IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received April 7, 2022, accepted May 11, 2022, date of publication May 16, 2022, date of current version May 27, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3175615

Toward Autonomous Physical Security Defenses Using Machine Learning

BASEL HALAK[1], CHRISTIAN HALL[1], SYED FATHIR[1], NELSON KIT[1], RUWAYDAH RAYMONDE[1], MICHAEL GIMSON[1], AHMAD KIDA[1], AND HUGO VINCENT[2]
[1]School of Electronics and Computer Science, University of Southampton, Southampton SO17 1BJ, U.K.
[2]Arm Ltd., Cambridge CB1 9NJ, U.K.

Corresponding author: Basel Halak (basel.halak@soton.ac.uk)

This work was supported in part by the Royal Academy of Engineering under Grant IF2021\36, and in part by Engineering and Physical Sciences Research Council (EPSRC) under Grant EP/R007268/1.

ABSTRACT The sheer increase in interconnected devices, reaching 50 B in 2025, makes it easier for adversaries to have direct access to the target system and perform physical attacks. This risk is exacerbated by the proliferation of Internet-of-Battlefield Things (IoBT) and increased reliance on the use of embedded devices in critical infrastructure and industrial control systems. Existing anti-tamper designs protect against limited forms of attacks and have deterministic tamper responses, which can undermine the availability of systems. Advancements in physical inspection techniques have enabled stealthier attacks. Therefore, there is a pressing need for more intelligent defenses that ensure a longer operational time while keeping up with the expected increase in the capabilities of adversaries. This study proposes to enhance existing physical protection methods by developing an intelligent anti-tamper using machine learning algorithms. It uses an analytic system capable of detecting and classifying multiple types of behaviors (e.g., normal operation conditions, known attack vectors, and anomalous behavior). The system also has a layered response mechanism and recovery scheme, which reduces false alarms and prolongs the operational time. An experimental platform was constructed and used for data collection and machine learning model training. This study also explored the impact of adversarial learning attacks on the proposed system and subsequently developed a countermeasure. The final prototype was capable of recognizing two types of normal operating conditions (sheltered and exposed environments) and four types of physical attacks. It also has adaptive response and recovery mechanisms.

INDEX TERMS Anti-tamper design, internet of battlefield things (IoBT), machine learning algorithms, adversarial machine learning, physical attacks, hardware security.

I. INTRODUCTION

This Surviving physical attacks in a hostile environment is of utmost importance for nowadays electronics. This need becomes increasingly pressing with the rapid growth of the Internet of Battlefield Things (IoBT) or Battlefield Management Systems. It is expected that the market size worldwide of these systems will exceed 26 billion U.S. dollars by 2027 [1]. One example of such a systems is an unmanned autonomous vehicle (UAV). The latter allows for safer operations in enemy territories without risking human lives, in addition, autonomous coordination can give greater precision, endurance, and reliability than humans. To exploit the UAV potential in military operations, these vehicles must satisfy demanding requirements concerning robustness reliability, and security. In fact, the need for ‘‘physical end-point protection’’ was highlighted as a core security requirement for these systems in a report by the Norwegian Defence Research Establishment (FFI) [2]. The need for physical protection is also crucial in numerous commercial applications such as payment terminals, military Network encryption devices, and Pay-Tv [3].

In general, tamper protection mechanisms consist of a combination of *Tamper Detection* (e.g. using sensors to detect tampering attempts), *Tamper Evidence* (e.g. logging the occurrence of a tamper event), and *Tamper Response* (i.e. actions taken to protect the system upon the detection of an attack) [4]. Tamper detection systems can be either in

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

the form of switches that can recognize whether the enclosure of the device has been breached and whether a special module of the device has been displaced, or sensors capable of noticing changes in the normal operating environment or even circuitry. A plethora of sensors can be used for Tamper detection, including temperature, light, voltage, pressure, and probe sensors, with each sensor type capable of detecting a specific attack [5]. Although defines mechanisms at the chip level are available, these are not included in every chip of a device due to the costly and long process of integrated circuits development [6]. Therefore, multiple chips systems that require a high level of security are placed in a physically secure enclosure [6], [7]. These security boxes encompass the whole system and protect it from physical attacks such as drilling, etching, and probing. One example of this type of defences is the tamper-proof envelope developed in [8], which consists of a multi-layer mesh of conductive traces that surrounds the system. The enclosed device performs continuous measurements of this mesh to detect any open circuits, which subsequently triggers a tamper response that consists of zeroization of critical security parameters (CSPs) such as cryptographic keys. The underlying assumption here is that physical tempering is going to destroy the tracks. However, this approach suffers from two main issues. First, the monitoring circuitry relies on static signals, which means, an attacker could potentially force the expected voltage from an external source, hence avoiding detection. A second drawback is the deterministic nature of the tamper response, which may hinder the operation of the device (e.g. accidental damage to the tracks will trigger an unnecessary response, leading to the removal of CSP and suspension of the device's operations).

The above approach requires a separate battery for the tamper detection system, therefore alternatives designs of security boxes have also been developed for energy-constrained applications, most notably the recent work of Immler V et al in [9], which relies on the use of physically unclonable functions [10]. Similar to [8], the PUF approach in [9] uses an envelope with a fine mesh of electrodes, however, the integrity of the mesh is only checked if a cryptographic key is needed. Subsequently, if the system has not been tampered with, the correct key will be derived from the envelope by measuring the capacitances of the mesh. This will be next used to decrypt and authenticate the firmware of the enclosed host system. On the other hand, a tamper event that leads to a change in the electrical characteristics of the envelope (e.g. removal of some tracks) will prevent the generation of the correct value of the key, which means the CSP remains encrypted.

The PUF approach removes the need for a separate battery to power the tamper detection mesh; however, accidental damage to the envelope has a more severe consequence in this case because it will lead to a deviation of the PUF response from what is expected, preventing the authentication of the firmware, hence rendering the device non-functional.

Overall, existing solutions for secure physical enclosure solutions [11] have several shortcomings. First, their deterministic tamper responses may undermine the availability of the device if detection circuitry triggers a false alarm. Furthermore, the advancement in physical inspection instruments means it can no longer be assumed that the adversary does not know the structure of the mesh; this means breaking into security boxes that mesh-type implementation is going to become easier. Finally, passive monitoring techniques are only capable of detecting attacks with known symptoms (e.g. a tampering attack that causes an open circuit), therefore these approaches are inherently incapable of detecting new forms of attacks (e.g. drilling a hole without destroying the mesh or sustained heating of the device to cause functional errors).

The above discussion shows that to withstand physical attacks in the future, electronics systems require additional layers of protection to keep up with expected advances on the attackers' side. To the best of our knowledge, none of the existing approaches for the detection of physical tampering attacks incorporates the use of machine learning algorithms. The latter are increasingly used in hardware security to construct defines mechanisms that are more effective. Examples of this trend include enhanced detection of hardware Trojan in [12], which uses features extracted from the design netlist or from an on-chip measurement of electrical parameters to construct a machine-learning model that indicates whether the chip has a Trojan. For example, the sensors-based approach [13], relies on the fact that a Hardware Trojan induces a range of abnormalities when activated, which affect various on-chip data sources (performance-counters, data streams, and current measurements).

Another example is the use of machine learning to detect IC counterfeits [14], [15]. For example, the authors of [14] demonstrated a technique that can identify recycled chips through the collection of parametric measurements from on-chip sensors, which are subsequently analysed and classified using a one-class support vector machine(SVM). This approach relies on the fact that recycled chips have different electrical characteristics due to the aging of CMOS devices [16]. Our previous work on anomaly detection has shown the feasibility of using machine-learning techniques to detect different form of physical attacks on embedded devices [17].

