Side-Channel-Driven Intrusion Detection System for Mission Critical Unmanned Aerial Vehicles

Alejandro Almeida*, Muneeba Asif*, Md Tauhidur Rahman†, and Mohammad Ashiqur Rahman*
*Analytics for Cyber Defense (ACyD) Lab, †Security, Reliability, Low-power, and Privacy (SeRLoP) Lab
Department of Electrical and Computer Engineering, Florida International University, USA
Emails: {aalme004, masif004, mdtrahma, marahman}@fiu.edu

***Abstract*—Hardware Trojans (HTs) are gradually becoming a growing threat in the IoT landscape. This type of attack can result in catastrophic incidents for unmanned aerial vehicles (UAVs). Examples of these incidents could be information leakage, drone malfunction, which leads to crashes, and data integrity issues in information gathered by the sensors. Other papers have tried to resolve this issue by focusing on enhancing encryption and hardening the physical properties of the device to restrict information leakage. However, this research aims to demonstrate the efficacy of a side channel-based intrusion detection technique. This technique uses machine learning to detect HTs. We test this by constructing a PWM-inverting HT and implementing it into a UAV with a Pixhawk flight controller. By doing so, we demonstrate how this Intrusion Detection System (IDS) technique effectively detects incidents related to HT implementation on UAVs, analyzing discrepancies in the system's impedance. Our proposed IDS yields ROC and accuracy scores up to 99.5% and 98%, respectively, in detecting HTs.**

***Index Terms*—Unmanned Aerial Vehicles; hardware Trojans; side-channel analysis; impedance analysis; intrusion detection.**

I. Introduction

Due to the decreasing prices, increasing ease of use, and growing commercially off-the-shelf (COTS) availability, small-scale unmanned aerial vehicles (UAVs) have gained much attention in the public eye. According to the US Federal Aviation Administration (FAA), small-scale UAVs will reach $3.17M by 2022 [1]. UAVs have a wide range of applicability in both military and civilian sectors. It is widely known that in the combat against terrorism, speed and efficiency are of the essence, and these factors could be significantly augmented by modern UAV [2]. They are a valuable asset in disaster response by providing real-time aerial imaging that can help in search and rescue missions. In Portugal, most SAR operations count on aircraft and ship support to find and retrieve rescue targets, but this requires a lot of resources and manpower. The evolving technology in UAVs is an opportunity to improve the response capabilities of the Portuguese Navy and Air Force, with the deployment of fewer resources and manpower [3]. Nonetheless, the positive contributions that UAVs can provide to society are numerous. However, there is one major caveat regarding UAVs, and that is their susceptibility to cyber-attacks. Hardware Trojan's (HT) effects are one of the biggest concerns in the modern age [4], [5], [6].

An HT is a malicious modification made to an authentic design to cause malfunction, steal or leak sensitive data, cause denial-of-service (DoS) attacks, or impede the normal functioning of the devices to disrupt the device completely. The vulnerable computing devices are not limited to CPU but also to ASICs, FPGAs, and others [7]. With the intensifying costs of circuit fabrication plants and the overheads involved, most IC designers prefer a setup devoid of fabrication. Consequently, the devices are manufactured off-shore, where security cannot be guaranteed or verified. With such globalization of the manufacturing process, the community has seen an escalation in implanted HTs in the devices [7] As part of this research, we investigated different side channels that could expose the presence of an HT on a UAV.

A vector network analyzer (VNA) analyzes how the device under test (DUT) responds at various frequencies. A VNA is a unique test device that consists of a source to generate a signal and a receiver to detect alterations to the stimulus signal created by the DUT. VNA injects a known stimulus signal into the DUT and measures the signal reflected from the input side and transmitted to the output side. However, active digital circuits exhibit a time-varying voltage and current tend to behave differently under different logic instructions. Using the VNA, we determined that specific patterns in impedance can be a characteristic signature of an HT being present.

The device impedance for active digital circuits is a function of frequency, time, and the processed instruction set. Since there is a correlation between the switching activity and the impedance side-channel, the underlying method to detect hardware anomalies on a board relies on measuring the equivalent impedance from various locations on the board [8]. We collect the impedance measurements for different logic instructions and see the impacts to determine the correlation. We determined that impedance, resistance, and reactance were the most reliable parameters to help characterize the normal and abnormal behaviors in UAVs. Hence, we leverage these in building our intrusion detection system (IDS). The impedance was measured using the VNA in real-time and saved to a dataset. Hence, we used benign and attack data to train our machine learning-based classifier to help learn this characterization. We evaluated our classifier on various scenarios, e.g., trained with only benign and both benign and malicious data and varying the feature parameters in order to optimize the model's performance in detecting an HT attack in UAVs.

Overall, this study aims to develop a side channel-based IDS against HTs on mission-critical UAVs that may selectively alter the PWM signal or have other ramifications. The main contributions of this study are as follows:

- Determined a feasible side channel that can be used to construct an effective IDS and demonstrated how changes in impedance could lead to the discovery of an HT.
- Demonstrated how machine learning classification tools can be used to detect HTs on UAVs.
- Show how our proposed method can be more optimal and effective than other HT detection systems.

The rest of the paper is organized as follows: We provide an overview of UAVs and the impact that HTs have on UAVs and IDS applications in detecting HTs in Section II. Then, we introduce the problem definition and attack model, along with a clear understanding of the attacker's goal in Section III. Section IV goes over the validation of the HTs activation frequency to ensure that it activates at our desired pace. Then, we discuss our experiments and observations in Section V before proceeding to our evaluations in Section VII. Finally, we provide a brief literature review in Section VIII before our conclusion in Section IX.

II. Background

Here, we provide a background on how UAVs work, how HTs impact UAVs, and how an IDS can detect HTs.

A. Working Principle of UAVs

UAVs contain several components that allow them to function as intended by the manufacturers. One of the most critical components is the flight controller; flight controllers ensure that PWM signals are delivered to the appropriate motors. Flight controllers have other functions, such as coordinating sensors and power systems. The UAV platform has five parts: the battery module, a differential power system, a wireless communication and positioning system, and the main control module [9], which all work collectively to help determine the precise attitude for the UAVs.

