SAND2019-3828C

Supervising Communication SoC for Secure Operation: Machine Learning Classification of Operation

Abdelrahman Elkanishy*, Abdel-Hameed A. Badawy*‡, Paul M. Furth*,
Laura E. Boucheron*, and Christopher P. Michael†
Electrical and Computer Engineering, New Mexico State University, Las Cruces, NM 88003, USA
† Sandia National Laboratories, Albuquerque, NM 87185, USA
‡ Los Alamos National Laboratory, Los Alamos, NM 87545, USA

***Abstract*—Wireless communication protocols are widely-used in smart devices and systems. Manufacturers normally buy and/or fabricate their communication chips from third-party suppliers, which are then integrated into a complex hardware-software stack with a variety of potential vulnerabilities. Direct measurement of the output or power signals can prevent unauthorized data transmission. This work proposes a compact supervisory circuit to classify the operation of a Bluetooth (BT) SoC at low frequencies by monitoring the radio frequency (RF) output of the BT chip through an envelope detector. The idea is that we can inexpensively fabricate envelope detector, power supply current monitor, and classification algorithm on a low-frequency integrated circuit in a trusted legacy technology. When the supervisory circuit detects abnormal behavior, it shuts off power from the BT chip. We extract simple yet descriptive features from the envelope of the RF output signal. Then, we train machine learning (ML) models to classify different BT operation modes, such as separating BT advertising and transmit/receive modes of the BT chip. Our results show very high classification accuracy ($\sim 100\%$).**

***Index Terms*—Hardware Security, Supervisory Circuit, Bluetooth, Machine Learning, Security, RF Signals, Classifier.**

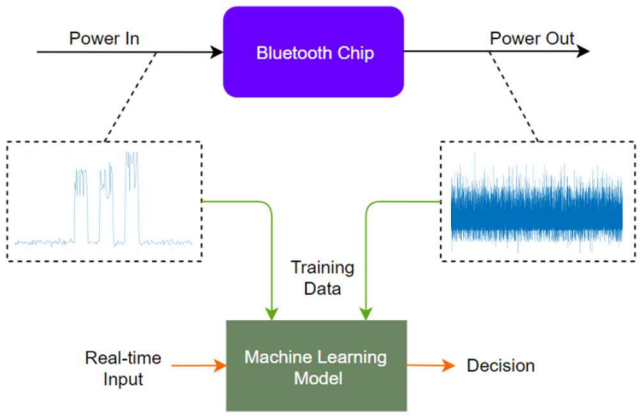

Fig. 1. Diagram of the supervisory circuit's idea. In which, the power signals during the normal operation are collected. Then, the ML model is used in real-time to verify the normal operation.

I. Introduction

Due to the complexity and multi-functionality of smart systems, most manufacturers outsource communication chips from third-party suppliers. The integration between many outsourced ICs has resulted in the need to add a hardware security layer to ensure appropriate operation. For example, in Apple's smartphones, there is a dedicated co-processor, Secure Enclave, to handle all cryptographic operations and maintain the integrity of data protection for the entire system [1].

BT, like any communication protocol, has vulnerabilities. For instance, in 2017, Armis [2] identified a new BT attack vector called BlueBorne (BB) that can take control of the target device. BB attacks regular computers, smartphones, and IoT devices. This security breach occurs without pairing to the targeted device nor even while the BT IC is in discovery mode. As the BT chip is responsible for establishing the connection and controlling the flow of data, BB and other security breaches could attack the BT IC without the consent of the controller chip. Therefore, monitoring a BT chip at the hardware level is necessary to verify its correct operation.

As shown in Fig. 1, one way to monitor the chip is to consider it as a black box which consumes and transmits power. Thus, abnormal behavior can be detected by learning the normal input/output (I/O) power signatures, for example. A second way is to parameterize aspects of the BT connection (e.g., profile type, the distance between paired devices, number of connected devices) and compare the detected behavior to the expected behavior based on the controller chip instructions. Supervisory circuits are commonly used in detecting power failures but are not common for security purposes [3]. Recently, PFP Cybersecurity [4] has partnered with XILINX to detect security breaches in XILINXs devices using artificial intelligence. Their work is focused on self-monitoring, not monitoring another IC, and is intended to monitor XILINX devices only.