This work proposes to enhance existing physical protection methods by developing an intelligent anti-tamper system capable of identifying attack types and responding accordingly. The proposed system can also provide tamper evidence by storing attack-related data. The main contributions are:

1) A security architecture that uses machine-learning algorithms to detect a range of physical attacks, and an comparative analysis of the performance of machine learning algorithms and their efficacy in modeling typical environments of electronic devices and detecting tamper events.
2) A layered tamper response mechanism that ensures the availability of system while mitigating the risks of physical attack.

3) A new adversarial learning attack and a mitigation technique
4) A proof of concept hardware implementation

The remainder of this paper is structured as follows. Section 2 explains the threat model adopted here. Section 3 outlines the security objectives and the system architectures of the proposed solution. Section 4 explains the design rationale of the intelligent detection mechanism, including a comparative analysis of the performance of machine learning algorithms used to build this scheme. Section 5 presents the details of the proposed tamper response mechanism. Section 6 discusses the resilience of the proposed approach to adversarial learning attacks. Section 7 summarises the testing results of the hardware prototype. Section 8 provides a comparative analysis of existing methods. Finally, conclusions are drawn in section 9.

II. THREAT MODELING

It is vital to establish an unambiguous threat model to identify the possible attack scenarios one might anticipate that a system will encounter during its operation. The threats can be classified as deliberate or accidental. An adversary instigates deliberate threats while accidental threats may occur naturally, for example, when there are changes in the operating environment. Deliberate threats are further divided into known and unknown. The former type includes previously encountered attacks mechanisms such as drilling, cold boots, and temperature attacks [18]. Unknown attacks are unpredictable and dependent on the attacker's capabilities and the functionality of the targeted system. The remainder of this section outlines the attack mechanisms considered in this work and explains the assumed capabilities of the expected adversaries.

A. ATTACK MECHANISMS

Three types of attack mechanisms are considered for the development of the proof of concept:

1) TEMPERATURE ATTACKS

This approach consists of running an electronic device outside the range of its operational temperature. In this context, one can differentiate between two types of attacks.

a: HEATING ATTACK

This consists of using extensive heating as a fault injection technique, which has been experimentally demonstrated in [19], wherein the authors showed how to successfully compromise the security of an RSA decryption. Memory blocks are also vulnerable to this type of attack, as the extreme temperature can induce errors in both volatile and non-volatile implementations [20]. This attack can also be performed relatively cheaply, assuming the adversary has sufficient knowledge of the system under attack. It should be mentioned here that other types of fault injection mechanisms also exist such as the use of electromagnetic radiation, laser beam, or power supply glitches [20].

b: FREEZE ATTACK

This consists of inducing a significant reduction of the temperature of random-access memory (RAM) to steal the data. This type of memory retains data for several minutes after being powered down when a freezing attack is performed [21]. The latter is done by cooling the RAM with a cooling agent such as liquid nitrogen [22]. A static RAM (SRAM) will require a temperature as low as −20 °C to perform a freezing attack, in some cases; the adversary can expose the RAM to ionizing radiation to burn in the retained memory permanently after freezing the drive. This allows more time for the attacker to extract sensitive information inside. These attacks became increasingly simple as the physical size of the design shrunk.

2) DRILL ATTACKS

This attack applies to enclosed devices and aims to drill a hole into the protective enclosure, which, in turns, facilitates more advanced physical attacks. A common scenario, in this case, is for the adversary to wage a subsequent freezing attack by pouring liquid nitrogen through the opening.

3) OPEN DEVICE

This attack applies to devices that are placed in protective enclosures. The aim here is to expose the underlying hardware and carry out further invasive attacks.

In practice, these types of security enclosures can be broken into using a mechanical drill as described above or using other methods such as thermal drilling, which can be detected using temperature or vibration sensors. Nevertheless, it is vital to have a mechanism in place (i.e. light sensors) to detect if the enclosure was opened, which acts as the last protection layer if everything else fails.

B. CAPABILITIES AND GOALS OF THE ADVERSARY

There are two types of applications, wherein anti-tamper design is mostly needed.

i. Critical infrastructure means buildings, systems, and other assets whose destruction or disruption would have a major negative impact on national security, public health, and the economy. Internet o things technology is increasingly being deployed for monitoring, control, and data collection in various areas of critical infrastructure such as power plants, water supply systems, and smart grids. Physical security is required in this case to prevent services disruption and protect sensitive information.
ii. Military systems in a contested environment include a wide range of electronic devices used for espionage and undercover activities.

In both of the above cases, it is reasonable to assume that the adversary is either a large criminal organization or a state-funded actor, as they are the most likely beneficiaries of attacking such systems [23].

Therefore, this work assumes the attacker has physical access to the target system and can wage several sophisticated tamper attempts. It is also assumed that they have full knowledge of the internal system architecture, including the encryption algorithms, security protocols, and communication infrastructure.

Based on the above discussions, the adversary can have one of the following three objectives:

1) Undermining the integrity of the system by inducing faulty behavior, which subsequently leads to a malfunction.

2) Stealing the sensitive information stored on the device

3) Sabotaging the entire system, hence undermining its availability.

III. PROPOSED COUNTERMEASURE

A. SECURITY OBJECTIVES

The first step to designing secure systems is to define clearly the security objectives based on the most likely threats and related attack mechanisms. Based on the discussion in section 3, the main objectives of the proposed countermeasure are:

1) CONFIDENTIALITY

This is to ensure the protection of sensitive information and critical security parameters. This is a vital requirement given the fact that the systems are operating in hostile territories (e.g. a drone can have secret data that must not fall in the hands of the enemy such as origin, mission, and likely destinations).

2) INTEGRIT

This is to ensure the system continues to operate correctly and provide protection against tamper attempts.

3) AVAILABILITY

This is to ensure the system remains functional for the expected duration of its operation. The proposed countermeasure aims to ensure availability by mitigating the impact of accidental threats (i.e. those that occur naturally such as changes in the environment). However, the availability of the systems can also be undermined by a deliberate physical attack, such a risk cannot be completely removed given the assumption that the adversary can access the target system, and destroy it or shut it down.

To deliver the above objectives, this work develops an anti-tamper architecture that combines the use of a secure enclosure with additional data analytics ability, which gives the system the capabilities of filtering out false alarms, and distinguishing between different attacks scenarios, hence deploying more effective defenses.

B. SYSTEM ARCHITECTURE

The envisioned solution consists of an intelligent tamper detection SCHEME and a layered response mechanism. To design, such a system one needs to measure the parameters of the environment, analyse the data and trigger an appropriate defines.

The monitoring part of the system requires the use of sensors, whose type and specifications are dependent on the operating conditions and the type of attacks to be detected. This work considers two types of working environments; sheltered and exposed. The former may apply to network encryption devices while the latter applies to espionage and related military devices.