The main control module on the platform used in this study is the Pixhawk 2.4.8, a 32-bit flight controller. Sensor systems include several gyroscopes and IMUs, which can directly deal with more accurate posture and location information. The battery module is responsible for the power supply to each module. This module converts and regulates voltages for the flight control and peripheral modules. The wireless communication and positioning module includes a data transmission, receiver, and a GPS module. The data transmission module is mainly responsible for data communication between flight control and upper ground stations. The GPS module provides global positioning information for flight control. UAV flight motors are usually connected to the flight control through electronic speed controllers (ESC). For this study, we focused on the flight controller, which provides the PWM signal instructions. This is where our testbed HT will be implemented.

B. Impact of Hardware Trojans on UAV Flights

HTs can have catastrophic impacts on the performance of UAVs. These Trojans can leak sensitive information, turn off critical portions of the IC, self-destruct the chip, or hinder performance. An example of this is as follows. A PWM is a digital signal which is used in UAV control circuitry. The PWM-generated signal remains high for a certain amount of time, known as on-time. On the other hand, the period when the signal remains low is known as off time. The on-time and off-time of a signal control the speed of a motor.

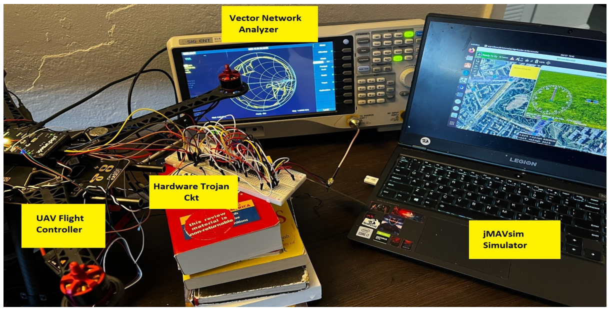


Fig. 1. Experimental setup of establishing an appropriate side-channel for detecting HTs in UAVs.

C. IDS Applications in the Detection of HTs

An IDS is a system capable of detecting anomalies in a device's operation. IDS are commonly used in network infrastructures to detect unauthorized intrusions. There are many types of IDS, but one of the most common and effective ones is signature-based. These types of IDS look for specific behavioral patterns to detect malware, intrusions, and other anomalies. Signature-based IDS suffers from the huge number of signatures stored in its database. Some researchers provided the concept of frequent signature databases to solve database size problems but never discussed how to deal with new and old signatures that became unnecessary [10].

However, for purposes of this study, we do not require a large database of signatures, and due to the erratic behavior of activation frequencies found within HTs, a signature-based IDS would be the most feasible option since we are looking into behavioral patterns of impedance. Our testbed setup, consisting of a UAV flight controller, the HT circuit, a VNA, and a jMAVsim simulator, can be seen in Figure 1.

III. Attack Model

This section illustrates the threat model that is used for conducting our tests. Our threat model consists of an HT capable of inverting the PWM signals on UAVs, along with a description of how the attacks are carried out. HTs are increasingly becoming a significant threat that risks the security of the modern Integrated Circuits (ICs) industry. An adversary intentionally modifies the IC design to cause operational failure, denial of service, or leaking of valuable information from the IC. To facilitate research work in this area, a better understanding of what various types of HTs would look like and their impact on an IC design is essential.

A. Attack Assumptions

The framework consists of the following assumptions:

- Attacker has complete knowledge of flight controllers, FPGA, IC and HT design.

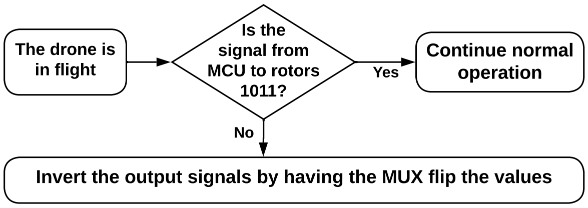

The drone is in flight
Is the signal from MCU to rotors 1011?
Yes
Continue normal operation
No
Invert the output signals by having the MUX flip the values

Fig. 2. HT activation flowchart for our HT design.

TABLE I
HT COMPONENTS

CDx4HC21 Dual 4-Input AND Gates	x1
CD4053B CMOS Single 8-Channel Analog Multiplexer	x2
CD4011B CMOS NAND GATES	x1

- We assume that the HT can be implemented at any phase within the supply chain, and the HT can go undetected.
- We consider the attacker to have access to the appropriate tools and components that are necessary for HT design.

B. Attack Technique

In this threat analysis framework, we will implement an HT between the flight controller and the motors of the quadcopter drone. An HT is composed of a trigger circuit and a payload circuit. A good design of the trigger and payload is crucial for this study's effectiveness, as it controls the HTs' behavior and allows us to determine later whether our IDS can detect stealthy designs. We define the following two circuit utilities:
Trigger Circuit: A trigger circuit consists of a triggering condition capable of switching the payload circuit on or off. The trigger circuit only activates the HT payload during specific case scenarios. This controlled environment allows the malicious payload to be executed in precise moments. Thus, it provides a layer of obfuscation in which the user may not suspect anything wrong until it's too late. A well-designed and sophisticated trigger circuit makes the HT much more stealthy and dangerous. The main requirement for the trigger is not to be activated continuously because it can be detected from the manufacturer's functional tests. The second mechanism, the payload, implements the effective function of a Trojan [11].
Payload Circuit: In this circuit, the malicious phenomena are executed upon being triggered. This is the most essential part of the HT, where the malicious instructions are stored and carried out. A series of multiplexers that invert the PWM signal is an example of a payload circuit that provides disruption.

C. Attack Goal

The attack goal generates parameterized attack procedures and functions that target a specific IoT device, in our case, a UAV. The attacker's main goal is to cause instability in the UAV's flight by inserting an HT that will selectively alter the duty cycle of the PWM signal being sent to the motors.

