Protecting Bluetooth User Privacy Through Obfuscation of Carrier Frequency Offset

Ali Nikoofard, *Member, IEEE*, Hadi Givehchian, *Graduate Student Member, IEEE*,
Nishant Bhaskar, *Graduate Student Member, IEEE*, Aaron Schulman, *Member, IEEE*,
Dinesh Bharadia, *Member, IEEE*, and Patrick P. Mercier, *Senior Member, IEEE*

***Abstract*—This brief presents the analysis, design, and measurement results of an integrated circuit designed to prevent tracking the location of Bluetooth Low Energy (BLE) transmitters. Conventional BLE transmitters have unique RF fingerprints due to variation-induced imperfections in the underlying circuits. Coupled with BLE's wide adoption in mobile devices and tendency to transmit continuously, BLE has become a significant threat to the location privacy of individual users. The primary source of this privacy threat is that BLE transmitters have a unique Carrier Frequency Offset (CFO) that can be easily fingerprinted by passive adversaries. To combat this, a test chip is developed that pseudo-randomly changes its CFO by switching in a binary-weighted set of semi-identical MIM capacitors into the tank of an *LC* voltage controlled oscillator, all while maintaining compatibility with BLE standard specifications. Measurement results reveal that privacy preservation can be improved from only a few seconds with a conventional design, to over a day with the proposed design.**

***Index Terms*—Privacy, BLE, WiFi, carrier frequency offset, Tx, machine learning, user identification, hardware security.**

I. Introduction

Bluetooth low energy (BLE) radios are ubiquitous in consumer electronic devices. To facilitate easy synchronization and coordination between devices, BLE radios transmit advertisement packets frequently - up to 10s to 100s of times every minute. Unfortunately, it is possible for an attacker to track a user's location by placing BLE receivers in locations the target is likely to visit, and identify the target's presence, movement, and activities simply by observing unique signatures of these packets. To combat this at the digital layer, cryptographic techniques can be used, for example by periodically re-encrypting their MAC addresses. However, this does not offer a holistic solution; there are still identifiable signatures unintentionally embedded in the physical layer radio circuitry that an adversary can fingerprint.

Specifically, recent work in [1] identified that the carrier frequency offset (CFO), and to a lesser extent I/Q offset and imbalance of commercial BLE transmitters, which are almost exclusively built using I/Q architectures for integration into BLE/WiFi combo chips for cost saving and system integration purposes, have subtle process variations between manufactured units that enable physical-layer fingerprinting by an adversary. Fig. 1(a) shows measured CFO from a collection of 20 different BLE chipsets from the same manufacturer and model, showing clear, distinguishable features between all 20 units.

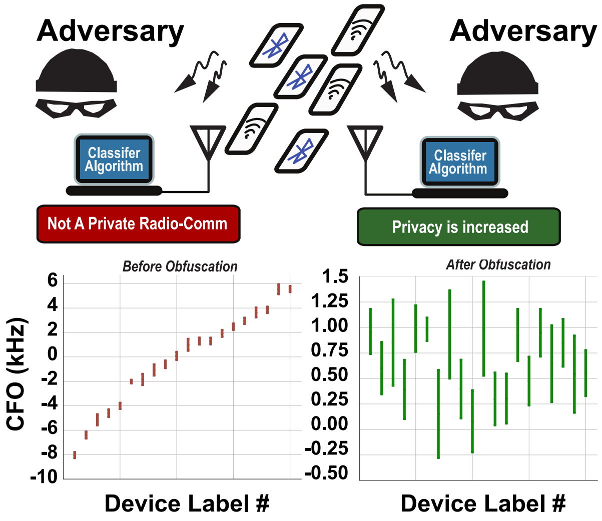


Fig. 1. (a) $[\mu-\sigma, \mu+\sigma]$ of CFO data measured across hundreds of packets from 20 separate ESP32 BLE chipsets, showing unique and identifiable CFOs. (b) Theoretical data for CFO obfuscation, illustrating that unique identification is much more difficult.