Background: A BT device can broadcast data to all nearby devices, or it can establish a secure connection to particular devices. Generic Access Profile modes and procedures are responsible for authorization of a connection [5]. First, the advertising process takes place in which the slave device is set to discovery mode in order to be noticed by the master device. During this process, in each specific advertising interval, the slave re-transmits advertising packets. On the master side, the device begins the general discovery procedure which scans for slaves' advertising packets, then lists the available devices [5], [6]. At the application layer, the master device chooses the desired device to which to connect [6]. Once the connection

Sandia National Laboratories is a multimission laboratory managed and operated by National Technology & Engineering Solutions of Sandia, LLC, a wholly owned subsidiary of Honeywell International Inc., for the U.S. Department of Energy's National Nuclear Security Administration under contract DE-NA0003525.

is established, Generic Attribute Profile (GATT) describes the transfer of the data through the devices in a handshaking process. We can categorize the transmitting states into two states: advertising state, before establishing the connection, and transreceiving state, after the connection.

Proposed Supervisory Circuit: We are creating a supervisory circuit to detect abnormal operation of a complex, mixed-signal communication System-on-Chip (SoC). BT is the communications standard of choice for this work. We could use similar techniques to other communications protocols. We design a supervisory circuit that can operate at low frequency, low power, and inexpensive computationally. This will facilitate our ability to fabricate the supervisory circuit in an inexpensive circuit technology or integrate soft intellectual property (IP) into a more advanced SoC. When the supervisory circuit detects a security abnormality, the circuit can intervene and shut down the BT IC.

The supervisory circuit design is split into several major blocks, as shown in Fig. 2. First, the circuit that provides and controls power is comprised of a controlled low-dropout (LDO) voltage regulator. LDOs are widely used in portable communications systems since they occupy a small area, have low noise, and provide high transient performance. Embedded in the LDO is a current sensor that monitors the output current of the LDO. External to the supervisory circuit is an RF coupler, which splits the transmitted RF signal into the main path and a monitored path. The monitored path passes through the envelope detector, which lowers the frequency of the RF signal in order to be able to sample it at frequencies much lower than 4.8 GHz, the Nyquist rate of the 2.4 GHz BT signal. As such, the supervisory circuit can be entirely implemented using low-speed technology. The outputs of the current sensor and the envelope detector are digitized using analog-to-digital converters (ADCs). Finally, digital signal processing (DSP) circuits, or soft IP, will be used to extract the features from all relevant signals. At run-time, the system extracts the necessary features to feed into the Machine Learning (ML) models to determine what operation is running on the BT IC. In future work, We will compare monitored behavior to accepted behavior via the Control Bus shown in Fig. 2. More details about the supervisory circuit implementation is available here [7].

Related Work: Outsourcing ICs fabrication to third party manufacturers increases the possibility of an untrusted modification to the circuitry, i.e., a hardware Trojan, during production. According to Hasegawa's classification [8], a non-destructive, non-invasive hardware Trojan detection techniques can be classified as either run-time or test-time. Both of these approaches are based on comparing test IC parameters with a golden IC model i.e., the parameters obtained from a known Trojan-free IC. Test-time approaches use logic testing and/or side-channel parameters to inspect the IC before integrating them into a system. Even when combining logic testing and side-channel analysis [9], test-time approaches are still limited, since attacks may only be triggered after deployment. Run-time hardware Trojan detection methods aim to monitor the

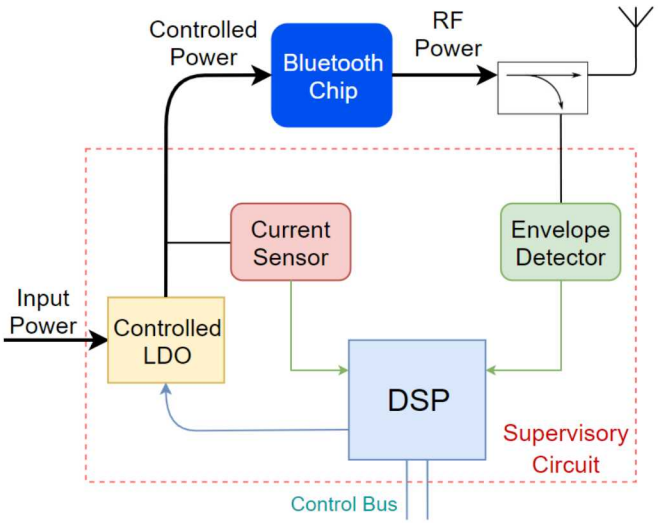

Fig. 2. Block diagram of communication SoC monitoring system comprised of a digital signal processing (DSP) block, analog-to-digital converters (ADCs) and the two proposed supervisory circuits: an output-current-monitoring low-dropout voltage regulator (LDO) and an envelope detector. A classification algorithm is implemented in the DSP block.

chip continuously through the addition of monitoring circuitry. Bao et al. [10] use variations in temperature sensors readings to detect hardware Trojans. Hasan et al. [11] use formal verification as a framework to develop run-time hardware Trojan detection units for digital circuits. In this paper, we monitor a communication IC through RF and power input signals during run-time. Unlike the aforementioned hardware Trojan detection techniques, we are not focused only on the security of digital circuits. We propose additional hardware that can verify correct IC operation to detect abnormal operation that might be a hardware or a software attack in a mixed-signal communication SoC.