D. Discussion on the HT Design

HTs have great potential to be undetected if they can be inserted at places not directly protected by Trojan detection mechanisms [12]. This would cause instability in the UAV rotors, potentially leading the device to crash or have difficulty navigating its intended flight course. Each input would correspond to a data channel in the PX4 flight controller responsible for sending instructions to the rotors; an illustration of the circuitry setup is provided in Figure 3. The trigger circuit is highlighted in blue, whereas the payload is the multiplexers. Another illustration is provided in Figure 2 for the activation process. If an HT is always active, i.e., the payload is always executed, the purpose of stealth will be defeated; hence, an optimal trigger should be selected to remain stealthy yet achieve the attack goal. For our design, the trigger circuit's output, based on our logic table that follows the schematic, will be true if the values of the PWM signal are 1011, i.e., the PWM signals for the first, third, and fourth motors are active while that of the second is not active.

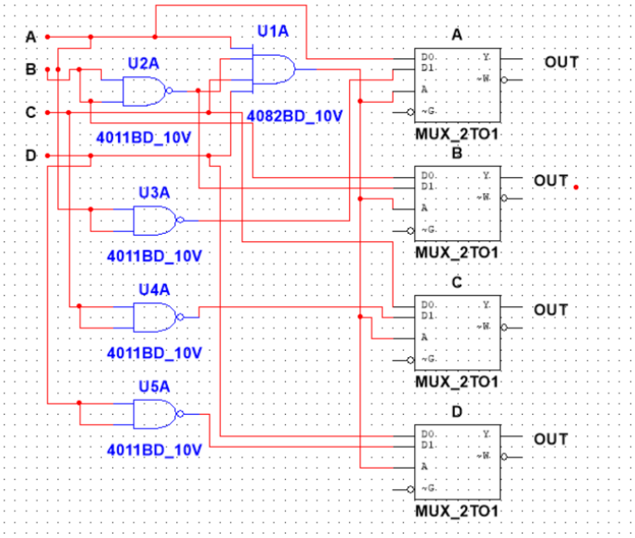


Fig. 3. Our HT design to be implemented on the UAV.

Another design that was used as part of our evaluation had a payload that would leak information regarding the sensor data. This would allow attackers to gain valuable insights into the UAV's surroundings and operation, as well as discover possible side channels that can later be exploited in future attacks. While this setup may appear to be simplistic by design, the impact it has on the PWM signals is significant enough to cause catastrophic damage. The impact of HT on the UAV's flight by our design can be further understood from the case study in section VII of this paper. Due to the trigger circuit activation frequency, our design can be difficult to detect without analyzing its side channels. [13] and [14] both provide effective stealthy HT detection methods that use a simple design. This demonstrates that a complex HT design is unnecessary in innovating effective countermeasures.

IV. HARDWARE TROJAN ATTACK IMPLEMENTATION

This section provides a detailed overview of IC components used to develop the circuit containing the hardware Trojan. An initial design for the HT did not work as intended. The initial design utilized a CD74HC04 Hex inverter. However,

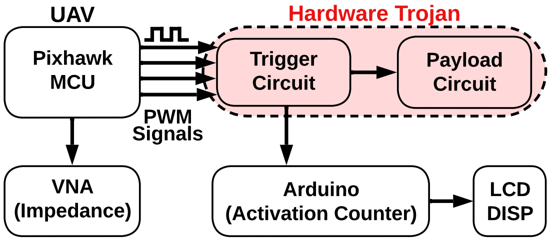


Fig. 4. HT validation setup using LCD and Arduino UNO.

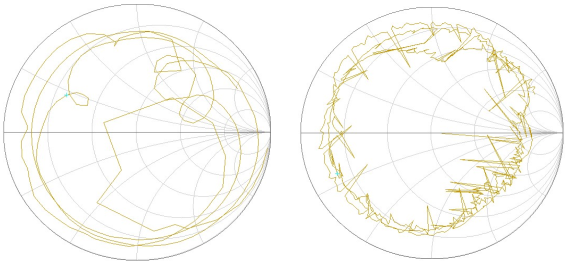
the hex inverter was incompatible with our design due to its parasitic properties. Due to insufficient voltage made it much more difficult for the payload to be delivered to the motor ESC. The current design replaces the hex inverters with NAND gates. We can have it output the opposite value by shorting one input for each NAND gate. This value inversion is crucial for the development of the payload circuit of the HT. The Pixhawk flight controller has male pins on its side that are meant for the user to connect their motor power module. These pins transmit a PWM signal, which indicates instructions for the motor. Using a female-to-male cable, we can redirect the PWM signal to our trigger circuit, as shown. The output from our multiplexers is then sent to the four ESC motor power modules. It is important to note that the PWM signal remains unchanged if the trigger circuit is inactive.

A. Side Channel Analysis

Active digital circuits exhibit a time-varying voltage and current draw dependent on the internal instruction cycle. The device impedance for active digital circuits is a function of frequency, time, and the processed instruction set. Instead of targeting the primary algorithm or data directly, side-channel analysis focuses on the operational characteristics and properties of a system, such as its power consumption, EM radiation, timing patterns, or even sound; this is then used to infer sensitive information that would usually be considered confidential, such as encryption keys. Cryptography should be designed securely against power analysis attacks so that attackers cannot extract its security key [15].

B. Validation of HT Activation Frequency

To assimilate an accurate understanding of how the HT operates, we first needed to gather data about how often the HT gets activated at separate time intervals. To capture this data, we used an ARDUINO UNO, which takes in the output from the multiplexers of the payload circuit as a new input. We also used an LCD screen and programmed the Arduino so that whenever the HT gets activated, it displays a value on the screen. The value is an incremental counter that adds one whenever the HT starts. We can then fetch the data using the Data Streamer feature on Microsoft Excel and save the results onto a spreadsheet to produce a graph. We can start gathering data after making the necessary connections and uploading the code to the microcontroller. The data was collected for a maximum of 100,000 cycles using the data streamer utility. This test ensures that the HT is activating less than 6.25 percent of the time. This is to ensure that our model accurately depicts a stealthy HT. Our first test consisted of letting the model run for 10 minutes. This resulted in an activation rate of 3.46 percent. Table II illustrates the activation rate values. Based on these results, we can discern that the HT operates within its permissible activation rate.