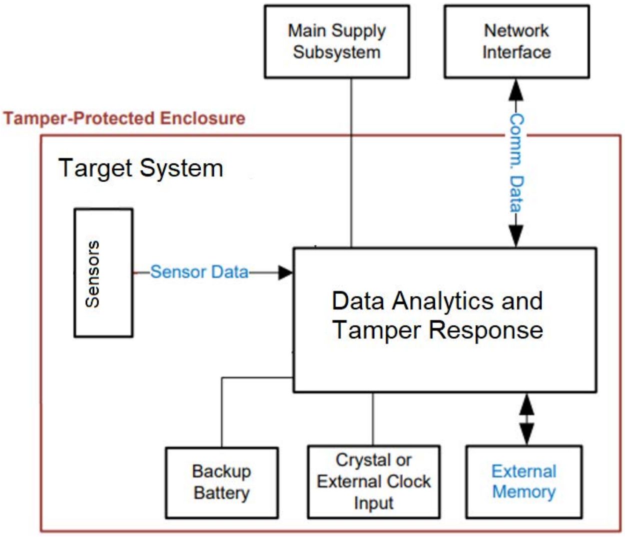

FIGURE 1. An illustration of the architecture of the proposed defense system.

The attacks listed in section 2 require temperature, pressure, movement, vibration, and light sensors, respectively. These are also sufficient for environmental monitoring purposes. The specifications of chosen sensors have been identified based on the characteristic of each attack. For example, the temperature sensors have a range of −40 to 85 °C, which allows them to detect both forms of temperature attacks (heat and freeze).

The second part of the system is responsible for data analytics and attack detection. This is achieved through building machine-learning models of attacks mechanisms based on the output of the sensors' array. Section 4 will explain the experimental setups and the procedures used for data collection and training of the machine learning models.

The third part of the system is the response mechanism, which implements defences suitable for the type of attack detected. Ideally, such a system can also have tamper evidence capabilities such as recording attack data.

An illustrative diagram of the proposed defines is shown in figure 1. It consists of a secure enclosure that surrounds the whole system, sensors that are located in relevant locations on the target system, and the data analytics and response mechanisms. In addition, the proposed countermeasure requires a backup memory to mitigate the risk of power failure. External tamper-proof memory is also required to store attack-related data and critical security parameters if needed.

IV. ATTACK DETECTION AND CLASSIFICATION SYSTEM

A. DESIGN RATIONALE

II. The purpose of the detection part of the system is to analyze incoming data from the sensors to establish whether

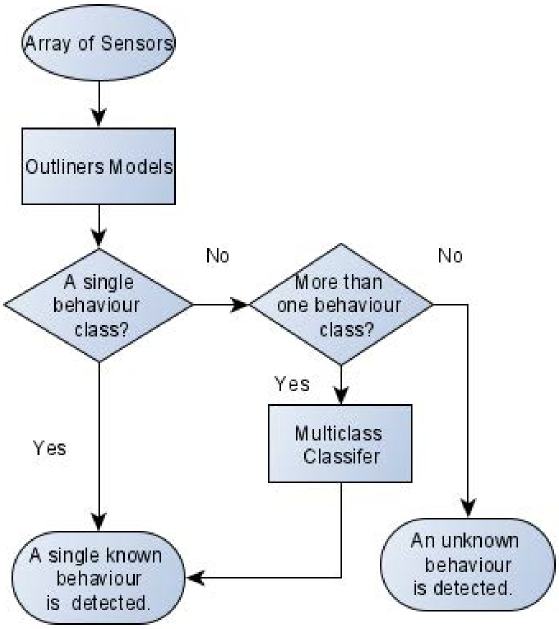
Array of Sensors
Outliners Models
A single behaviour class?
No
More than one behaviour class?
No
Yes
Multiclass Classifer
Yes
A single known behaviour is detected.
An unknown behaviour is detected.

FIGURE 2. Principles of the proposed anomaly detection/classification scheme.

there is an attack. One approach to designing such a system is to use a multinomial machine learning approach [24], capable of classifying multiple behaviors. However, this approach on its own is not reliable because this type of algorithm is very likely to misclassify data points, corresponding to new types of attacks due to the known bias effect. Another design approach is to use outlier algorithms [25], which are excellent at identifying the important features that are associated with specific behaviors. However, the use of the outliers modeling approach, on its own, means that some data points can fall into an unknown zone, where none of the outlier models can account for them. This is because the constructed models are ignorant of the other behavior types leading to decision boundaries overlapping and resulting in contested data points, which may cause misclassification errors.

III. Based on the above considerations, it has become clear that the detection system cannot be constructed using only a multinomial classification or an outlier modeling approach. Therefore, this work adopts an architecture for the detection scheme, wherein the outlier models are used initially to detect known behaviors, subsequently, a multinomial classifier is invoked to resolve contested points. Figure 2 illustrates the working principles of the proposed data analysis and behavior classification scheme.

Training the outlier models can be done using a multiclass algorithm, and giving it two sets of training data, behavior, and non-behavior data points, which generates very accurate models that would create gap-less classification; however, this would suffer from the same known bias issue as the multinomial algorithms. Therefore, a better approach to training these models is to use outlier algorithms, where the only training data given is the behavior that it is being modeled (e.g. the data points are the output of the sensors under a specific attack scenario). The latter approach was adopted in this work.

B. EXPERIMENTAL SETUPS

To select the appropriate machine algorithm for the data analytics part of the system, an experimental platform was constructed. The latter consists of a Raspberry Pi4 device incorporated with an array of sensors that monitor light, temperature, pressure, altitude sensor, humidity, and movement. A motion sensor was also used to control the response mechanism as will be explained later. The system was placed in an opaque protective enclosure, as shown in figure 1.

Next, data were collected for the following cases:

a) Normal operating conditions. Two scenarios have been studied, sheltered (i.e. indoor) and exposed (i.e., outdoor) environments.
b) The device is under attack, wherein four attack scenarios are considered, heating, freezing, open device, and mechanical drill. The attacks have been carried out in a secure lab environment. Three replicas of the experimental platforms had to be created to account for the destructive nature of such attacks.

For each of the above cases, a Python program was created to run on the Raspberry Pi when data was to be collected. The program records the sensors' measurements for each case into a CSV file for later analysis. These files were prefaced with information related to the test (duration and scenario of the test, the refresh rate of the sensors, etc.). The following features were subsequently extracted from the sensors readings: temperature, pressure, humidity, total light, visible light, infrared, full-spectrum light, as well as the X-axis, Y-axis, and Z-axis accelerations. For each experiment, 10000 data points were collected.

C. COMPARATIVE ANALYSIS OF OUTLIER ALGORITHMS

Several outlier algorithms have been evaluated to identify the most accurate anomaly detection algorithm for each type of behaviour. Evaluation metrics have been computed using the scikit-learn python library [12]. The outlier algorithms required a contamination value, which represents the proportion of outliers present in the training dataset, and this has been evaluated to be 0.1.

1) NORMAL BEHAVIOUR IN AN INDOOR ENVIRONMENT

A model of the normal environment was created based on the data collected as described above. Three testing sets have been employed; one containing only tamper data, one containing a mixture of normal data and attack data, and a third set containing only normal data. In each case, the accuracy, recall, precision, and the F1 score have been calculated for each algorithm. Accuracy determines the percentage of correct predictions. When assessing the different machine learning models using the test set containing only tamper data, the classification accuracy obtained for all tested

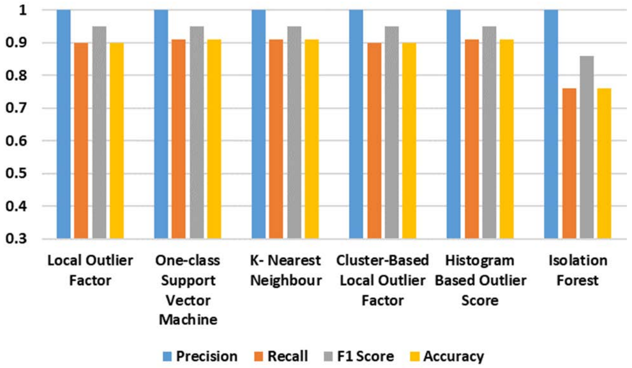

FIGURE 3. Performance comparison of outlier algorithms or normal behaviour indoor using a test set containing normal data only.