On-chip classification of IC behavior requires relatively simple computations. Iwase et al. [12] use a discrete Fourier transform feature of the voltage signal. Other classifiers select features from multiple domains [13], [14] after transforming the signal using both wavelet and/or Fourier analysis. Also, researchers used statistical features from both time and frequency domains [15]. Still, other approaches extract large numbers of features from signals, then apply computationally intensive dimensionality reduction techniques [16], [17]. In this paper, we are concerned with the computational complexity of the selected features and classification algorithms. Since frequency transformations require high computational overhead, the selected features are exclusively extracted from the time domain. We experimented with frequency domain features and we verified that they provide more computational complexity without any performance advantages. In addition, we selected novel features that are computationally smart, cheap, while achieving high accuracy ($\sim 100\%$).

II. Methodology

Prior to fabricating the supervisory circuit, we prototype our supervisory circuit using off-the-shelf components and

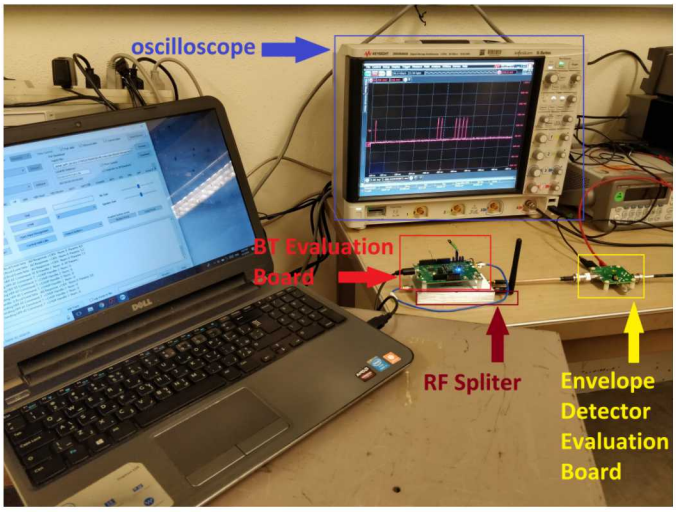

Fig. 3. The preliminary laboratory experimental setup showing the laptop, BT evaluation board, RF splitter, envelope detection evaluation board, and oscilloscope.

an oscilloscope in order to collect a data set sufficient for training and testing the classification algorithm. We placed small-valued series resistors on the power supply pins to the CYW20706 [18] BT SoC in order to monitor the supply current to each of the SoC's voltage domains. In addition, the RF output of the BT IC passes through an RF splitter. One side of the splitter goes to an antenna for pairing with other BT devices. The other side of the splitter is attached to an AN-2264 LMH2121 envelope detection evaluation board [19]. This particular envelope detector has an input bandwidth from 0.1 to 3 GHz which covers the BT band. The envelope detection stage lowers the frequency of the signal, so the Nyquist rate is decreased. This experimental setup is depicted in Fig. 3. A laptop controls the BT board via USB. The oscilloscope samples and saves the envelope-detected RF stream and input power signals.

The BT board is programmed to act as 2 popular profiles: hands-free and headset, in addition to customized profiles using GATT services. While each profile is running, different events occur, such as dialing, hangup, and streaming music. The events are controlled using a graphical user interface, as shown in Fig. 4, which utilizes a serial port through USB to send commands to the BT evaluation board. The network topology of two devices is defined. Moreover, we collect the RF streams of each profile in both the advertising and transmitting/receiving (transceiving) states. First, the hands-free profile RF output signal is recorded while executing multiple events, such as dialing, answering, and hang up. Second, the headset profile RF output signal is captured during various events, such as streaming music, rewind, scrub, and volume control. Lastly, a customized profile is used to simulate a simple embedded system which can be connected through BT communication. Basically, it notifies the BT evaluation board of a sensor reading to a paired device, which can control the number of blinks of a light-emitting diode.

