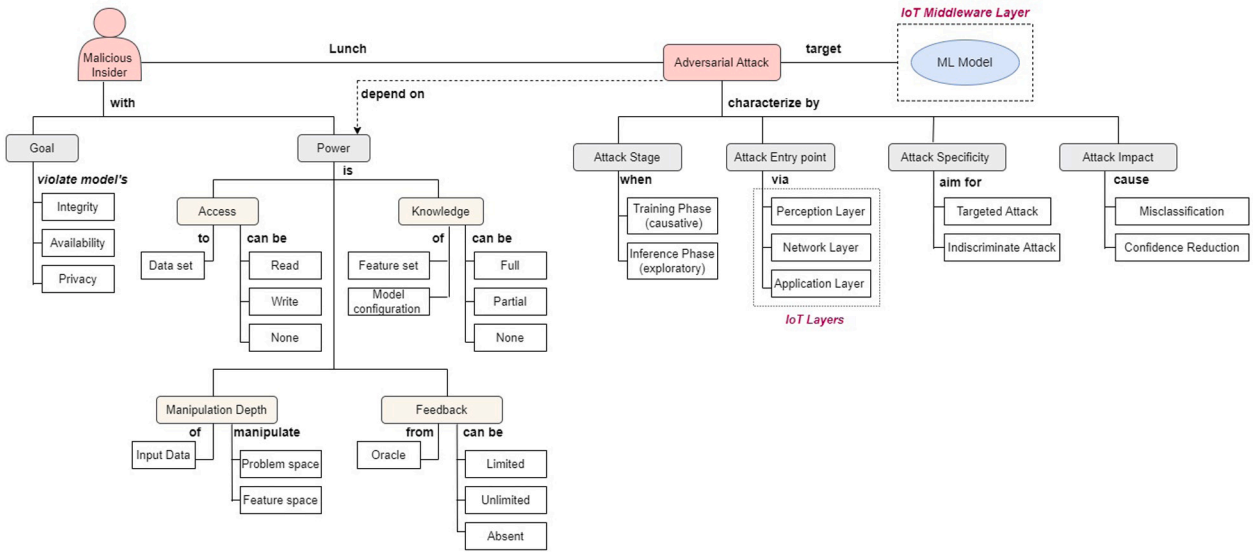

Fig. 3. Taxonomy of adversarial insider attacks in IoT.

8.1. Chatty factory description

Chatty Factory [64] is a three-year investment by the Engineering and Physical Sciences Research Council (EPSRC) through its program for New Industrial Systems. The chatty factory concept explores the increasingly significant role of the IoT in the manufacturing sector. In other words, the chatty factory concept investigates the potential of placing data collected from IoT sensors at the core of design and manufacturing processes. The project focuses on how consumers' collected sensor data that represent real-time use might be immediately transferred into usable information for the benefit of design and manufacturing processes. Further information about the project can be found in [64]. In what follows, we discuss how the proposed taxonomy in Fig. 3 can be applied to the chatty factory case.

8.2. Chatty factory architecture

In order to understand the potential insider threats, it is imperative to get a good understanding of the chatty factory architecture. As depicted in Fig. 4, the chatty factory consists of different blocks [64, 65]:

- **Product data block**: is the starting point of the chatty factory. The factory products are embedded with various IoT sensors that collect live data from the wild about their use and environment.
- **Data annotation block** : is where the collected data from the IoT sensors is annotated by a semi-automated process; where the data is partly annotated by human and partly by ML tools.
- **Product use model block**: is where the processed data from the previous block and their labels are used to train ML models to automatically identify points of interest in the collected data.
- **New form of design block**: is where the insights obtained from the analyzed data can be used to support enhancements to existing products and/or the creation of a totally new product.
- **Rapid product manufacture**: is where the new production instruction is mapped onto a digital twin of the product design; so the design tweaks can be made.

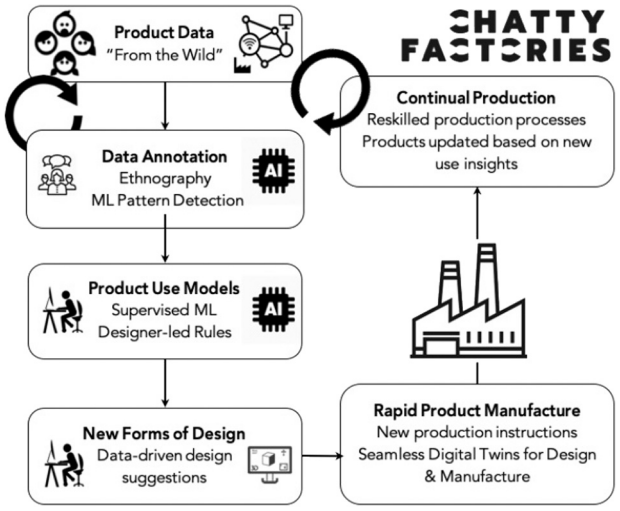

Fig. 4. Chatty factory vision.
Source: Adopted from [64].

8.3. Potential insider threats against chatty factory

Cyber security is a crucial concern when it comes to adopting technological innovations, namely the IoT. Opening the factory floor to the IoT will capitalize on cyber-attack surfaces and introduce new vulnerabilities for adversaries [64]. Despite different cyber-attacks that could be launched against the chatty factory, this section focuses on adversarial attacks against ML-based systems from a malicious insider point of view.