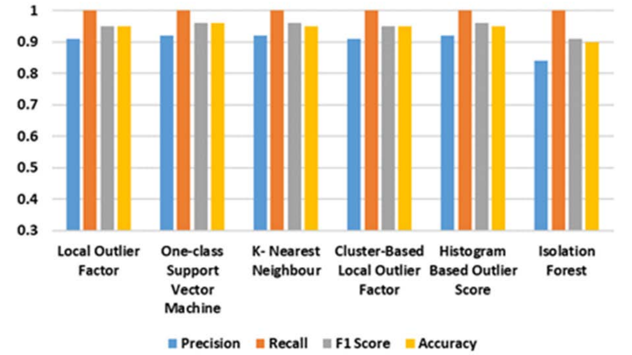

FIGURE 4. Performance comparison of outlier algorithms for normal behaviour indoors using a test set containing a mixture of normal and tamper data (tamper analysis).

algorithms was 100%. The accuracy of the machine learning algorithms was comparable in the remaining data sets, except for the isolation forest algorithm, which seems to have lower accuracy, as shown in figures 3 and 4 respectively. Accuracy is not the only metric that should be considered. Precision (i.e. number of true positives divided by the sum of true positives and false positives) and recall (i.e. the number of true positives has been determined divided by the total number of points that should have been classified as true positives) are also important. F1 score combines both precision and recall. A precision value of ''1'' suggests the classification algorithm does not generate false positives (i.e. no attack points are classified as normal behaviour). A recall value of ''1'' suggest the algorithm has no false negatives (i.e. no normal points are classified as an attack). The results obtained from the above analysis indicate the one-class support vector machine has slightly better overall accuracy (96%), so it was chosen to model the normal behaviour of the system. The experiment above was also carried out in an exposed environment (i.e. the device was placed outdoor). It was also found that the one-class support vector machine provided the best overall results.

2) ATTACK SCENARIOS

The outlier detection models for each considered attack have been constructed similarly. Two test sets have been employed, one containing only data representing the tamper event under test (inlier test set), one containing normal behaviour data. A 100% classification score was again obtained for all algorithms when assessed with the test set containing normal environment data. The results of the inlier test are depicted in figures 5, 6, 7, and 8 respectively. It can be observed that the one-class support vector machine models produce marginally better accuracy in the case of the freezing, open device, and drill attacks, respectively, therefore it was chosen for these cases. For the heating attack, the isolation forest model was chosen as it has the best overall results.

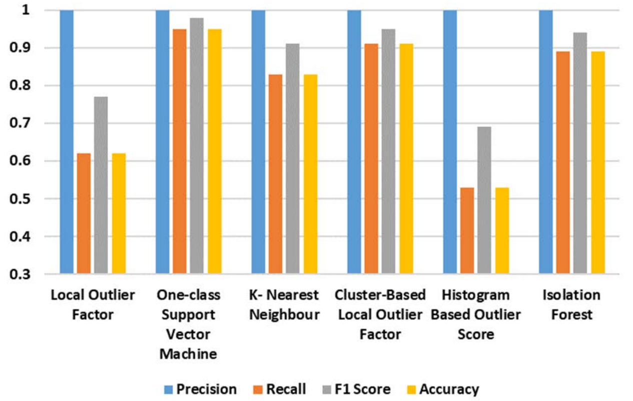

FIGURE 5. Performance comparison of outlier algorithms for freezing attack using a test set containing a mixture of normal and tamper data.

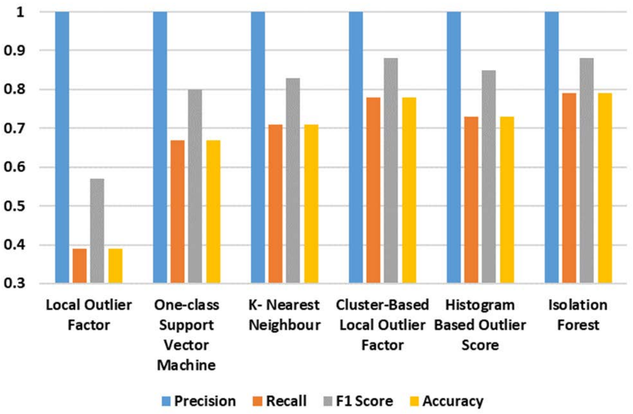

FIGURE 6. Performance comparison of outlier algorithms for the heating attack using a test set containing a mixture of normal and tamper data.

3) COMPARATIVE ANALYSIS OF MULTICLASS ALGORITHMS

The multinomial algorithm is employed to settle disputes between the outlier models, as shown in figure 2. Therefore, it is not expected to be trigged frequently so algorithms that are more complex can be employed in this case. The machine learning models have been constructed based on the data collected previously, which include typical behaviours and attack scenarios. The testing used a python script that was developed based on the scikit-learn package, which included all the algorithms being considered. Three evaluation metrics are used, accuracy, F1 score, and the lowest accuracy. The latter is the lowest precision value associated with a single

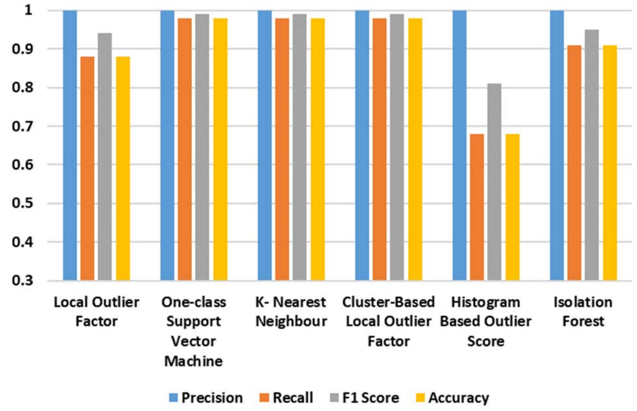

FIGURE 7. Performance comparison of outlier algorithms for an open device attack using a test set containing a mixture of normal and tamper data.

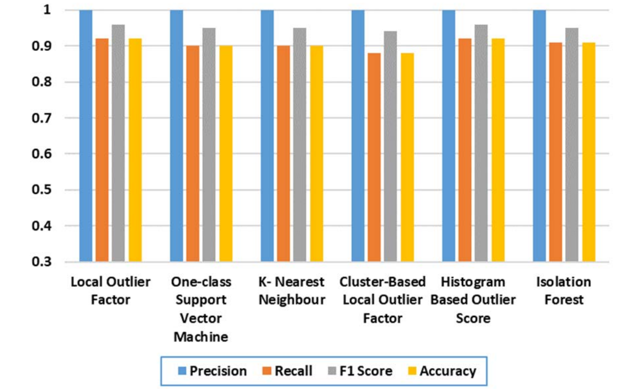

FIGURE 8. Performance comparison of outlier algorithms for an open device attack using a test set containing a mixture of normal and tamper data.