(a) (b)

Fig. 5. (a) VNA impedance results without HT. (b) VNA impedance results with HT. The presence of irregularities here is indicative of the presence of HTs.

TABLE II
HT Execution Validation

Timeframe	Number of Cycles	HT Activation Frequency	HT activation Rate
10 minutes	2251	78	3.46%
1 hour	7237	451	6.23%
1 hour repeated	11954	502	4.19%

V. Case Study

Here, we demonstrate the experiments that were carried out to validate the relationship between impedance and abnormal behavior. We also discuss our observations and results.

A. Experiments

As a part of this study, we were interested in knowing the effects and impact of the HT on the UAV. We used a simulation environment provided by Pixhawk called jMAVSim to accomplish this task. Using jMAVSim and observing the results stored in the logs, we observed a relationship between the HT activation and the thrust of the UAV. These fluctuations in thrust affect the overall altitude control of the drone and, when subsequently activated, cause the drone to crash. To perform HT detection using machine learning models, an essential first step is to extract and quantify distinguishing features for characterizing Trojan logic [16].

Our deep learning model, consisting of an XGBoost classifier (XGB), provided promising results in detecting HTs on UAVs. We then plotted the predicted and actual values for the training dataset. Our values are stored and loaded onto XGB, which generates an output of 1 for each row. This indicates that the HT is activated for each row that contains a 1. Conversely, if a row has a 0, the HT remains inactive, meaning the three parameters are within normal operational ranges, and no HT is present. Remember that we are dealing with an inverter HT; these results may differ depending on the Trojan used. We used Smith charts to visualize impedance patterns comprising real and imaginary components, resistance, and reactance. For this reason, it has become an industry standard in RF engineering

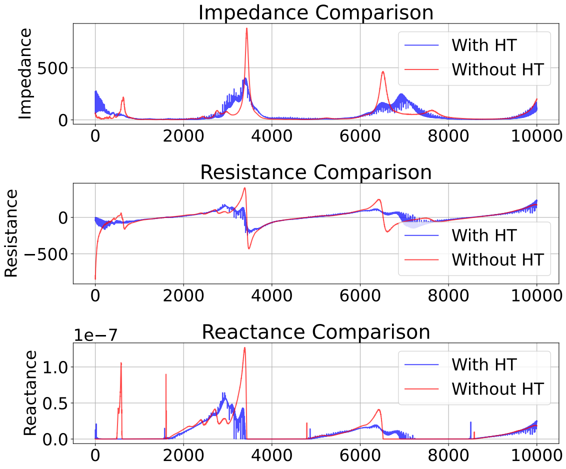


Fig. 6. Behavior of the parameters with and without a HT.

to use Smith charts to analyze impedance. An experiment demonstrated that a combination of impedance, resistance, and reactance is a good input for our IDS model. This was done by comparing different physical parameters of the DUT operation, which were also gathered by the VNA, such as resistance, phase, reactance, and power consumption. We also set up different payloads for the HT, which leaks sensor data rather than invert the PWM signals. This experiment is further evaluated in Section VII-B.

B. Observations

One observation made relates to the IDS performance when using different ML classifiers. Our classification results demonstrate a high level of precision and accuracy. Our research indicates that combining deep learning and classification models can be integrated to detect and mitigate hardware attacks on UAVs. Precision is vital to us because it shows the quality of the model's predictions. This quality can be calculated as the ratio of the total number of true positives over the sum of true and false positives. Conversely, recall is also beneficial in our analysis because it demonstrates sensitivity and specificity. Observing the score is also essential because it measures the test's accuracy. Without a good F1 score, the results for the precision and recall would lack credibility.

Looking at the two Smith charts generated by the VNA, graphs show a clear difference when an HT is present. We notice much more erratic behavior in the impedance during the presence of an HT due to the disruption it causes and the additional logic-switching activities that aren't considered in the manufacturing process of the DUT. We can also see differences in impedance, resistance, and reactance.

VI. Proposed Defense Technique

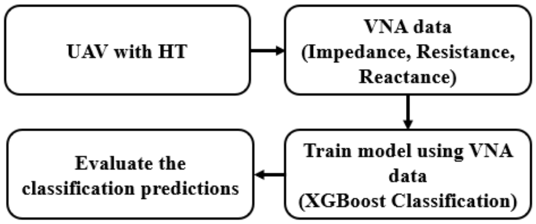


Fig. 7. Flowchart illustrating the process flow of the proposed IDS.

Our defense model proposal utilizes a neural network with XGB to detect an HT effectively in UAVs. Based on the results gathered by the experiments in our case study, the best XGB inputs are three parameters: impedance, resistance, and reactance. We can train the time series model using these three parameters and provide a set of predicted values for each. This allows us to use XGB later to categorize each accurately predicted parameter combination as benign or malicious. We selected XGBoost for our UAV IDS because it excels in processing complex data, crucial for identifying hardware Trojans through parameters like impedance and resistance. Its robustness against overfitting and ability to highlight key features align with our IDS requirements. Additionally, XGB's efficiency and adaptability are crucial for potential real-time IDS applications in UAVs, offering a scalable solution that remains effective against the dynamically evolving landscape of HT threats. Using this method, we could predict the timing of the trigger circuit activation for the HT. This allows researchers to look into mitigation strategies and countermeasures that can be put in place that consider the HT activation timing. The data we collected using the VNA is a time series, as it provides new parameter values for each frequency sweep. Frequency sweeps were set to occur every 65 milliseconds.