Based on the chatty factory architecture depicted in Fig. 4, we can observe that the IoT perception layer is outside the chatty factory yet connected to the chatty factory boundary. In contrast, the other IoT layers are inside the chatty factory boundary. Fig. 5 shows how the IoT layers are distributed in chatty factory architecture. In what follows, we identify potential attack scenarios in the chatty factory:

- **Product data block**: this stage is where the IoT perception layer is located. It mainly deals with the wild (outside the chatty factory boundary). Thus, identifying the possible attacks in this stage is outside our review scope.
- **Data annotation block** : the input to this stage is the collected IoT sensors data, and the output is the data set(s) used to train the ML-based system in the next stage. This stage represents the initial phase of ML-based model(s) building and has an interaction

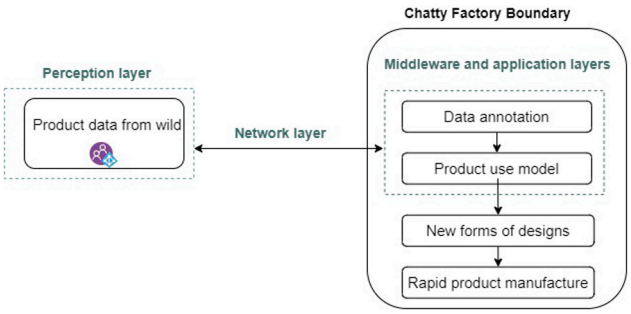

Fig. 5. IoT layers in chatty factory.

with the ML-based system. As mentioned earlier, collected data is partially annotated by humans(insiders), which means that they have direct access to the training data set(s). Based on the answers presented in Table 3, we can conclude that a malicious insider at this stage can exploit a possibility of a causative adversarial attack (can be a label flipping). Depending on the collected data, they can manipulate the data at problem or feature spaces as they have access to raw and transformed data. A malicious insider with a competitive interest can manipulate the annotation process to produce refinement products that may not represent consumers' views, consequently degrading product revenue and consumer satisfaction.

- **Product use model block**: The input to this stage is the processed data from the previous block along with their labels, and the output is insights into the new design form. This stage is where the ML models are trained. Based on the stage's functionality, the malicious insider can exploit a possibility of a causative attack as in the previous stage, however, with an advantage of knowledge about the ML-based model configuration. This, in turn, can help in launching more sophisticated causative attacks. It is worth mentioning that insiders may only manipulate data at feature space as they usually do not have access to raw data at this stage. They can launch targeted and indiscriminate adversarial causative attacks to harm the chatty factory vision.
- **New form of design block**: The input to this stage is the drawn design insights, and the output is the suggested enhancement. This stage has does not interact with IoT ML-based systems; thus, the potential cyber threats are outside our taxonomy scope.
- **Rapid product manufacture**: The input to this stage is new production instruction, and the output is the digital twin of the product design. This stage has does not interact with IoT ML-based systems; thus, the potential cyber threats are outside our taxonomy scope.

Table 3 provides a summary of the above-discussed scenarios based on the taxonomy questions.

9. Second use case: Facial recognition system

One of the most common IoT use cases is Facial Recognition Systems (FRSs). IoT has been applied in FRS in a wide range of applications, including airports, bank lockers, and home and workplace security. Although machine learning algorithms embedded in IoT products improved how FRS works, it opens the doors for new vulnerabilities. With reference to Fig. 3, this section discusses potential adversarial insider attacks in FRSs.

9.1. Facial recognition system description

IoT based FRS, such as [66–69], is a biometric technology that aims to identify individuals by measuring their facial variables and matching

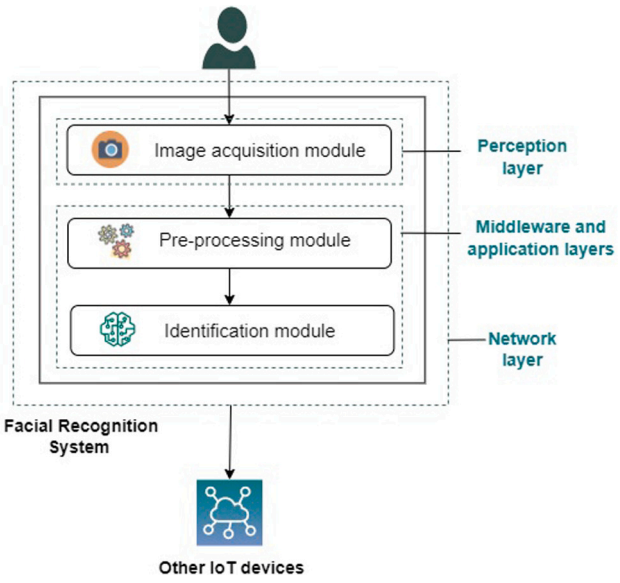

Fig. 6. Facial recognition system architecture.

them with pre-enrolled facial samples. If there is a match, the FRS can approve the individual identity. Then, this identification can be used, for example, to grant an access to a proprietary entity or record an individual attendance. Below, we present the general architecture of IoT-based FRSs followed by the identified adversarial attack scenarios.

9.2. Facial recognition system architecture

As shown in Fig. 6, the facial recognition system generally consists of three basic components:

- **Image acquisition module**: this is considered as the FRS's sensing layer where the system captures the individual face via a webcam. The captured image is then used as an input to the second layer of the FRS.
- **Pre-processing module**: this is considered as a part of the FRS's middleware layer where all the image pre-processing for ML tasks occurs. In this module, the FRS extracts patterns from the captured image, aiming to find the main features for classification.
- **Identification module**: this is considered as a part of the FRS's middleware and application layers where the extracted features are compared to the training dataset samples to classify the individual image as known or unknown. If the individual is known, then depending on the application of the FRS, the identification module can send a command to other IoT devices, such as a smart lock, to unlock a door or grant access to a service.