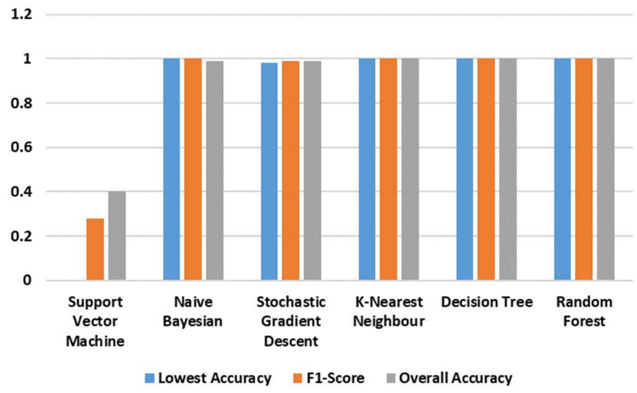

FIGURE 9. Performance comparison of multiclass algorithms.

class, this metric allows for insight into whether the algorithm is reliable at its worst. The results, shown in figure 9 show that K-nearest, the Decision tree, and the random forest have the best performance in this case.

V. TAMPER RESPONSE DESIGN AND IMPLEMENTATION

The purpose of the tamper response scheme is to deploy defences mechanisms appropriate for the detected attacks and ensure the continued operation of the device in case of false alarms.

A. DESIGN RATIONALE OF THE LAYERED TAMPER RESPONSE

This work has adopted a layered approach for designing the response mechanism, wherein the system can recognize distinct risk levels and respond accordingly. The advantage of such an approach is twofold:

- It allows the system to modulate its response depending on the risk level, preventing undue severe measures owing to false alarms.
- This makes it feasible to keep the system in sleep mode when the risk is perceived to be low, thereby preserving energy.

Therefore, the system was constructed to include the following risk levels:

Level 0: this is the lowest level of risk, wherein the system remains largely in a sleep mode and only the motion sensor is active. This stops the detection algorithm from constantly running and wasting energy.

Level 1: this level is activated as soon as the motion sensor is triggered, indicating that a potential adversary is approaching. Upon reaching this level, the system wakes up, reads the output of the sensors, and runs the detection algorithm. It will subsequently, move back to level 0 if there is no attack or level 2 if an attack is detected. In the latter case, appropriate countermeasures are deployed.

Level 2: this is the highest level of risk, wherein the attack detection is run again to confirm the attack is still being carried out before triggering the last layer of defences.

Recovery Mode: the system moves to this state if no more attacks are detected after the deployment of level 1 defences. Recovery measures are implemented to ensure the system can go back to a normal operational mode.

B. DESIGN RATIONALE OF THE PROPOSED DEFENCES

The description and rationale for each of the defences of the tamper response mechanism are explained below.

1) INFORMING THE SYSTEM OWNER

This work has used a GSM module to transmit SMS messages to the owner when an attack is detected or a false alarm is raised. This defines provides tamper evidence and allows the owner to take extra measures if needed.

2) SUSPENDING ALL SENSITIVE OPERATIONS AND MOVING THE CSP TO THE TAMPER-PROOF MEMORY

This is deployed in the first phase of the level 1 response to a heat/unknown attack to prevent the adversary from waging a fault injection attack that reveals the encryption key.

3) OVERWRITING ANY CSP STORED IN RAMs

This is deployed in the first phase of the level 1 response to freezing/drill or open device attacks. It is designed to prevent a scenario wherein the adversary opens the device's enclosure and pours liquid nitrogen onto the SRAM block. In this case, overwriting all the sensitive data stored in the SRAM memory will render such an attack less effective [26].

4) DELETION OF CSP FROM THE EXTERNAL MEMORY

This is deployed in the first phase of the level 2 response to heart/unknown attacks. This zeroization process [27] is designed to prevent the adversary, who is assumed to have broken the device enclosure, from obtaining the encryption key through further invasive attacks.

5) STORAGE OF SENSORS DATA IN A TAMPER-PROOF MEMORY

This is deployed in the first phase of the level 1 response to unknown attacks. It is designed to allow future analysis of the data and update the detection model with new attack vectors.

6) POWERING DOWN THE DEVICE

This is deployed in the second phase of the level 2 response to all attacks. It is designed to prevent the adversary, who is assumed to have broken the device enclosure and have access to the system, from carrying out any further analysis.

7) RECOVERY ACTIONS

These measures are designed to ensure the device can recover from a failed attack or a false alarm. The steps taken depend on the attack that was detected initially. For heart/unknown attacks, the system will retrieve the CSP from the tamper-proof memory and resume its operations. For other types of attack, a new CSP will need to be obtained through a secure key provision scheme.

Figure 10 includes a detailed diagram of the tamper response mechanism, which shows where each of the above-mentioned defences is invoked. The response mechanism was implemented using Python and was subsequently integrated with the detection scheme presented in section 3.

VI. RESILIENCE TO ADVERSARIAL LEARNING ATTACKS

Adversarial machine learning refers to a set of attacks techniques against intelligent systems, wherein the adversary aims to perturb, poison, or seal the underlying machine learning models [28], [29]. These types of attacks are of particular concern in safety-critical applications [30].

This section explores the use of this mechanism to undermine the operations of the proposed tamper-proof detection mechanism and develops a countermeasure to enhance the security of our solution.

A. THREAT MODEL

The threat model and assumptions in section 2 are still valid in this case. However, these may not be sufficient for an attacker to devise an adversarial machine learning attack. Therefore, this section assumes the adversary has the following additional capabilities:

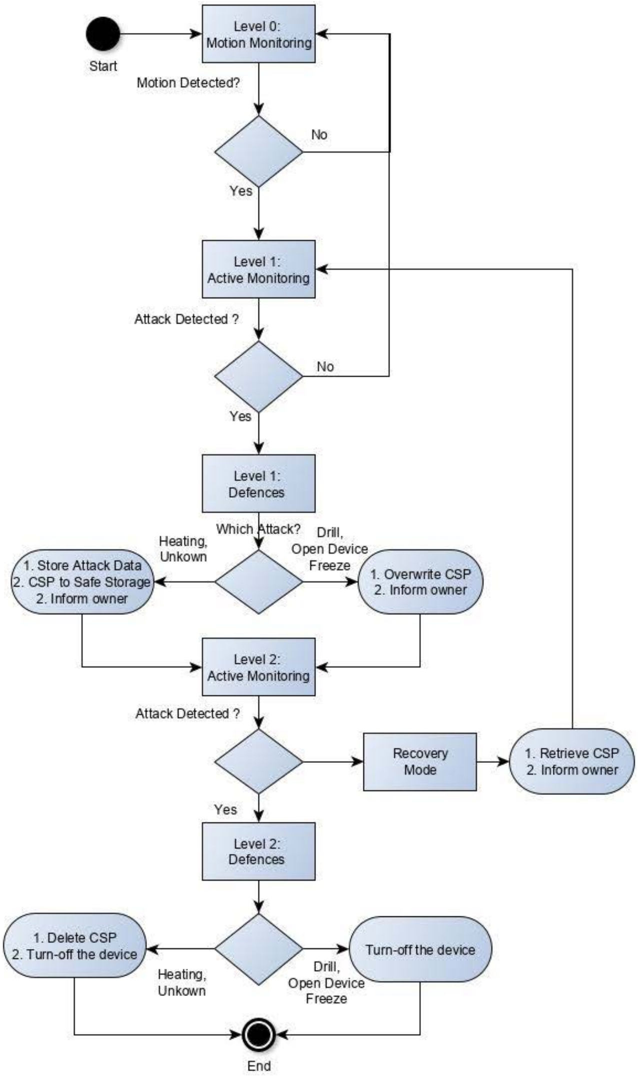

FIGURE 10. **Tamper response flow.**

1) White-box knowledge of the tamper detection architecture, which is the functionality of the tamper detection system as well as all the machine-learning models and their respective parameters.