Therefore, we would obtain fresh values for impedance, resistance, and reactance after every sweep. By comparing the HT and HT-free models using the integrated Smith chart, we observed significant changes to the UAVs' impedance, resistance, and reactance levels, as shown in section VI. We chose the XGB classification model for its high accuracy and robustness. As a part of our dataset, we must collect many power traces (10002, to be exact). The XGB allows us to avoid over-fitting and handling complex, high-dimensional data. It provides a higher accuracy rate than other models; this is very important to consider due to the nature of the application. The XGB also provides feature importance, allowing us to understand which features contribute the most to the classification. Figure 7 shows the flowchart that depicts our IDS process, where data from a UAV with a hardware Trojan is analyzed for impedance, resistance, and reactance using a VNA, and then XGBoost classification is employed to train the model on this data, subsequently evaluating the predictions to determine the presence of an HT. By using XGB, we can develop a side channel-driven IDS that we could use for UAVs. This impedance-side channel-based IDS can detect the presence of HTs through impedance patterns. A real-time application would require a method of gathering impedance data while the UAV is in operation, which does not rely on a VNA. It is currently out of scope but holds potential for future work.

Discussion on Practical Implementation of the Proposed IDS: To practically implement our proposed IDS on board the UAVs for detecting HTs, a design focused on resource efficiency, real-time operation, integration with UAV control systems, and adaptability is essential. The IDS should be lightweight to accommodate UAVs' limited computational and power resources, employing algorithms that are efficient yet effective in detecting anomalies associated with HTs. Real-time detection capabilities are crucial for timely identification and mitigation of threats, necessitating the IDS to process data quickly and with minimal latency. Seamless integration with the UAV's existing control systems ensures that the IDS can monitor relevant system behaviors without disrupting normal operations. Lastly, the system must be adaptable to various UAV configurations and capable of updating its detection algorithms to counter new and evolving threats, ensuring long-term viability in diverse operational environments.

Considering the resource costs and hardware options for implementing an IDS in UAVs, it's essential to evaluate computational efficiency, energy consumption, and physical integration constraints. Field-programmable gate Arrays (FPGAs) are an optimal choice for IDSs as they balance computational power and energy efficiency. FPGAs allow for the execution of complex detection algorithms, including machine learning models, with greater flexibility and lower power consumption compared to traditional microcontrollers (MCUs) or general-purpose processors. This adaptability is crucial for maintaining the IDS's relevance against evolving threats while adhering to the stringent resource limitations of UAV platforms. Application-specific integrated Circuits (ASICs) represent the high end of performance and energy efficiency for fixed-functionality IDS implementations but lack the flexibility of FPGAs, which renders them less ideal for environments where threat dynamics frequently change. Although powerful, Graphics Processing Units (GPUs) are generally unsuitable for most UAVs due to their significant energy requirements. Thus, FPGAs strike the most effective balance for IDS implementations in UAVs, providing the necessary computational resources for advanced security measures without compromising the UAV's operational efficiency or mission duration.

VII. EVALUATION

This section presents the HT model findings and the feasibility of implementing our proposed framework. We evaluate the proposed IDS considering the following research questions.
RQ1 Can the IDS detect the presence of HTs of varying activation frequencies?
RQ2 How effectively is our IDS detecting different HTs?
RQ3 Do different parameter variations improve results?
RQ4 How effective is the side channel-based defense technique in detecting the HT at different thresholds?
RQ5 What is the computational load?
RQ6 How feasible is it to implement the IDS model?
RQ7 How do different classification models perform, and which one is most optimal?

A. Performance with Varying Activation Frequencies [RQ1]

While our IDS tends to perform well with a 1/16th of a time activation frequency, we wish to evaluate our performance with other frequencies. Based on our observations, it becomes clear that the ROC Score tends to decrease with higher activation frequencies. The results demonstrate that our IDS may be less effective on extremely low activation frequencies due to the possibility of it detecting too many type I and type II errors. As shown in a similar study, When a Trojan affects the behavior of the circuit rapidly, the detection of Trojan becomes easier [17].

This experiment was done by comparing the IDS results at different activation frequencies. This experiment aimed to determine how the IDS performed at varying trigger circuit activations, which is essential to consider as HTs tend to all have different trigger mechanisms. Unlike other studies, our results demonstrate that our IDS performs well even with varying activation timings and detects HTs even when the activation frequency is less than 3 percent of the time.

B. IDS Performance with Different HT Payloads [RQ2]

To test whether our IDS would work for different HTs, We added a temperature, humidity, and light sensor to our HT for this test. We used an Arduino UNO to mimic a microcontroller that can send this sensor data over to an SQL database only when the trigger circuit of the HT is activated. The purpose of this is to simulate a scenario where there is an information leakage payload. Using the new parameter dataset collected for this HT Payload, the IDS performed reasonably well in detecting the presence of an HT. While the ROC Score could be better when compared to our original design, it still yielded a score of 0.97, as shown below. With some minor tweaking, this IDS model can be applied to different types of HT payloads. The classification model was XGB because it yields the highest performance. This experiment was needed to understand the effectiveness that our IDS has for other HT designs. The performance results also demonstrate that this IDS can detect HTs using different payloads.

C. Performance with Varying Feature Combinations [RQ3]

We evaluated the results using different parameters to understand which worked best with our IDS. We do this by comparing the ROC Score values for each. Upon evaluation, it is clear that the best parameter combination that provides the highest ROC score is the one that considers the impedance, resistance, and reactance. The second best-performing parameter setting is the one consisting of impedance and resistance. The parameter that provides the lowest ROC score is reactance, whereas the one that provides the highest is Impedance. This was done by manually testing each parameter combination and observing the results individually. This is important to determine the best possible combination for a precise and accurate IDS. One advantage of these findings over other proposed IDS methods for HTs is that they can be universally applied, as impedance, resistance, and reactance are all expected consequences of the physics of a UAV operation.

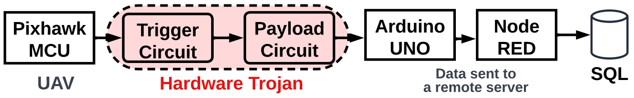


Fig. 8. Information leakage of new HT flowchart.