9.3. Potential insider threats against facial recognition system

Organizations have adopted the FRS for its benefits; However, FRS is susceptible to misuse, creating cybersecurity concerns. While evaluating the FRS architecture for vulnerabilities, we have identified a number of cyber-attacks that can render the system useless. Using the proposed taxonomy questions, we focused on identifying insider attacks that target ML-based systems associated with the FRS. Attacks scenarios are summarized in Table 3.

- **Image acquisition module**: the input to this module is real-time face patterns, and the output is the captured image. The module provides inputs to FRS's ML-based system. A malicious insider with knowledge of how the system works and a victim's identity

Table 3
Taxonomy-based adversarial insider attack scenarios.

Use case	Scenario ID	Stage	Insider power				Insider goal	Attack category	IoT entry point	Attack specificity	Attack impact
			Access	Knowledge	Manipulation depth	Feedback					
Chatty factory	Scenario 1	Data annotation	Write	Feature set: full ML model: none	Both	None	Integrity violation	Training phase	IoT middleware	Both	Misclassification
	Scenario 2	Product use model	Write	Feature set: full ML model: full	Feature space	None	Integrity violation	Training phase	IoT middleware	Both	Misclassification
FRS	Scenario 1	Image acquisition module	Read	Feature set: full or partial ML model: full or partial	Both	None	Integrity violation	Inference phase	IoT sensing	Targeted	Misclassification
	Scenario 2	Identification module	Write	Feature set: none ML model: none	Feature space	None	Integrity violation	Training phase	IoT application	Both	Confidence reduction

information and face image can manipulate the input to fool the ML-based system embedded in the webcam. The malicious insider can present a fabricated face image for the webcam using the virtual camera app [70] or physically manipulated image [71] to evade the model. This can result in targeted misclassification and, consequently, impersonation of the victim with all of its privileges. Depending on the strategy used by the attacker, both problem and feature space perturbations can be applicable at this stage.

- **Pre-processing module**: the input to this module is the captured image and the output is the processed image features. The module acts as a bridge between the first and last modules; hence, no realistic adversarial attack scenarios can be identified at this stage.
- **Identification module**: the input to this module is the image features, and the output is the identification result. When an organization asks its employees to submit an ID photo for the FRS dataset, a malicious insider can exploit the possibility of a training phase attack. A malicious insider with a "write to training dataset" power can provide a poisoned ID photo, feature space perturbations, as a backdoor to give the adversary control over the FRS. Depending on the adversary's goal, this can generate a targeted or indiscriminate impersonation.

The attack identification process is the first step toward defending the ML-based system against adversarial insiders. The proposed taxonomy can be seen as a starting point of what adversarial attacks exist. What entry point can a malicious insider use? At which stage can attacks happen? Given this information, an appropriate security control can be implemented. Section 10 presents different defensive solutions that can be used to improve the robustness of ML-based systems against adversarial insiders.

10. Defense against adversarial insider attacks in IoT

The defense against adversarial attacks remains an open problem. Nevertheless, several promising solutions have been proposed during the last years. Before presenting the reviewed defensive solutions, we would like to discuss *why adopting the IoT applications highlights the need for awareness about defensive solutions against adversarial insider attacks?*

In fact, the inherent nature of IoT devices renders them vulnerable to cyber-attacks. Moreover, intelligence provided by ML-based systems can be manipulated, as proven by recent research, if they are not well protected. Therefore, malicious insiders can use their familiarity and level of access to exploit the IoT devices' vulnerabilities and manipulate the IoT ML-based system at either training or inference phase. In addition, the impact of a successful adversarial attack can be tightly bounded to physical world like in IIoT and smart healthcare applications; thus, influencing the ML-based decision of IoT devices can negatively affect the physical world, including human life. Finally, despite the growing number of cybersecurity-related incidents, many organizations do not know how to secure their system and need guidance. A study conducted across 20 countries involving 3100 IT and business decision-makers as participants showed that 84% of organizations adopting IoT have experienced an IoT-based security breach [72]. A survey on an industrial grade level showed that 25 out of 28 organizations do not know how to secure their ML-based systems and need explicit guidance [73]. Similarly, Gartner's report [74] showed that fewer than one-third of chief information security officers are confident about the reliability of their information systems in assessing and mitigating IoT-related risks.

As a results, raising the awareness of potential adversarial insider attacks is a need. The upcoming sections present in detail the defensive solutions proposed in the IoT literature to answer *how can ML-based systems in IoT environments adapt when an insider adversary actively manipulates the system's inputs?*

Defensive Solutions can be generally divided into two classes [30]. The first class is a proactive solution that aims to improve the ML-based system's robustness during the training phase. The second class is a reactive solution whose main objective is to detect the AEs in real-time during the inference phase. Sections 10.1 and 10.2 present the defensive solution against training and inference attacks discussed in Section 5.2 respectively.

10.1. Defense against training phase attacks

10.1.1. Tamper-free provenance frameworks

Baracaldo et al. [59] further enhanced in [39] proposed a proactive defense that acts as a filter prior to the learning phase to identify and remove poisonous data in IoT systems. They take advantage of provenance data which is meta-data associated with each data point and show information about its creation, origin and derivation [59]. The provenance data is used to segment the training data points that share a provenance signature into groups. Once the training data has been segmented appropriately, each segment is then evaluated by comparing the performance of a model trained on the full data set with a model trained on a data set that excludes that segment. By doing so, the poisonous segment, which degrades the performance of the model trained with that segment, can be identified and removed from the training data set.

10.2. Defense against inference phase attacks

10.2.1. Adversarial training

Adversarial training, in which the ML-based system is trained on a data set containing both the original and adversarial data samples, is one of the proactive defenses against adversarial attacks that withstands strong attacks. Goodfellow et al. [42] demonstrated that adversarial training could result in ML-based system regularization, which in turn improves its efficiency against adversarial attacks.