The oscilloscope captures the RF envelope-detected signal

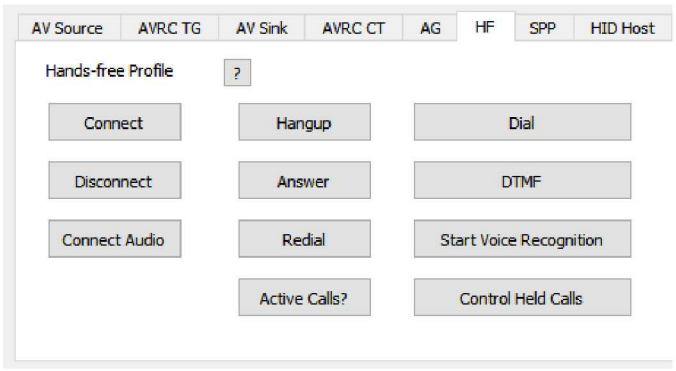

Fig. 4. The graphical user interface of the BT board while controlling the hand-free profile.

with a sampling frequency of 50 kHz. A processing window of 640 ms (corresponding to 32,000 samples) with one sample advance is selected to collect as many transmitting events as possible with a minimal computational load. Three features are extracted from each window to train the ML models. The first feature extracted is the maximum signal value in the given window, since the maximum signal value is expected to vary from one transmitting state to the next. Changes in the maximum value are related to the different profiles. As we are interested in the pattern of the BT transmission, the other two features are extracted after thresholding the envelope-detected stream into two binary levels. In other words, the signal is 1-bit quantized. Therefore, after quantization, value 1 means the BT is transmitting, whereas value 0 indicates no transmission. The remaining two features extracted are the area under the curve and the number pulses, both extracted from the 1-bit quantized signal in a given window. The area under the curve is correlated to the total transmission duration in a certain window, whereas the number of pulses represents the density of the transmission events in the window.

The data set is constructed of 189,522 unique windows, or, in the language of ML, observations. MATLAB is used to train and test the models [20]. The data set is fed to different ML algorithms, including decision tree, K-Nearest Neighbor (KNN), support vector machine (SVM), and quadratic discriminant analysis, for the purpose of comparing their accuracy and prediction speed. For all classifiers, 25% holdout validation is used for testing the models. The prediction speeds are measured on the same computer machine using MATLAB.

III. Results and Discussion

Table I summarizes the performance of various ML algorithms which are applied to classify the transmission state (State Classifier). We present the prediction speed after training and the classification accuracy of different ML algorithms. As can be seen, for the State Classifier, logistic regression is the fastest algorithm in prediction but is less accurate than the other six tested algorithms. Among the remaining six algorithms, the decision tree is the most accurate model with 99.99% accuracy and also the second fastest predictor. The KNN algorithm has a high accuracy with an average 99.7%,

TABLE I
COMPARING DIFFERENT ML MODELS IN TERMS OF ACCURACY IN PERCENTAGE (%) AND PREDICTION SPEED (PS), MEASURED IN OBSERVATIONS PER SECOND.

	Accuracy	Prediction Speed (obs/sec)
Dicsion Tree	99.99%	890,000
KNN (K=1)	99.98%	250,000
Quadratic Discriminant	67.30%	640,000
Logistic Regression	90.10%	1,500,000
Cosine KNN	98.90%	380
Cubic KNN	99.98%	20,000
Weighted KNN	99.98%	78,000
Linear SVM	71.80%	2,700
Quadratic SVM	35.20%	46,000
Qubic SVM	30.60%	1,300,000
Gaussian SVM	99.80%	13,000

higher than the different flavors of KNN (1-NN, cosine, cubic, weighted). Of those, KNN with k=1 (1-NN) and weighted KNN have the highest prediction accuracy out of the varieties of KNN and 1-NN is the fastest in prediction.

Looking simultaneously for both high accuracy and prediction speed, 1-NN and decision tree are the top candidates for the State Classifier. Nevertheless, the decision tree is $3.5\times$ faster than 1-NN in prediction. Also, regarding the hardware implementation of the models, the 1-NN model needs more storage than that of the decision tree, because 1-NN keeps a copy of the training data in order to calculate the Euclidean distance between the prediction point and the nearest training set observation. The point is then classified according to the class of the closest observation. In contrast, the implementation of the decision tree algorithms is based on branching, with a maximum number of branches per feature of 100 in this case. Therefore, the computational load of the decision tree is much less than that of 1-NN.

As the classifier is applied at the last point of the BT physical layer, that is, the RF output, the proposed design of the supervisory circuit can detect many security breaches or abnormalities in the data transfer behavior. For example, the BB attack takes control of BT enabled devices without any authorization from the user. The State Classifier can detect the connection to the attacking device, and report it to the targeted device. Thus, the targeted device can discover that there is a connection at the physical layer without the authentication of the software layers. Then, the targeted device can shut down the BT chip through the controlled LDO. This initial proof-of-concept demonstrates high classification accuracy for BT states. We note, however, that the current classifier is limited in the number of profiles.