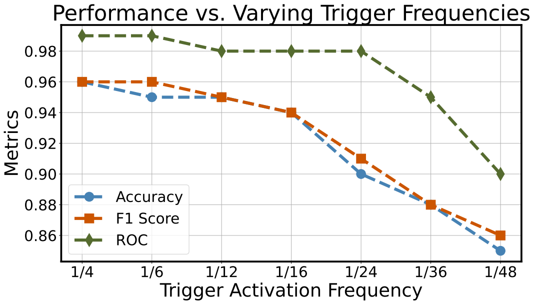


Fig. 9. IDS performance with varying trigger activation frequencies.

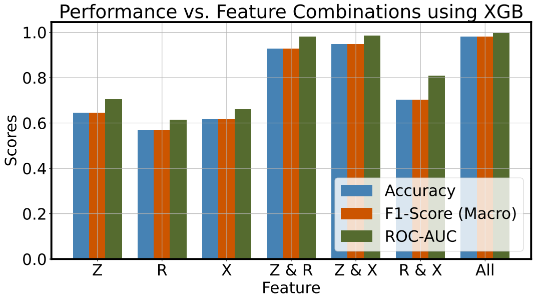


Fig. 11. Different feature combination performance results. Z: impedance, R: resistance, X: reactance.

D. Varying Threshold Performance Evaluation [RQ4]

We evaluate the performance of the IDS using different threshold settings for the XGB. Our results demonstrated a negligible impact on the IDS performance when using different threshold settings for the XGB. Specifically, the higher the threshold, the lower the ROC Score. We were able to determine that the optimal threshold setting is within 0.6. A threshold setting of 0.6 is suitable for detecting HTs because it requires a fair balance between precision and recall. Higher thresholds lead to higher precision but lower recall. Lower thresholds lead to higher recall but lower precision. In our case, higher precision would be more critical to minimize false detection. In contrast, a higher recall would allow us to detect HTs even with false positives. We wanted to prioritize precision, so our threshold was set to 0.6. The thresholds were manually configured within the classification models script. The results for each threshold were then compared to determine the appropriate threshold value.

E. Computational Load and Ease of Implementation [RQ5, RQ6]

Computational load is essential to consider when implementing an IDS on a UAV. Heavy computational loads could increase the time it takes for the IDS to detect an HT; on mission-critical UAVs successfully, this would pose a problem as they require a fast response time. For this section, we will calculate the time it takes for the IDS to run and how much space it occupies in memory. We calculated the process time in milliseconds using Python's 'time' library. On average, it takes roughly 86ms and occupies 197MB of memory. Due to these results, it is safe to assume that this IDS implementation method is feasible as modern computational power exceeds 197MB of memory. Although some UAVs will require the IDS to perform their functions on an edge node within a network due to the limiting specifications of flight controllers if the IDS were to be running while the UAV is in operation. However, if we were to assume that our IDS would run before each flight mission, we wouldn't need additional resources. Due to implementation ease and its cost-effectiveness, this proposed IDS technique is much more feasible than current methods.

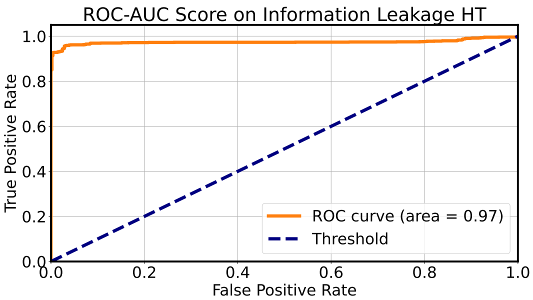


Fig. 10. ROC score for the performance of side-channel driven IDS on information leakage HT.

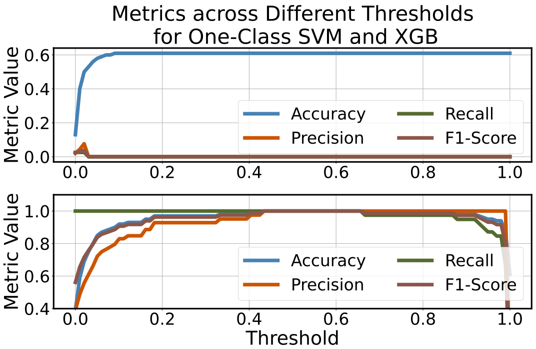
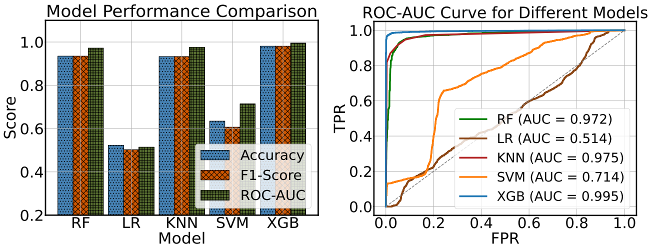


Fig. 12. Performance metrics for different thresholds.

F. Performance Evaluation of Classification Models for our IDS Technique [RQ7]

For this evaluation, we tested the performance of our IDS using different ML classification models to see which one performed optimally. Our performance metrics consisted of Accuracy, F1-Score, and ROC-AUC. These three metrics were measured for the following classification models: Random Forest, Linear Regression, K-Nearest Neighbor, Support Vector Machine, and XGboost. Based on our results, XGBoost provided the highest performance at correctly classifying whether or not there is an HT based on our input parameters of impedance, resistance, and reactance. The second and third best classification models are that of K-Nearest Neighbors and Random Forest. The classification models that performed the worst were Linear Regression and Support Vector Machine. We could see an illustration of our performance for different ML classifiers in Figure 13(a) and Figure 13(b). Using a separate dataset, we also evaluated and compared the ROC-AUC curves for the different classification models and obtained the same observations as our previous tests. Based on these results,



(a)

(b)

Fig. 13. Performance comparison for different ML classifiers.

we decided to stick to XGBoost as our side channel-based intrusion detection system classification model.

VIII. Related Work

This section provides a brief literature review on similar techniques and findings that have helped develop this IDS.