10.2.2. Defensive distillation

Papernot et al. [75] presented defensive distillation, which defends DNN against adversarial perturbation by leveraging the distillation technique. The defensive distillation proceeds in two main steps. Firstly, it trains a teacher network by setting the temperature parameter T of the soft-max to a significant value resulting in smooth labels. Secondly, it trains a distilled or so-called student network with the same architecture as the teacher network on the smooth labels using the temperature T. Finally, when running the distilled network at the test phase, it sets the temperature T to 1. This, in turn, reduced the effectiveness of AEs and increased the average minimum number of modified features required to create AEs [75]. Nevertheless, It is worth mentioning that research, such as [49,50], showed that the defensive distillation could not withstand C&W and EAD attacks.

10.2.3. Generative Adversarial Network (GAN)

Yumlembam et al. [76] presented a defensive solution that leverages the GAN architecture. GAN consists of two rival networks: the generator (GN) and the discriminator (DN) [77]. On the one hand, The GN's objective is to generate fake data, which tends to be malicious, that is similar to real data, and cannot be detected by DN (which plays the adversary role). On the other hand, the DN aims to distinguish between real and fake data generated by GN (which plays the defender role). Yumlembam et al. [76] showed that retraining an Android malware detection model with the GN samples after labeling them as malware can help harden the detection model against adversarial inference attacks. The authors focus mainly on the robustness of graph-based Android malware detection models that uses Graph Neural Network (GNN) to discriminate malware from benign applications.

10.2.4. Large Margin Cosine Estimation (LMCE)

Wang and Qiao [78] proposed a defensive solution against adversarial inference attacks in IoT. They aim to withstand white and semi-white box attacks, including FGSM, BIM, JSMA, C&W, and MIM, that target DNN. Their approach was based on the intuition that if the neural network layers can provide critical features that distinguish the samples, then adversarial examples can be detected. Therefore, they propose a mathematical model that predicts the degree to which the sample is deviated from the actual sample in the data manifold dimension using the LMCE feature and Kernel Density Estimate (KDE) feature from the neural network layers. The LMCE detects points in low confidence regions, while KDE detects points far from the data manifold. The logistic regression was then used with these two features to generate the mathematical model. Experiments reflected the robustness and the pervasive of the proposed approach.

It is worth mentioning that adversarial defensive solutions were substantially less explored in the IoT ecosystem, although their effectiveness was proven at withstanding adversarial attacks in traditional systems. For example, solutions proposed in [79–81].

11. Conclusion

The wide adoption of IoT has made it a trendy area of scientific research regarding what threats they face and how we can defend against them. In this paper, we review the recent papers that discuss the vulnerability of ML-based models in IoT to adversarial attacks from an insider point of view. Given that the insider can access critical ML-based resources, such as training data sets and model configuration, we propose a taxonomy of the adversarial approaches that the insider can exploit. We categorize adversarial attacks into two main classes: training attacks and inference attacks. Then, we showed how the proposed taxonomy could be used to identify adversarial insider attacks using two representative use cases: the chatty factory and the FRSs. In addition, we review how the malicious insider can apply the adversarial attack methods in real-life IoT applications, including environmental sensing, smart home, healthcare, and malware detection. Finally, we explored the proposed countermeasures for adversarial insider threats in IoT environments.

We can conclude that malicious insiders can deteriorate the performance of ML-based systems residing in the IoT if their vulnerability to adversarial attacks is not considered. To further extend the studies that were analyzed in this review, we suggest the following for future work:

- There is a greater variety of insider attack methods in the inference phase than attack methods in the training phase. Consequently, exploring training attack methods in IoT from a malicious insider point of view is a promising venue for future work.
- The defense strategy against adversarial insider attacks in the IoT ecosystem should include more attention. Many research works have demonstrated the vulnerability of ML-based systems in IoT compared to those which proposed defensive solutions. Therefore, developing a secure ML-based system under adversarial insider settings is a challenge that can be investigated further.
- Developing a unified framework for supervised, unsupervised, and semi-supervised ML-based models that will digest each newly introduced attack or defense in an IoT ecosystem and adapt accordingly can be extremely useful. It will act as a solid knowledge base that will raise awareness of the adversarial insider threats landscape and serve as a stepping stone toward improving defense mechanisms.

CRediT authorship contribution statement

Fatimah Aloraini: Conceptualization, Methodology, Formal analysis, Writing – original draft. **Amir Javed:** Resources, Review and editing, Supervision. **Omer Rana:** Resources, Supervision. **Pete Burnap:** Resources, Supervision.

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.

Data availability

No data was used for the research described in the article.

References

[1] Gould Rick. Architecting a connected future. International Organization for Standard-Ization (ISO); 2019, https://www.iso.org/cms/render/live/en/sites/isoorg/contents/news/2019/01/Ref2361.html (accessed 28 June 2022).

[2] Patel Mark, Shangkuan Jason, Thomas Christopher. What's new with the internet of things. mckinsey & company. McKinsey & Company; 2017, https://www.mckinsey.com/industries/semiconductors/our-insights/whats-new-with-the-internet-of-things (accessed 26 June 2022).

[3] Lasse Lueth Knud. State of the IoT: 2020 12 billion IoT connections, surpassing non-IoT for the first time. IoT analytics. IoT Anal 2020. https://iot-analytics.com/state-of-the-iot-2020-12-billion-iot-connections-surpassing-non-iot-for-the-first-time/ accessed 31 Aug 2022.