2) While-box knowledge of the tamper response mechanisms.

The primary goal of the adversary is to deceive the detection system, hence bypassing the defences.

To achieve their objectives, the attacker has two possible routes, either poison the training data or spoof the system. The first approach is highly unlikely to succeed for the type of applications being considered (e.g. defines, critical infrastructure), this is because these systems are normally developed and trained in secure locations. Therefore, this work explores alternative approaches to wage adversarial learning attacks, as will be explained next.

B. ADVERSARIAL ATTACK DESIGN

A closer inspection of the layered response mechanism in figure 10 indicates that the best way to bypass the defences is

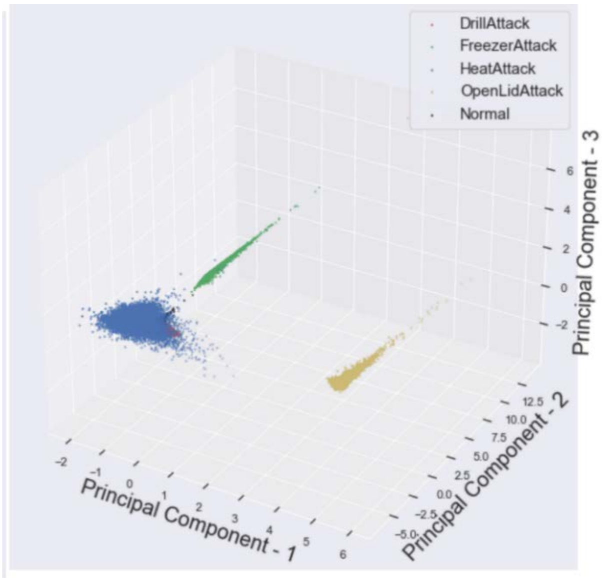

FIGURE 11. Principal component analysis of multiclass classifier training data.

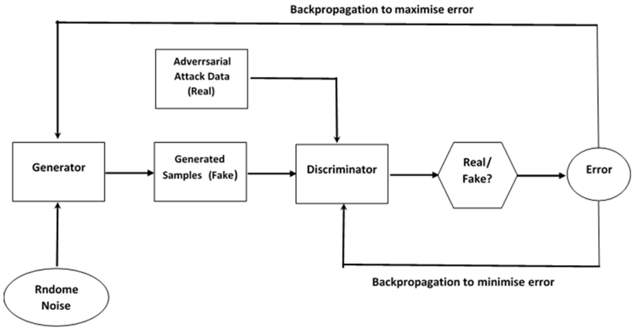
Backpropagation to maximise error
Adverrsarial Attack Data (Real)
Generator
Generated Samples (Fake)
Discriminator
Real/ Fake?
Error
Rndome Noise
Backpropagation to minimise error

FIGURE 12. Adversarial attack data generation process.

TABLE 1. The impact of adversarial data on detection accuracy.

Test Data	Attack Detected	
	Heating Attack	Drill Attack
Adversarial Data Points	8%	92%
Normal Training Data	0%	100%

to trick the detection system into classifying a serious attack as normal behaviour or even a mild attack.

There are two ways the adversary can achieve this goal. The first approach is to cause the outliers models to misclassify an ongoing attack. This was deemed highly unlikely as these models amount to a series of decision boundaries, which can only be manipulated by poisoning the training data, which is not possible. A second method is to target the second phase of the detection system i.e. the multi-nominal classifier.

To explore the feasibility of this approach, principal component analysis (PCA) was conducted to visualize how the different attack behaviours were distributed in space, as shown in figure 11. It was found that all but one defined attack behaviour, the open lid attack, are closely packed in one cluster. This means it may be possible to cause the multi-nominal models to misclassify a serious attack, hence triggering an incorrect or ineffective response. It was subsequently observed, from the response mechanisms in figure 10, that the only effective approach is to get the system to misclassify a drill attack as a heating attack. This is because the level 1 response to a heating attack does not delete the critical security parameters from the RAM, which may give the attacker the time required to drill a hole into the box and freeze the memory before the system has a chance to overwrite its sensitive contents.

To achieve this goal, there is a need to create "combined data points" wherein the temperature sensor represents a heating attack while the remaining sensors' outputs represent a drill attack. This was achieved by waging a simultaneous heating and drilling attack.

The combined data points have been created using a generative adversarial network, which was constructed and trained based on the approach in [31] as illustrated in figure 12.

The real adversarial data attack was based on the combined attack points described previously (a heating and drilling simultaneous attack). A linear neural network architecture was used for the generator. A similar structure was used for the discriminator, wherein an additional linear layer and an increased number of neurons in the middle layers were added. In both models, Pytorch's "Dropout" layer was used to reduce the risk of overfitting our models. The hyperactive parameters used in this case are as follows. A learning rate of 0.00001, a batch size of 50, and 1000 epochs for training. The training process is as follows. The generator generates new data points based on information from previous epochs and random noise. These "fake" points are subsequently passed to the discriminator along with the real data. The prediction error is fed back to each model. The process is repeated until the specified number of epochs has been reached.

Next, the trained generator model was used to create 50,000 adversarial data points. The latter was then applied to the detection algorithm from section 2. The results are summarised in Table 1.

The results indicate that 8% of the adversarial data points are misclassified, all of which have invoked the multiclass classifier according to the description of the detection system in figure 2. This reduction in the detection accuracy shows that it may be possible in a small number of cases to deceive the detection system. This may not be acceptable in certain applications; therefore, next section develops a countermeasure for this type of attack.

C. COUNTERMEASURES FOR ENHANCED PROTECTION FROM ADVERSARIAL ATTACKS

The previous section has demonstrated that the adversarial attack was successful in 8% of the cases. To remove such risk, the design of the detection system should be enhanced to reduce such probability to 0%. This was achieved by retraining the with adversarial examples included in the dataset as suggested in [32]. To implement this approach, the adversarial data points were added to the training dataset of the Random Forest algorithm used for the multi-nominal classifier. The latter was re-trained and tested using a test set that included a mixture of adversarial data points and real attack data. The newly trained model was able to classify all adversarial data points correctly.

VII. PROTOTYPE DEVELOPMENT

A prototype of the anti-tamper design was built to test the fully integrated system. This was constructed using the same experimental platform described in section 3 and consists of the external enclosure, detection system, and the tamper response mechanism as depicted in figure 1. The system was tested for the following behaviours. Normal operating conditions: the device was placed indoors and the output of the detection algorithm and response mechanism was monitored. The system was initially placed in a sleep mode. Next, the system was awakened by triggering the motion centre and was able to detect successfully normal operating conditions. For the heating attack, the ambient temperature was slowly increased, until it hit around 60 °C. At the start and end of the test, the behaviour classification program gave the correct output. At the start of the test, the system detected a normal operating condition. At the end of the test, a heating attack was detected, which triggered the response mechanism to initially move the encryption key to the external memory and subsequently delete it (level 2 response). Next, a mechanical drill was used to create a hole in the metal enclosure. This was also detected correctly and led to overwriting the non-volatile memory initially. The attack continued to test level 2 defines, which resulted in a power shut down of the device. The open device attack was emulated by lifting the cover of the enclosure, which triggered the light sensors and activated the corresponding response mechanism. To test the design's capability of detecting unknown behaviour, the enclosed device was repeatedly shaken. This behaviour was not learned previously. This was classified as an unknown attack; the corresponding sensor data have also been stored in the external memory. Finally, a freeze attack was emulated with a deep freeze aerosol spray that cools down to −51 °C, which triggered the overwriting process. Overall, the test results have shown that the prototype satisfies the design requirements.