A. *Side Channel Techniques for HT Detection*

A paper that describes a side channel technique for HT detection using thermal maps demonstrates that side channels can effectively detect counterfeit circuits. The researchers of this study performed post-silicon power mapping on residual thermal maps to see HTs. The critical problem faced with this study is that sometimes, the heat generated is below the required resolution for the heat detector. Due to this issue, the proposed method could fail at detecting specific HTs that activate less frequently. Another study demonstrates how a backscattering side channel can also be used to detect HTs. These channels can be created by propagating a continuous-wave signal toward the chip. The transistor switching activities cause changes in the chip impedance, which modifies the circuit's radar cross-section (RCS). The changes will be reflected in the backscattered signal, which is beneficial to detecting HTs [18].

There are many advantages to using the backscattering side channel. These advantages are higher bandwidth, signal strength unaffected by leakage, and an adaptable frequency. The issue with these methods is that they are hard to implement due to the cost of necessary equipment and the expertise and skill required for the implementation and result analysis. Unlike these studies, this paper demonstrates an ease of implementation method that is also cost-effective. Our method does not rely on too much equipment or knowledge beyond that of junior-level programming experience. Due to these advantages, our side channel-based IDS can be widely integrated into many applications. Despite the isolation mechanisms available to cloud service providers, like virtual machines and containers, the problem of side channel vulnerabilities due to shared caches and multicore processors remains a threat [19].

This paper demonstrates that side channels can be protected by isolating cloud processes when multicore chips share the same caches. This was accomplished through a Co-scheduling technique. The kernel can invoke CLFLUSH on the entire virtual address space. While this is guaranteed to work across processor generations, this approach is too costly, taking up to 10x the kernel timeslice on applications tested [19]. Our approach is much more feasible as it does not impose a heavy burden on the computational load. Another study presents a novel representation of cyber-physical systems wherein the states of the physical system are incorporated into the cyber system and vice versa. Next, using this representation, optimal strategies are derived for the defender and the attacker using the zero-sum game formulation, and iterative Q-learning is utilized to obtain the Nash equilibrium [20]. This type of defense takes advantage of the Q-learning model instead of our XGB model. Using this model, the researchers found relationships between physical and cyber systems.

As a result, the cyber system states affect the biological systems' controller design and vice versa. Unfortunately, this approach would not be practical for UAVs because we cannot risk leveraging either side of the system, physical or cyber. For this reason, the decision to use an XGB rather than a Q-learning model was made. Our results demonstrate a high level of accuracy without imposing any risk to the physical or cyber components of our UAV.

B. *Machine Learning for Anomaly Detection*

Several studies demonstrate that anomaly detection can be performed using ML classifiers. One study concluded that the number of layers and cells are essential parameters, alongside window size and threshold. One of the best machine learning methods is autoencoder-based anomaly detection. An autoencoder aims to understand a representation (encoding) for a set of data, typically for dimensionality reduction, by training the network to ignore signal "noise" [21]. Traditional anomaly detection techniques include isolation forest algorithm, local outlier factor detection algorithm, one-class support vector machine algorithm, and statistical model [22]. The isolation forest algorithm distinguishes outliers by establishing isolation trees and calculating anomaly scores, which can effectively determine and detect isolated outliers. Still, it is not applicable, particularly for high-dimensional data [23]. The deep feature extraction capability and the efficient massive data processing of ML algorithms make them stand out in detecting time series anomalies [24]. Contrarily, our model does not use autoencoding as it only uses XGB. This study demonstrates how using. XGB to detect HTs provides exceptional results.

C. *Automated Techniques in Hardware Trojan Detection*

There has been some research that looks into the possibility of having an automated method in hardware Trojan detection. One particular paper is [25], which incorporates an assertion-based validation method in order to automate the detection of hardware Trojans. This method provides a solution that is scalable, due to its linear memory requirement, as mentioned on [26]. Because [25] uses reinforcement learning, it differs from our proposed method which uses supervised learning. While it may be beneficial to have an automated technique in detecting HTs, we deemed it to be unnecessary to demonstrate our proof of concept. This is because it would be unfeasible to measure impedance while a UAV is in a flight mission due to the many factors that may impact the quality of the results. Measuring impedance is a very sensitive task, and something as simple as wind would produce false readings. Due to this reason, we elected to propose a method that can detect hardware Trojans while the UAV is stationary and under supervision.

IX. Conclusion

In conclusion, this paper demonstrates the importance of effective countermeasures to ensure the security of mission-critical UAVs. We introduced HTs and their implications in UAVs. Through experiments, we determined a side channel to construct an IDS to detect HTs. We defined our problem and attack model by creating our testbed HT and inserting it into a UAV. Our defense proposal consists of an XGB classifier that takes in three side-channel parameters, impedance, resistance, and reactance, to detect the presence of HTs. Our results demonstrated that by using ML, we can efficiently detect HTs on UAVs. For future studies, we would like to see if we can develop other side channel-based techniques for different kinds of threats in UAVs. Moreover, our defense mechanism's versatility underscores the broader potential of ML in ensuring UAV security against evolving threats. For future works, we would like to demonstrate the possibility of an IDS technique that does not depend on machine learning. We could also benefit from a node insertion analysis, to study the impact that the HT may have depending on its location of implementation.

X. Acknowledgment

This research was supported in part by the National Security Agency (NSA) under award H98230-22-1-0327, and the Department of Energy (DOE) under awards DE-CR0000024 and DE-NA0004016.

References

[1] Joshua Gordon, Victoria Kraj, Ji Hun Hwang, and Ashok Raja. A security assessment for consumer wifi drones. In *2019 IEEE International Conference on Industrial Internet (ICII)*, pages 1–5, 2019.

[2] Krisztián Bálint. Uavs with biometric facial recognition capabilities in the combat against terrorism. In *2018 IEEE 16th International Symposium on Intelligent Systems and Informatics (SISY)*, pages 000185–000190, 2018.