[4] Mahdavinejad Mohammad Saeid, Rezvan Mohammadreza, Barekatain Mohammadamin, Adibi Peyman, Barnaghi Payam, Sheth Amit P. Machine learning for internet of things data analysis: A survey. Digit Commun Netw 2018;4(3):161–75. http://dx.doi.org/10.1016/j.dcan.2017.10.002.

[5] Hussain Fatima, Hussain Rasheed, Hassan Syed Ali, Hossain Ekram. Machine learning in IoT security: Current solutions and future challenges. IEEE Commun Surv Tutor 2020;22(3):1686–721. http://dx.doi.org/10.1109/COMST.2020.2986444.

[6] European Network and Information Security Agency (ENISA). Baseline security recommendations for IoT. 2017, https://www.enisa.europa.eu/publications/baseline-security-recommendations-for-iot (accessed 15 July 2022).

[7] Le Duc C, Zincir-Heywood Nur, Heywood Malcolm I. Analyzing data granularity levels for insider threat detection using machine learning. IEEE Trans Netw Serv Manag 2020;17(1):30–44. http://dx.doi.org/10.1109/TNSM.2020.2967721.

[8] Liu Liu, Chen Chao, Zhang Jun, De Vel Olivier, Xiang Yang. Insider threat identification using the simultaneous neural learning of multi-source logs. IEEE Access 2019;7:183162–76. http://dx.doi.org/10.1109/ACCESS.2019.2957055.

[9] Tuor Aaron, Kaplan Samuel, Hutchinson Brian, Nichols Nicole, Robinson Sean. Deep learning for unsupervised insider threat detection in structured cybersecurity data streams. 2017, http://dx.doi.org/10.48550/arXiv.1710.00811, http://arxiv.org/abs/1710.00811.

[10] H Schulze. Insider threat 2018 report. Technical report, NY, USA: CA Technologies; 2018, https://ca-security.inforisktoday.com/whitepapers/insider-threat-2018-report-w-4131 accessed(31 Aug 2022).

[11] Ware Bryan. Insider attacks. Technical report, Haystax Technology; 2017.

[12] Joshi Chaitanya, Aliaga Jesus Rios, Insua David Rios. Insider threat modeling: An adversarial risk analysis approach. IEEE Trans Inf Forensics Secur 2021;16:1131–42. http://dx.doi.org/10.1109/TIFS.2020.3029898.

[13] Huang Ling, Joseph Anthony D, Nelson Blaine, Rubinstein Benjamin IP, Tygar J D. Adversarial machine learning. In: Proceedings of the 4th ACM workshop on Security and artificial intelligence. AISec, NY, USA; 2011, http://dx.doi.org/10.1145/2046684.2046692.

[14] Szegedy Christian, Zaremba Wojciech, Sutskever Ilya, Bruna Joan, Erhan Dumitru, Goodfellow Ian, Fergus Rob. Intriguing properties of neural networks. 2014, http://dx.doi.org/10.48550/arXiv.1312.6199.

[15] Duddu Vasisht. A survey of adversarial machine learning in cyber warfare. Defence Sci J 2018;68(4):356. http://dx.doi.org/10.14429/dsj.68.12371.

[16] de Boer Mario. AI as a target and tool: An attacker's perspective on ML. gartner. Gartner 2019. https://www.gartner.com/en/documents/3939991/ai-as-a-target-and-tool-an-attacker-s-perspective-on-ml (accessed 31 Aug 2022).

[17] Wang Xianmin, Li Jing, Kuang Xiaohui, Tan Yu-an, Li Jin. The security of machine learning in an adversarial setting: A survey. J Parallel Distrib Comput 2019;130:12–23. http://dx.doi.org/10.1016/j.jpdc.2019.03.003.

[18] Biggio Battista, Roli Fabio. Wild patterns: Ten years after the rise of adversarial machine learning. Pattern Recognit 2018;84:317–31. http://dx.doi.org/10.1016/j.patcog.2018.07.023.

[19] Pitropakis Nikolaos, Panaousis Emmanouil, Giannetsos Thanassis, Anastasiadis Eleftherios, Loukas George. A taxonomy and survey of attacks against machine learning. Comp Sci Rev 2019;34:100199. http://dx.doi.org/10.1016/j.cosrev.2019.100199.

[20] Martins Nuno, Cruz José Magalhães, Cruz Tiago, Henriques Abreu Pedro. Adversarial machine learning applied to intrusion and malware scenarios: A systematic review. IEEE Access 2020;8:35403–19. http://dx.doi.org/10.1109/ACCESS.2020.2974752.

[21] Apruzzese Giovanni, Andreolini Mauro, Ferretti Luca, Marchetti Mirco, Colajanni Michele. Modeling realistic adversarial attacks against network intrusion detection systems. Digital Threats Res Pract 2021. http://dx.doi.org/10.1145/3469659.

[22] Elrawy Mohamed Faisal, Awad Ali Ismail, Hamed Hesham FA. Intrusion detection systems for IoT-based smart environments: a survey. J Cloud Comput 2018;7(1):21. http://dx.doi.org/10.1186/s13677-018-0123-6.

[23] Li Jiao, Liu Yang, Chen Tao, Xiao Zhen, Li Zhenjiang, Wang Jianping. Adversarial attacks and defenses on cyber–physical systems: A survey. IEEE Internet Things J 2020;7(6):5103–15. http://dx.doi.org/10.1109/JIOT.2020.2975654.

[24] Greer Christopher, Burns Martin J, Wollman David, Griffor Edward. Cyber-physical systems and internet of things, special publication (NIST SP). National Institute of Standards and Technology; 2019, http://dx.doi.org/10.6028/NIST.SP.1900-202.