VIII. COMPARISON WITH EXISTING SOLUTIONS

Existing anti-tamper designs can be classified into two categories, Mesh-based and PUF based. Mech-based techniques consist of a multi-layer mesh of conductive traces embedded into the secure enclosure. The integrity of this mech is constantly monitored to detect physical tampering [6], [8]. A PUF-based approach relies on the intrinsically unique characteristics of a fine grid of connections that are embedded into a secure enclosure [9]. This is used to generate a distinctive digital identifier for each device, which is subsequently used to derive a decryption key required to decipher sensitive data stored on the device. Any damage to this grid by a physical attack will make it impossible to generate the key; therefore, the adversary will only have access to encrypted data. Mesh-based approaches require an external battery to run the continuous monitoring. On the other hand, PUF-based techniques only check the integrity of the grid when the system is required to walk up, which makes it more energy-efficient, however, a PUF-based approach does not allow information to be retrieved if the grid is damaged, as this will prevent the generation of the correct key. Table 2 compares the above-mentioned techniques with the proposed method. The comparison is performed using three categories, the characteristics of the anti-tamper design, the security properties, and the energy requirements.

TABLE 2. Comparative analysis with existing anti-tamper design of security boxes.

Design Metric		Mesh-based Secure Enclosures [8]	PUF-based Secure Enclosure [9]	This work
Anti-Tamper Design Qualities	Tamper Response	Deterministic, CSP recoverable	Deterministic, CSP Irrecoverable	Attack-Dependent, CSP recoverable
	Tamper Evidence	Not Provided	Not Provided	Provided
	Tamper Resistance	Physical damage to the enclosure	Physical damage to the enclosure	Drilling Device Open Heat Freeze Attack Unknown
Security Properties	Availability	Not provided	Not Provided	Provided
	Integrity	Partly	Partly	Provided
	Confidentiality	Provided	Provided	Provided
Overheads	Energy Requirement	Separate Battery Required	None	Separate Battery Required

In terms of anti-tamper characteristics, the proposed approach has a tailored response mechanism that is dependent on the attack type compared to the deterministic responses of existing types, which makes it more effective. In addition, the proposed solution defends against multiple forms of attacks. Another point to highlight here is that the CSP is recoverable in both the Mesh-based and the ML-based techniques, which is not the case for the PUF method as explained above. In addition, the proposed method provides tamper evidence through the storage of attack data as explained in section 5. In terms of security properties. The use of machine learning makes it feasible for the system to recognize the normal operating environment and distinguish between different types of attacks. As a result, the proposed solution can reduce the number of false alarms and respond appropriately to different types of attacks, which enhances the availability of the system. Existing tamper mechanisms can guarantee system integrity for a specific attack scenario (i.e. physical damage), while the proposed solution can recognize and defend against more types of attacks, in addition to its capability to recognize new forms of behaviour.

Finally, in terms of energy requirement, the PUF-based solution is the most energy-efficient approach.

IX. CONCLUSION AND FUTURE WORK

The proliferation of the Internet of Battlefield Things (IoBT) combined with the increased reliance on the use of embedded devices in the critical infrastructure and industrial control systems have significantly increased the need for anti-tamper design techniques, which protect electronics systems deployed in a hostile or physically exposed environment. Existing methods that rely on monitoring the electrical characteristics of a mesh-based grid fitted onto a security enclosure may not be sufficient, due to the limited forms of threats they can mitigate, and the deterministic nature of their responses, which undermine the system's availability in case of false alarms. Other approaches that use physically unclonable functions provide a more energy-efficient solution but suffer from the same shortcomings, moreover, it may lead to irrecoverable loss of data in case of accidental damage to the enclosure. This work proposed an autonomous anti-tamper design using machine learning algorithms. The essence of this approach consists of using an analytic system capable of detecting and classifying multiple types of behaviours (e.g. normal operation conditions, known attack vectors, and anomalous behaviour). The system also has a layered response mechanism and a recovery scheme, which reduces false alarms and prolongs operational time. An experimental platform has been built and used to train the machine learning models and explore the efficacy of available algorithms. Testing results of the final prototype have demonstrated the capability of the system to indenting a range of known attack mechanisms (e.g. temperature attack, mechanical drilling, Device open...), in addition to its ability to record new forms of attacks. This work has also investigated the impact of adversarial learning attacks against the proposed system and devised a mitigation technique. Future work will extend the systems to include other forms of physical attacks such as electromagnetic radiation and power supply glitches. It will also explore the trade-off between performance and security requirements, to reduce overheads.

REFERENCES

[1] Research and Markets. (2020). *Battlefield Management Systems Market to 2027—Global Analysis and Forecast by Component; System; Application*. [Online]. Available: https://www.researchandmarkets.com/reports/4987798/battlefield-management-systems-market-to-2027#rela0-4897475

[2] S. B. F. Mancini, R. Fardal, J. H. Wiik, B. Greve, L. E. Olsen, and B. Bjerketveit, "Information security for unmanned and autonomous vehicles—Main challenges and relevant operational concepts," Norwegian Defence Res. Establishment, Kjeller, Norway, FFI Rep. 19/00888, 2019. [Online]. Available: https://publications.ffi.no/nb/component/jcar/asset/dspace:7002/1772319.pdf

[3] A. Paulshus, "Anti-tamper and cryptography in pay-TV—Lessons learned," presented at the AVT-337 Res. Workshop Anti-Tamper Protective Syst. NATO Oper., 2021.

[4] J. Grand, "Practical secure hardware design for embedded systems," in *Proc. Embedded Syst. Conf.*, San Francisco, CA, USA, 2004, pp. 1–25.

[5] S. H. Weingart, "Physical security devices for computer subsystems: A survey of attacks and defenses," in *Cryptographic Hardware and Embedded Systems—CHES 2000*. Berlin, Germany: 2000, pp. 302–317.

[6] J. Obermaier and V. Immler, "The past, present, and future of physical security enclosures: From battery-backed monitoring to PUF-based inherent security and beyond," *J. Hardw. Syst. Secur.*, vol. 2, no. 4, pp. 289–296, Dec. 2018.

[7] *CryptoServer Se-Series Gen2 Security Policy (Compliant to FIPS 140-2 Level and 3)*, UTIMACO, Aachen, Germany, 2018.

[8] P. Isaacs, T. Morris, Jr., M. J. Fisher, and K. Cuthbert, "Tamper proof, tamper evident encryption technology," presented at the Pan Pacific Symp. (SMTA), 2013.

[9] V. Immler, J. Obermaier, M. König, M. Hiller, and G. Sig, "B-TREPID: Batteryless tamper-resistant envelope with a PUF and integrity detection," in *Proc. IEEE Int. Symp. Hardw. Oriented Secur. Trust (HOST)*, Apr. 2018, pp. 49–56.

[10] B. Halak, *Physically Unclonable Functions: From Basic Design Principles to Advanced Hardware Security Applications*. Cham, Switzerland: Springer, 2018.

[11] C. Bao, D. Forte, and A. Srivastava, "On reverse engineering-based hardware Trojan detection," *IEEE Trans. Comput.-Aided Design Integr. Circuits Syst.*, vol. 35, no. 1, pp. 49–57, Jan. 2016.