IV. CONCLUSION

In this paper, we demonstrate that we can monitor and verify the correct BT chip operation, thus preventing unauthorized connections and/or transmission of data. We use low-frequency measurements of the RF output signal from a BT SoC.

The state classifier enables the devices that use BT to keep track of what happens at the physical layer. Consequently, the supervisory circuit can monitor the activity of the BT chip by comparing the classifier's output with the software layer commands to the BT chip. We select three simple features from the chip's RF output envelop. These features are enough to achieve a very high classification accuracy ($\sim 100\%$).

ACKNOWLEDGMENT

This work is supported by an LDRD at SNL and conducted at NMSU under contract #GR0005652. SNL is managed/operated by a subsidiary of Honeywell Inc., for the NNSA under contract DE-NA0003525. Any views or opinions expressed in this paper do not necessarily represent the views of the US DOE or the US Government.

REFERENCES

[1] Apple Inc., *iOS Security, iOS 11*, Jan 2018. [Online]. Available: https://www.apple.com/business/docs/iOS_Security_Guide.pdf

[2] Armis Inc., *The Attack Vector BlueBorne Exposes Almost Every Connected Device*, 2019. [Online]. Available: https://www.armis.com/blueborne/

[3] Power Fingerprinting Inc., *Power Fingerprinting (PFP) cybersecurity*, 2019. [Online]. Available: https://www.pfpcyber.com/

[4] Tyco Electronics, *Coordinated Circuit Protection Schemes Help Prevent Overvoltage and Overcurrent Damage*, 2005. [Online]. Available: http://www.te.com/documentation/whitepapers/pdf/eDigest-Circuit_Protection_Devices.pdf

[5] K. Townsend, *GAP — Introduction to Bluetooth Low Energy*, Mar 2014. [Online]. Available: https://learn.adafruit.com/introduction-to-bluetooth-low-energy/gap

[6] Microchip Technology Inc., *Bluetooth Low Energy GAP Modes and Procedures*, 2019. [Online]. Available: http://microchipdeveloper.com/wireless:ble-gap-modes-procedures

[7] P. M. Furth *et al.*, "Supervisory circuits for low-frequency monitoring of communication soc," *In Review at MWSCAS*, 2019.

[8] K. Hasegawa, M. Yanagisawa, and N. Togawa, "Hardware trojan: Threats and emerging solutions," in *HLDVT Workshop*, Nov 2009.

[9] S. Mal-Sarkar *et al.*, "Design and validation for fpga trust under hardware trojan attacks," *IEEE Trans MSCS*, Jul 2016.

[10] C. Bao, D. Forte, and A. Srivastava, "Temperature tracking: Toward robust run-time detection of hardware trojans," *IEEE Trans CADICS*, Oct 2015.

[11] S. R. Hasan *et al.*, "Translating circuit behavior manifestations of hardware trojans using model checkers into run-time trojan detection monitors," in *IEEE AsianHOST*, Dec 2016.

[12] T. Iwase *et al.*, "Detection technique for hardware trojans using machine learning in frequency domain," in *IEEE GCCE*, Oct 2015.

[13] G.-S. Hu, J. Xie, and F.-F. Zhu, "Classification of power quality disturbances using wavelet and fuzzy support vector machines," in *ICMLC*, Aug 2005.

[14] P. Geng *et al.*, "Fault pattern recognition method for the high voltage circuit breaker based on the incremental learning algorithms for svm," in *IEEE DEIS CMD*, Sep 2016.

[15] A. Emrani and M. Pourhomayoun, "Applying machine learning techniques to recognize arc in vehicle 48 electrical systems," in *IEEE COMPEL*, Jul 2017.

[16] R. Shende and D. D. Ambawade, "A side channel based power analysis technique for hardware trojan detection using statistical learning approach," in *IEEE WOCN*, Jul 2016.

[17] J. G. Ferreira and A. Warzecha, "An application of machine learning approach to fault detection of a synchronous machine," in *IEEE SME*, Jun 2017.

[18] Cypress Semiconductor Corp., *CYW920706WCDEVAL Hardware User Guide*, 2017. [Online]. Available: http://www.cypress.com/file/378381/download

[19] Texas Instruments Inc., *AN-2264 LMH2121 Evaluation Board*, May 2013. [Online]. Available: http://www.ti.com/lit/an/snoa873b/snoa873b.pdf

[20] The MathWorks Inc., *Classification Learner*, 2019. [Online]. Available: https://www.mathworks.com/help/stats/classificationlearner-app.html