It's possible that an adversary can record the CFO of these devices, and later uniquely identify each device, even if their MAC address has been changed, simply by measuring the CFO of the received packets and matching the measured CFO with the recorded ones. It was estimated that an adversary can learn and identify such devices in Fig. 1(a) with 97% accuracy after receiving packets from the device for only about a minute [1]. In fact, it takes only about a minute for the attacker to have a very robust and distinguishable estimate of the CFO of a device, and identify the device very accurately.

If the CFO could somehow be obfuscated at the same time as a MAC address change, as illustrated in Fig. 1(b), then we will show that the identification to 97% accuracy would instead require more than a day (See Fig. 8). Despite this clear privacy issue, no prior-art integrated circuits (ICs) have been designed to help reduce this risk.

This brief first studies methods of enhancing the privacy in transmitters that use I/Q-path modulators popular in

Manuscript received 24 September 2022; accepted 17 October 2022. Date of publication 21 October 2022; date of current version 9 February 2023. This brief was recommended by Associate Editor K. Waheed. *(Corresponding author: Ali Nikoofard.)*

The authors are with the University of California San Diego, La Jolla, CA 92093 USA (e-mail: anikoofa@ucsd.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSII.2022.3216281.

Digital Object Identifier 10.1109/TCSII.2022.3216281

BLE/Wi-Fi combo chips. Since CFO has been widely used to fingerprint wireless transmitters [1], [2], [3], [4], [5], [6], [7], [8], [9], [10], it is selected as the feature to obfuscate. A test chip is then designed to pseudo-randomly obfuscate CFO, where measurement data shows that the proposed idea delays user classification/identification by adversaries from a few seconds to more than a day.

II. Tx Obfuscation Techniques

A. Obfuscation Method

How to Obfuscate: To prevent attackers from identifying the transmitter device by measuring hardware imperfections such as CFO, devices need to obfuscate these imperfections when sending BLE signals. For each hardware imperfection feature h_i (e.g., CFO), there is a set of values $H_i^r = \{h_i^1, h_i^2, h_i^3, \ldots, h_i^R\}$ from which the circuit selects a value ($h_i^r \in \{h_i^1, h_i^2, h_i^3, \ldots, h_i^n\}$) at random and adds it to the original imperfection. As a result, instead of the original unique hardware imperfection h_i, the attackers can only observe the obfuscated imperfection $h_i' = h_i + h_i^r$ when they measure the hardware imperfections embedded in the BLE signals they sniff. In this brief, only CFO obfuscation is considered, as it has been recognized as the most important hardware imperfection to distinguish different transmitters [1], [2] (thus, the index i is dropped hereafter).

When to Obfuscate: As suggested by the BLE protocol standards, MAC address randomization happens once every 15 minutes in most devices [11]. Consequently, the attacker knows all the packets with the same MAC address are coming from the same device during a 15 minute period. If obfuscation randomizes the physical layer identity of packets during that period, the attacker can estimate the original imperfection by averaging multiple packets during a MAC address lifetime. As a result, significant changes in CFO to obfuscate the device's identity should occur in synchronization with MAC address randomization; namely, after a MAC address lifetime expires.

B. Obfuscation Makes the Devices Less Identifiable

Although the obfuscated imperfections randomly change once every MAC address lifetime, there are a limited number of values that a radio circuit can randomly add to the original imperfections. Therefore, if the attackers knows the obfuscation method and the set H^r from which the circuit randomly selects a value and adds to the original hardware imperfection, they can map the measured obfuscated hardware imperfections to possible original hardware imperfection values. Consequently, the attackers can still get some information about the identity of the device using the measured hardware imperfections, however, with a much lower accuracy (See Fig. 1(b)).

Consider there exists N number of users transmitting BLE beacons. The attacker sniffs the BLE packets and measures the CFO for each received packet. The measured CFO values for several packets from a device i ($i = 1, 2, 3, \ldots, N$) have an average $h^{(i)}$ which represents the original CFO of the device, and standard deviation $\sigma^{(i)}$ which represents the packet-to-packet variations of the measured CFO for device i (for instance, because of SNR, natural variations of CFO caused by oscillator, and limited resolution of CFO estimation algorithm). The larger the $\sigma^{(i)}$, the larger the chance of a packet being mis-identified by the attacker. This is simply because there will be a higher chance that the measured CFO of the packet is far from the average CFO of the device; thus, being identified as another device.