[25] Fatima Iqra, Malik Saif UR, Anjum Adeel, Ahmad Naveed. Cyber physical systems and IoT: Architectural practices, interoperability, and transformation. IT Prof 2020;22(3):46–54. http://dx.doi.org/10.1109/MITP.2019.2912604.

[26] Wu Yulei. Robust learning-enabled intelligence for the internet of things: A survey from the perspectives of noisy data and adversarial examples. IEEE Internet Things J 2021;8(12):9568–79. http://dx.doi.org/10.1109/JIOT.2020.3018691.

[27] Kitchenham Barbara, Brereton Pearl. A systematic review of systematic review process research in software engineering. Inf Softw Technol 2013;55(12):2049–75. http://dx.doi.org/10.1016/j.infsof.2013.07.010.

[28] Lin Hsiao-Ying, Biggio Battista. Adversarial machine learning: Attacks from laboratories to the real world. Computer 2021;54(5):56–60. http://dx.doi.org/10.1109/MC.2021.3057686.

[29] Barreno Marco, Nelson Blaine, Sears Russell, Joseph Anthony D, Tygar J D. Can machine learning be secure? In: Proceedings of the 2006 ACM symposium on information, Computer and Communications Security. ASIACCS, NY, USA; 2006, http://dx.doi.org/10.1145/1128817.1128824.

[30] Papernot Nicolas, McDaniel Pat, Jha Somesh, Fredrikson Matt, Celik Z Berkay, Swami Ananthram. The limitations of deep learning in adversarial settings. In: 2016 IEEE european symposium on security and privacy (EuroS&P). 2015, http://dx.doi.org/10.1109/EuroSP.2016.36.

[31] Milenkovic Milan. Internet of things: Concepts and system design. Cham: Springer International Publishing; 2020, http://dx.doi.org/10.1007/978-3-030-41346-0.

[32] Khanam Shapla, Ahmedy Ismail Bin, Idna Idris Mohd Yamani, Jaward Mohamed Hisham, Bin Md Sabri Aznul Qalid. A survey of security challenges, attacks taxonomy and advanced countermeasures in the internet of things. IEEE Access 2020;8:219709–43. http://dx.doi.org/10.1109/ACCESS.2020.3037359.

[33] Hassija Vikas, Chamola Vinay, Saxena Vikas, Jain Divyansh, Goyal Pranav, Sikdar Biplab. A survey on IoT security: Application areas, security threats, and solution architectures. IEEE Access 2019;7:82721–43. http://dx.doi.org/10.1109/ACCESS.2019.2924045.

[34] Chaabouni Nadia, Mosbah Mohamed, Zemmari Akka, Sauvignac Cyrille, Faruki Parvez. Network intrusion detection for IoT security based on learning techniques. IEEE Commun Surv Tutor 2019;21(3):2671–701. http://dx.doi.org/10.1109/COMST.2019.2896380.

[35] Sikder Amit Kumar, Petracca Giuseppe, Aksu Hidayet, Jaeger Trent, Uluagac Selcuk. A survey on sensor-based threats to internet-of-things (IoT) devices and applications. 2018, http://dx.doi.org/10.48550/arXiv.1802.02041, http://arxiv.org/abs/1802.02041.

[36] Luo Zhengping, Zhao Shangqing, Lu Zhuo, Sagduyu Yalin E, Xu Jie. Adversarial machine learning based partial-model attack in IoT. In: Proceedings of the 2nd ACM workshop on wireless security and machine learning. WiseML, NY, USA: Association for Computing Machinery; 2020, p. 13–8. http://dx.doi.org/10.1145/3395352.3402619.

[37] Shaukat Kamran, Alam Talha Mahboob, Hameed Ibrahim A, Khan Wasim Ahmed, Abbas Nadir, Luo Suhuai. A review on security challenges in internet of things (IoT). In: 2021 26th international conference on automation and computing (ICAC). 2021, p. 1–6. http://dx.doi.org/10.23919/ICAC50006.2021.9594183.

[38] Ma Hua-Dong. Internet of things: Objectives and scientific challenges. J Comput Sci Tech 2011;26(6):919–24. http://dx.doi.org/10.1007/s11390-011-1189-5.

[39] Baracaldo Nathalie, Chen Bryant, Ludwig Heiko, Safavi Amir, Zhang Rui. Detecting poisoning attacks on machine learning in IoT environments. In: 2018 IEEE international congress on internet of things (ICIOT). 2018, p. 57–64. http://dx.doi.org/10.1109/ICIOT.2018.00015.

[40] Biggio Battista, Nelson Blaine, Laskov Pavel. Support vector machines under adversarial label noise. In: Proceedings of the asian conference on machine learning. PMLR; 2011, p. 97–112, https://proceedings.mlr.press/v20/biggio11.html (accessed 31 Aug 2022).

[41] Biggio Battista, Nelson Blaine, Laskov Pavel. Poisoning attacks against support vector machines. 2013, http://dx.doi.org/10.48550/arXiv.1206.6389.

[42] Goodfellow Ian J, Shlens Jonathon, Szegedy Christian. Explaining and harnessing adversarial examples. 2015, http://dx.doi.org/10.48550/arXiv.1412.6572.

[43] Kurakin Alexey, Goodfellow Ian J, Bengio Samy. Adversarial examples in the physical world. In: Artificial intelligence safety and security. Chapman and Hall/CRC; 2018.

[44] Dong Yinpeng, Liao Fangzhou, Pang Tianyu, Su Hang, Zhu Jun, Hu Xiaolin, Li Jianguo. Boosting adversarial attacks with momentum. 2018, p. 9185–93.