[12] C. Dong, Y. Liu, J. Chen, X. Liu, W. Guo, and Y. Chen, "An unsupervised detection approach for hardware trojans," *IEEE Access*, vol. 8, pp. 158169–158183, 2020.

[13] Z. Huang, Q. Wang, Y. Chen, and X. Jiang, "A survey on machine learning against hardware trojan attacks: Recent advances and challenges," *IEEE Access*, vol. 8, pp. 10796–10826, 2020.

[14] Y. Jin, D. Maliuk, and Y. Makris, "A post-deployment IC trust evaluation architecture," in *Proc. IEEE 19th Int. On-Line Test. Symp. (IOLTS)*, Jul. 2013, pp. 224–225.

[15] K. Huang, J. M. Carulli, and Y. Makris, "Parametric counterfeit IC detection via support vector machines," in *Proc. IEEE Int. Symp. Defect Fault Tolerance VLSI Nanotechnol. Syst. (DFT)*, Oct. 2012, pp. 7–12.

[16] A. Stern, U. Botero, F. Rahman, D. Forte, and M. Tehranipoor, "EMFORCED: EM-based fingerprinting framework for remarked and cloned counterfeit IC detection using machine learning classification," *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.*, vol. 28, no. 2, pp. 363–375, Feb. 2020.

[17] B. Halak, *Ageing of Integrated Circuits: Causes, Effects and Mitigation Techniques*. Cham, Switzerland: Springer, 2020.

[18] C. H. B. Halak, S. Fathir, N. Kit, R. Raymonde, and H. Vincent, "On the feasibility of using machine learning for an enhanced physical security of embedded devices," presented at the IEEE SMARTTECH, May 2022, pp. 206–2011.

[19] R. J. Anderson, "Physical tamper resistance," in *Security Engineering: A Guide to Building Dependable Distributed Systems*, 2nd ed. Hoboken, NJ, USA: Wiley, 2008, pp. 483–521.

[20] E. Johansson, "Tamper protection for cryptographic hardware: A survey and analysis of state-of-the-art tamper protection for communication devices handling cryptographic keys," Dept. Electr. Eng., Linkoping Univ., Linköping, Sweden, Tech. Rep. ISRN: LIU-ISY/LITH-EX-A–20/5306–SE, 2020.

[21] C. H. Kim and J.-J. Quisquater, "Faults, injection methods, and fault attacks," *IEEE Des. Test. Comput.*, vol. 24, no. 6, pp. 544–545, Nov. 2007.

[22] M. Hutter and J.-M. Schmidt, "The temperature side channel and heating fault attacks," in *Smart Card Research and Advanced Applications*. Cham, Switzerland: Springer, 2014, pp. 219–235.

[23] B. Halak, "CIST: A threat modelling approach for hardware supply chain security," in *Hardware Supply Chain Security*. Cham, Switzerland: Springer, 2021, pp. 3–65.

[24] M. Aly, "Survey on multiclass classification methods," *Neural Netw.*, vol. 19, pp. 1–9, Nov. 2005.

[25] H. C. Mandhare and S. R. Idate, "A comparative study of cluster based outlier detection, distance based outlier detection and density based outlier detection techniques," in *Proc. Int. Conf. Intell. Comput. Control Syst. (ICICCS)*, Jun. 2017, pp. 931–935.

[26] S. B. Joshi, "Standards and techniques to remove data remanence in cloud storage," in *Proc. IEEE Punecon*, Nov. 2018, pp. 1–4.

[27] S. W. Smith, "Zeroization," in *Encyclopedia of Cryptography and Security*, H. C. A. van Tilborg and S. Jajodia, Eds. Boston, MA, USA: Springer, 2011, p. 1401.

[28] E. Anthi, L. Williams, M. Rhode, P. Burnap, and A. Wedgbury, "Adversarial attacks on machine learning cybersecurity defences in industrial control systems," *J. Inf. Secur. Appl.*, vol. 58, May 2021, Art. no. 102717.

[29] C. Pauling, M. Gimson, M. Qaid, A. Kida, and B. Halak, "A tutorial on adversarial learning attacks and countermeasures," 2022, *arXiv:2202.10377*.

[30] A. Kloukiniotis, A. Papandreou, A. Lalos, P. Kapsalas, D.-V. Nguyen, and K. Moustakas, "Countering adversarial attacks on autonomous vehicles using denoising techniques: A review," *IEEE Open J. Intell. Transp. Syst.*, vol. 3, pp. 61–80, 2022.

[31] J. Kalin, *Generative Adversarial Networks Cookbook*. Birmingham, U.K.: Packt Publishing, 2018.

[32] I. Ilahi, M. Usama, J. Qadir, M. U. Janjua, A. Al-Fuqaha, D. T. Hoang, and D. Niyato, "Challenges and countermeasures for adversarial attacks on deep reinforcement learning," *IEEE Trans. Artif. Intell.*, vol. 3, no. 2, pp. 90–109, Apr. 2022.

BASEL HALAK is an Associate Professor of secure electronics and the Director of the Cyber Security Academy with the University of Southampton. He is a Visiting Scholar with the Technical University of Kaiserslautern, an Industrial Fellow of the Royal Academy of Engineering, a Senior Fellow of the Higher Education Academy, and a National Teaching Fellow of Advance HE U.K. He has published more than 100 refereed conference and journal papers and authored five books on security and reliability of electronic devices and systems. His research interests include hardware security, digital design, and embedded systems.

He is with the several technical program committees, such as HOST, IEEE DATE, DAC, IVSW, ICCCA, ICCCS, MTV, and EWME. He is a member of the Hardware Security Working Group of the World Wide Web Consortium (W3C). He is an Associate Editor of IEEE ACCESS and the Editor of the *IET Circuits, Devices and Systems Journal*.

CHRISTIAN HALL received the B.Sc. degree in electronics engineering from the University of Southampton, U.K., in 2019, where he is currently pursuing the postgraduate degree in secure computer architecture. His research interests include security of cyber physical systems and artificial intelligence.

SYED FATHIR received the B.Sc. degree in electronics engineering from the University of Southampton, U.K., in 2018, where he is currently pursuing the postgraduate degree in computer engineering. His research interests include hardware security and physical attacks.

NELSON KIT received the B.Sc. degree in electronics engineering from the University of Southampton, U.K., in 2019, where he is currently pursuing the postgraduate degree in computer engineering. His research interests include hardware security and physical attacks.

RUWAYDAH RAYMONDE received the B.Sc. degree in computer science from the University of Southampton, U.K., in 2019, where she is currently pursuing the postgraduate degree. Her research interests include cyber security and machine learning.

MICHAEL GIMSON received the B.Sc. degree in computer science from the University of Southampton, U.K., in 2020, where she is currently pursuing the postgraduate degree. Her research interest includes adversarial learning attacks.

AHMAD KIDA received the B.Sc. degree in computer science from the University of Southampton, U.K., in 2020, where she is currently pursuing the postgraduate degree. Her research interests include intelligent systems and machine learning.

HUGO VINCENT is the Principal Research Engineer and the Head of the Security Group, Arm Research, Cambridge, U.K. He is working with DARPA, University of Cambridge, and Arm's commercial and academic partners. He has contributed to the design of security and cryptographic features in Arm processors and operating systems, including Mbed OS (Arm's operating system for the IoT), Arm's IoT strategy and technology, hypervisors, and trusted system architecture. His research interests include security in distributed systems and the IoT.