Intuitively, if the set H^r has a wider range of values, the obfuscation would be more effective, since even the devices with very different original CFO values may be confused after obfuscation. However, the range of values in the set H^r ($h^R - h^1$) is restricted by the BLE standards that does not allow arbitrarily large CFO values (receivers only need to tolerate CFO that is $\pm 150kHz$). Moreover, the step size (or LSB) with which the values in the set H^r increase ($s = h^{r+1} - h^r$), also plays an important role in the effectiveness of obfuscation since smaller LSB values cause more confusion between devices when the attacker maps the obfuscated CFO to the set of possible original CFO values. In fact, if the CFO measurement noise $\sigma^{(i)}$ dominates the LSB S, the attackers would confuse different devices when they map the measured obfuscated CFO to the set of possible original CFO values.

To measure the effectiveness of our obfuscation method and the attacker's ability to identify the devices after obfuscation, an experiment using 20 commercial ESP32 chipsets is conducted. The chipsets are set up to transmit 4 BLE packets per second for reception with a Universal Software Radio Peripheral (USRP). First, in order to compute the probability that the attacker can successfully identify these 20 chipsets, fingerprinting and identifying the packets coming from these chips is performed by measuring the CFO of each packet, without considering any obfuscation (see the blue curve in Fig. 2). In a longer time duration, as the attacker gets more packets, the probability that they can successfully identify the 20 chips (or the identification accuracy) increases since the attacker can use more packets with the same MAC address to collectively decide about the transmitter's identity. For instance, the attacker can simply average the measured CFO for several packets coming from the same device to get a more accurate estimate of the original CFO of the device (in other words, reduce $\sigma^{(i)}$). Once the attacker has a solid fingerprint for each device, even if the MAC addresses change, the attacker can still easily identify each device via their unique CFO fingerprints.

The same experiment is repeated, however now with the assumption that the obfuscation method is deployed in the ESP32 chipsets. Since the ESP32 chipset is not capable of actually doing this, we instead transmit and receive the packets, and then add in the random CFO value h^r based on the set of CFO obfuscation values that the proposed custom chip design might be able to generate, when post-processing the data. Note that while the MAC address is fixed, the attacker can identify the devices based on their MAC address. Therefore, we only consider a scenario in which the MAC address (and hence, CFO) has been randomized at least once, when the attacker attempts to identify the devices compared to when they fingerprinted them. Also, since in this experiment the time duration that the attacker tries to identify the devices lasts less than 15 minutes (See Fig. 2), MAC addresses and CFOs for each chipset remain constant during the identification itself.

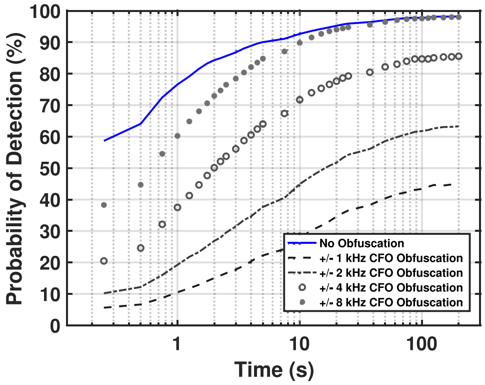


Fig. 2. The effect of CFO obfuscation method on attacker's ability to identify the BLE devices for different LSB values (1, 2, 4, 8 kHz) when 32 CFO states are available (5 bits).

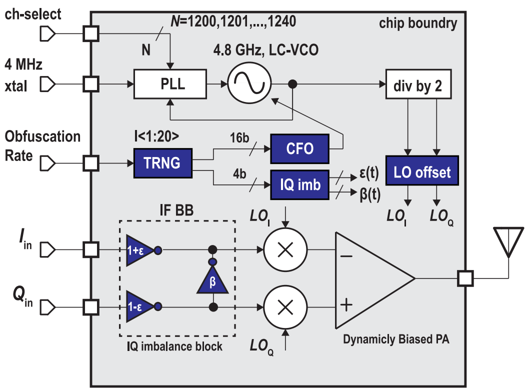


Fig. 3. Privacy enabled WiFi/BLE Tx with integrated TRNG and VCO obfuscation circuit.