[45] Madry Aleksander, Makelov Aleksandar, Schmidt Ludwig, Tsipras Dimitris, Vladu Adrian. Towards deep learning models resistant to adversarial attacks. 2019, http://dx.doi.org/10.48550/arXiv.1706.06083.

[46] Kannan Harini, Kurakin Alexey, Goodfellow Ian. Adversarial logit pairing. 2018, http://dx.doi.org/10.48550/arXiv.1803.06373.

[47] Moosavi-Dezfooli Seyed-Mohsen, Fawzi Alhussein, Frossard Pascal. DeepFool: A simple and accurate method to fool deep neural networks. 2016, p. 2574–82.

[48] Yuan Xiaoyong, He Pan, Zhu Qile, Li Xiaolin. Adversarial examples: Attacks and defenses for deep learning. IEEE Trans Neural Netw Learn Syst 2019;30(9):2805–24. http://dx.doi.org/10.1109/TNNLS.2018.2886017.

[49] Carlini Nicholas, Wagner David. Towards evaluating the robustness of neural networks. 2017, http://dx.doi.org/10.48550/arXiv.1608.04644.

[50] Chen Pin-Yu, Sharma Yash, Zhang Huan, Yi Jinfeng, Hsieh Cho-Jui. Ead: elastic-net attacks to deep neural networks via adversarial examples. In: Proceedings of the AAAI conference on artificial intelligence, vol. 32. 2018.

[51] Abusnaina Ahmed, Khormali Aminollah, Alasmary Hisham, Park Jeman, Anwar Afsah, Mohaisen Aziz. Adversarial learning attacks on graph-based IoT malware detection systems. In: 2019 IEEE 39th international conference on distributed computing systems (ICDCS). 2019, p. 1296–305. http://dx.doi.org/10.1109/ICDCS.2019.00130.

[52] Anthi Eirini, Williams Lowri, Javed Amir, Burnap Pete. Hardening machine learning denial of service (DoS) defences against adversarial attacks in IoT smart home networks. Comput Secur 2021;108:102352. http://dx.doi.org/10.1016/j.cose.2021.102352.

[53] Anthi Eirini, Williams Lowri, Słowińska Małgorzata, Theodorakopoulos George, Burnap Pete. A supervised intrusion detection system for smart home IoT devices. IEEE Internet Things J 2019;6(5):9042–53. http://dx.doi.org/10.1109/JIOT.2019.2926365.

[54] Rahman Abdur, Hossain M Shamim, Alrajeh Nabil A, Alsolami Fawaz. Adversarial examples—Security threats to COVID-19 deep learning systems in medical IoT devices. IEEE Internet Things J 2021;8(12):9603–10. http://dx.doi.org/10.1109/JIOT.2020.3013710.

[55] Alasmary Hisham, Anwar Afsah, Park Jeman, Choi Jinchun, Nyang Daehun, Mohaisen Aziz. Graph-based comparison of IoT and android malware. In: Chen Xuemin, Sen Arunabha, Li Wei Wayne, Thai My T, editors. Computational data and social networks. Lecture notes in computer science, Cham: Springer International Publishing; 2018, p. 259–72. http://dx.doi.org/10.1007/978-3-030-04648-4_22.

[56] Pierazzi Fabio, Pendlebury Feargus, Cortellazzi Jacopo, Cavallaro Lorenzo. Intriguing properties of adversarial ML attacks in the problem space. In: 2020 IEEE symposium on security and privacy (SP). 2020, p. 1332–49. http://dx.doi.org/10.1109/SP40000.2020.00073.

[57] Allix Kevin, Bissyandé Tegawendé F, Klein Jacques, Le Traon Yves. AndroZoo: collecting millions of android apps for the research community. In: Proceedings of the 13th international conference on mining software repositories. MSR '16, NY, USA: Association for Computing Machinery; 2016, http://dx.doi.org/10.1145/2901739.2903508.

[58] Zhou Yan, Kantarcioglu Murat, Thuraisingham Bhavani, Xi Bowei. Adversarial support vector machine learning. KDD '12, New York, NY, USA: Association for Computing Machinery; 2012, p. 1059–67. http://dx.doi.org/10.1145/2339530.2339697.

[59] Baracaldo Nathalie, Chen Bryant, Ludwig Heiko, Safavi Jaehoon Amir. Mitigating poisoning attacks on machine learning models: A data provenance based approach. In: Proceedings of the 10th ACM workshop on artificial intelligence and security. NY, USA: Association for Computing Machinery; 2017, p. 103–10. http://dx.doi.org/10.1145/3128572.3140450.

[60] Arp Daniel, Spreitzenbarth Michael, Hübner Malte, Gascon Hugo, Rieck Konrad. DREBIN: Effective and explainable detection of android malware in your pocket. 2014, http://dx.doi.org/10.14722/ndss.2014.23247.

[61] Demontis Ambra, Melis Marco, Biggio Battista, Maiorca Davide, Arp Daniel, Rieck Konrad, Corona Igino, Giacinto Giorgio, Roli Fabio. Yes, machine learning can be more secure! a case study on android malware detection. IEEE Trans Dependable Secure Comput 2019;16(4):711–24. http://dx.doi.org/10.1109/TDSC.2017.2700270.

[62] Kim Aram, Oh Junhyoung, Ryu Jinho, Lee Kyungho. A review of insider threat detection approaches with IoT perspective. IEEE Access 2020;8:78847–67. http://dx.doi.org/10.1109/ACCESS.2020.2990195.

[63] Khan Ahmed Yar, Latif Rabia, Latif Seemab, Tahir Shahzaib, Batool Gohar, Saba Tanzila. Malicious insider attack detection in IoTs using data analytics. IEEE Access 2020;8:11743–53. http://dx.doi.org/10.1109/ACCESS.2019.2959047.