[3] Luís Gonçalves and Bruno Damas. Automatic detection of rescue targets in maritime search and rescue missions using uavs. In *2022 International Conference on Unmanned Aircraft Systems (ICUAS)*, pages 1638–1643, 2022.

[4] Kan Xiao, Domenic Forte, Yier Jin, Ramesh Karri, Swarup Bhunia, and Mohammad Tehranipoor. Hardware trojans: Lessons learned after one decade of research. *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, 22(1):1–23, 2016.

[5] Yier Jin and Yiorgos Makris. Hardware trojan detection using path delay fingerprint. In *2008 IEEE International workshop on hardware-oriented security and trust*, pages 51–57. IEEE, 2008.

[6] Mohammad Ashiqur Rahman, Md Tauhidur Rahman, Mithat Kisacikoglu, and Kemal Akkaya. Intrusion detection systems-enabled power electronics for unmanned aerial vehicles. In *2020 IEEE CyberPELS (CyberPELS)*, pages 1–5, 2020.

[7] Abhijitt Dhavlle, Rakibul Hassan, Manideep Mittapalli, and Sai Manoj Pudukotai Dinakarrao. Design of hardware trojans and its impact on cps systems: A comprehensive survey. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–5, 2021.

[8] Huifeng Zhu, Haoqi Shan, Dean Sullivan, Xiaolong Guo, Yier Jin, and Xuan Zhang. Pdnpulse: sensing pcb anomaly with the intrinsic power delivery network. IEEE, 2023.

[9] Jiabao Song, Haiwen Yuan, and Bulin Zhang. Pixhawk-based scalable platform for multi-purpose surface unmanned vehicle. In *2022 4th International Academic Exchange Conference on Science and Technology Innovation (IAECST)*, pages 1577–1580, 2022.

[10] Abdullah H Almutairi and Nabih T Abdelmajeed. Innovative signature based intrusion detection system: Parallel processing and minimized database. In *2017 International Conference on the Frontiers and Advances in Data Science (FADS)*, pages 114–119, 2017.

[11] Fotios Kounelis, Nicolas Sklavos, and Paris Kitsos. Run-time effect by inserting hardware trojans, in combinational circuits. In *2017 Euromicro Conference on Digital System Design (DSD)*, pages 287–290, 2017.

[12] Georg T. Becker, Ashwin Lakshminarasimhan, Lang Lin, Sudheendra Srivathsa, Vikram B. Suresh, and Wayne Burelson. Implementing hardware trojans: Experiences from a hardware trojan challenge. In *2011 IEEE 29th International Conference on Computer Design (ICCD)*, pages 301–304, 2011.

[13] Qazi Arbab Ahmed, Tobias Wiersema, and Marco Platzner. Proof-carrying hardware versus the stealthy malicious lut hardware trojan. In Christian Hochberger, Brent Nelson, Andreas Koch, Roger Woods, and Pedro Diniz, editors, *Applied Reconfigurable Computing*, pages 127–136, Cham, 2019. Springer International Publishing.

[14] Yu Su, Haihua Shen, Renjie Lu, and Yunying Ye. A stealthy hardware trojan design and corresponding detection method. In *2021 IEEE International Symposium on Circuits and Systems (ISCAS)*, pages 1–6, 2021.

[15] Yilin Zhao, Qidi Zhang, Hiroki Nishikawa, Xiangbo Kong, and Hiroyuki Tomiyama. Power side-channel analysis for different adders on fpga. In *2021 18th International SoC Design Conference (ISOCC)*, pages 367–368, 2021.

[16] Lingjuan Wu, Xuelin Zhang, Siyi Wang, and Wei Hu. Hardware trojan detection at lut: Where structural features meet behavioral characteristics. In *2022 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 121–124, 2022.

[17] Nilanjana Das, Joy Halder, Baisakhi Das, and Biplab K Sikdar. Exploring hard to detect sequential hardware trojans. In *2020 24th International Symposium on VLSI Design and Test (VDAT)*, pages 1–6, 2020.

[18] Luong N. Nguyen, Baki Berkay Yilmaz, Milos Prvulovic, and Alenka Zajic. A novel golden-chip-free clustering technique using backscattering side channel for hardware trojan detection. In *2020 IEEE International Symposium on Hardware Oriented Security and Trust (HOST)*, pages 1–12, 2020.

[19] Read Sprabery, Konstantin Evchenko, Abhilash Raj, Rakesh B. Bobba, Sibin Mohan, and Roy Campbell. Scheduling, isolation, and cache allocation: A side-channel defense. In *2018 IEEE International Conference on Cloud Engineering (IC2E)*, pages 34–40, 2018.

[20] Haifeng Niu and S. Jagannathan. Optimal defense and control for cyber-physical systems. In *2015 IEEE Symposium Series on Computational Intelligence*, pages 634–639, 2015.

[21] Oleksandr I. Provotar, Yaroslav M. Linder, and Maksym M. Veres. Unsupervised anomaly detection in time series using lstm-based autoencoders. In *2019 IEEE International Conference on Advanced Trends in Information Theory (ATIT)*, pages 513–517, 2019.

[22] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 Eighth IEEE International Conference on Data Mining*, pages 413–422, 2008.

[23] Fei Tony Liu, Kai Ming Ting, and Zhi-Hua Zhou. Isolation forest. In *2008 eighth ieee international conference on data mining*, pages 413–422. IEEE, 2008.

[24] Riyaz Ahamed Ariyaluran Habeeb, Fariza Nasaruddin, Abdullah Gani, Ibrahim Abaker Targio Hashem, Ejaz Ahmed, and Muhammad Imran. Real-time big data processing for anomaly detection: A survey. *International Journal of Information Management*, 45:289–307, 2019.

[25] Zhixin Pan and Prabhat Mishra. Automated test generation for hardware trojan detection using reinforcement learning. In *2021 26th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 408–413, 2021.

[26] Yangdi Lyu and Prabhat Mishra. Automated test generation for activation of assertions in rtl models. In *2020 25th Asia and South Pacific Design Automation Conference (ASP-DAC)*, pages 223–228, 2020.