[64] Burnap P, Branson D, Murray-Rust D, Preston J, Richards D, Burnett D, Edwards N, Firth R, Gorkovenko K, Khanesar MA, Lakoju M, Smith T, Thorp J. Chatty factories: a vision for the future of product design and manufacture with IoT. 2019, http://dx.doi.org/10.1049/cp.2019.0129.

[65] Lakoju Mike, Javed Amir, Rana Omer, Burnap Pete, Atiba Samuelson T, Cherkaoui Soumaya. "Chatty devices" and edge-based activity classification. Discover Internet Things 2021;1(1):5. http://dx.doi.org/10.1007/s43926-021-00004-9.

[66] Balla Prashanth Balraj, Jadhao KT. IoT based facial recognition security system. In: 2018 international conference on smart city and emerging technology (ICSCET). 2018, p. 1–4. http://dx.doi.org/10.1109/ICSCET.2018.8537344.

[67] Peixoto Solon A, Vasconcelos Francisco FX, Guimarães Matheus T, Medeiros Aldísio G, Rego Paulo AL, Lira Neto Aloísio V, de Albuquerque Victor Hugo C, Rebouças Filho Pedro P. A high-efficiency energy and storage approach for IoT applications of facial recognition. Image Vis Comput 2020;96:103899. http://dx.doi.org/10.1016/j.imavis.2020.103899.

[68] Majumder AKM Jahangir, Izaguirre Joshua Aaron. A smart IoT security system for smart-home using motion detection and facial recognition. In: 2020 IEEE 44th annual computers, software, and applications conference (COMPSAC). 2020, p. 1065–71. http://dx.doi.org/10.1109/COMPSAC48688.2020.0-132.

[69] Do Tri-Nhut, Le Cong-Lap, Nguyen Minh-Son. IoT-based security with facial recognition smart lock system. In: 2021 15th international conference on advanced computing and applications (ACOMP). 2021, p. 181–5. http://dx.doi.org/10.1109/ACOMP53746.2021.00032.

[70] Henry Xuef Ant Group AISEC Team. Camera hijack attack on facial recognition system, case study: AML.CS0004 | MITRE ATLAS™. 2020, https://atlas.mitre.org/studies/AML.CS0004/ (accessed 21 August 2022).

[71] Henry Xuef Ant Group AISEC Team. Face identification system evasion via physical countermeasures, case study: AML.CS0012 | MITRE ATLAS™. 2020, https://atlas.mitre.org/studies/AML.CS0012/ (accessed 21 August 2022).

[72] FireceElctronics. IoT heading for mass adoption by 2019. 2017, https://www.fierceelectronics.com/components/iot-heading-for-mass-adoption-by-2019 (accessed 19 August 2022),

[73] Siva Kumar Ram Shankar, Nyström Magnus, Lambert John, Marshall Andrew, Goertzel Mario, Comissoneru Andi, Swann Matt, Xia Sharon. Adversarial machine learning-industry perspectives. In: 2020 IEEE security and privacy workshops (SPW). 2020, p. 69–75. http://dx.doi.org/10.1109/SPW50608.2020.00028.

[74] Gartner. Iot security primer: challenges and emerging practices. 2022, https://www.gartner.com/en/doc/iot-security-primer-challenges-and-emerging-practices (accessed 15 August 2022).

[75] Papernot Nicolas, McDaniel Pat, Wu Xi, Jha Somesh, Swami Ananthram. Distillation as a defense to adversarial perturbations against deep neural networks. 2016, http://dx.doi.org/10.48550/arXiv.1511.04508.

[76] Yumlembam Rahul, Issac Biju, Jacob Seibu Mary, Yang Longzhi. IoT-based android malware detection using graph neural network with adversarial defense. IEEE Internet Things J 2022;1. http://dx.doi.org/10.1109/JIOT.2022.3188583.

[77] Goodfellow Ian, Pouget-Abadie Jean, Mirza Mehdi, Xu Bing, Warde-Farley David, Ozair Sherjil, Courville Aaron, Bengio Yoshua. Generative adversarial nets. In: Advances in neural information processing systems, Vol. 27. Curran Associates, Inc.; 2014, https://proceedings.neurips.cc/paper/2014/hash/5ca3e9b122f61f8f06494c97b1afccf3-Abstract.html (accessed 30 Aug 2022).

[78] Wang Shen, Qiao Zhuobiao. Robust pervasive detection for adversarial samples of artificial intelligence in IoT environments. IEEE Access 2019;7:88693–704. http://dx.doi.org/10.1109/ACCESS.2019.2919695.

[79] Smutz Charles, Stavrou Angelos. Malicious PDF detection using metadata and structural features. In: Proceedings of the 28th annual computer security applications conference. ACSAC '12, New York, NY, USA: Association for Computing Machinery; 2012, p. 239–48. http://dx.doi.org/10.1145/2420950.2420987.

[80] Apruzzese Giovanni, Colajanni Michele, Ferretti Luca, Marchetti Mirco. Addressing adversarial attacks against security systems based on machine learning. In: 2019 11th international conference on cyber conflict (cycon), vol. 900. 2019, p. 1–18. http://dx.doi.org/10.23919/CYCON.2019.8756865.

[81] Shan Shawn, Wenger Emily, Wang Bolun, Li Bo, Zheng Haitao, Zhao Ben Y. Gotta catch'em all: Using honeypots to catch adversarial attacks on neural networks. In: Proceedings of the 2020 ACM SIGSAC conference on computer and communications security. CCS '20, NY, USA: Association for Computing Machinery; 2020, p. 67–83. http://dx.doi.org/10.1145/3372297.3417231.