To compute the probability of identification after obfuscation, the strongest possible attacker is considered: an attacker that exactly knows the obfuscation strategy. This means that the attacker knows that a random selection of a value from the set H^r is added. Thus, the attacker can map the measured obfuscated hardware imperfections to possible original CFO values.

Fig. 2 demonstrates the probability of identification when obfuscating CFO with different LSB values (the black curves). As the attacker gets more packets over time, they can reduce the CFO measurement noise $\sigma^{(i)}$, and thus a smaller LSB is needed to ensure the same level of confusion. Furthermore, for a specific time duration, decreasing the LSB decreases the attacker's ability to identify the device (probability of detection). The reason is that the attacker's measurement noise is fixed for a specific time duration, and hence, the measurement noise would dominate the LSB more as we decrease the LSB and the attacker would get more confused. If CFO obfuscation with a 1kHz LSB is deployed, the attacker would need more than 100x extra time to identify the devices with the same accuracy as without obfuscation in this experimental model.

C. Unlocked PLL Provides Even Better Obfuscation

In addition to the random obfuscation method, CFO can be further obfuscated if the packet-to-packet CFO variation ($\sigma^{(i)}$) is increased. As explained earlier, if $\sigma^{(i)}$ increases, the attacker would need more packets to achieve the same level of detection accuracy. This extra perturbation of CFO is implemented by using a temporarily-unlocked PLL in our circuit. Prior to packet transmission, the PLL is turned on and the VCO is locked. Then, once a packet needs to be sent, the loop is opened to add the random CFO obfuscation (random number from the set H^r) to the signal using the capacitor bank in the circuit. The frequency drifts as the VCO is operating open loop, causing an unpredictable additional CFO in the transmitted signal (in addition to the desired pseudo-random obfuscation which is determined by the capacitor bank in the circuit). This frequency drift makes it even harder for the attacker to fingerprint and identify the device as the CFO from packet to packet has an unpredictable drift. In fact, the standard deviation of the measured CFO from packet to packet for the same device ($\sigma^{(i)}$) drastically increases as the loop is opened. This can be observed by comparing the measured CFO standard deviation of commercial BLE chipsets in Fig. 1 (a few hundreds of Hz) and the measured CFO standard deviation of one instance of our chipset demonstrated in Fig. 6 (11 kHz). As a result, it becomes harder for the attackers to build a robust fingerprint and also identify the device when a new packet is seen by them.

D. Effect of CFO Obfuscation on Demodulation and BER

The BLE standard allows for a CFO of up to ±150 kHz. Anything below this will not adversely affect the ability to demodulate. As a result, our design is implemented to ensure that CFO never exceeds this range and, as a result, our design modifications would be acceptable by BLE standards. The reason why CFO is acceptable is that BLE decoders usually have a coarse CFO compensation step before demodulation [12] and, any small CFO remaining after coarse compensation does not affect the ability to demodulate GFSK signals. It should also be noted that the unlocked PLL approach has been shown in prior work such as in [13], where PVT was within the acceptable range of the standard.

III. Circuit Implementation

The block diagram of the proposed privacy-preserving BLE compatible transmitter is shown in Fig. 3.

To match the general architecture of the ubiquitous BLE/WiFi combo chips used in many consumer devices, the baseband of the design is implemented by using I and Q branches to meet the required M-QAM modulation employed in WiFi, even though BLE only uses GFSK modulation and could thus use a different architecture. An on-chip integer-N frequency synthesizer is utilized to tune a voltage controlled oscillator (VCO) to a center frequency ranging from 4800 to 4960MHz in 4MHz steps, due to a 4MHz crystal reference. After a divide-by-two circuit, in-phase and quadrature local oscillator (LO) signals are generated between 2400 to 2480MHz in accordance with BLE channel specifications.

After unlocking the PLL, an additional, explicit CFO can be added during a packet transmission. The exact amount of CFO applied depends on the output of an on-chip True Random Number Generator (TRNG), which controls a 16b capacitive DAC connected to the VCO's LC tank as shown in Fig. 4. The output of the mixers connect to an on-chip power

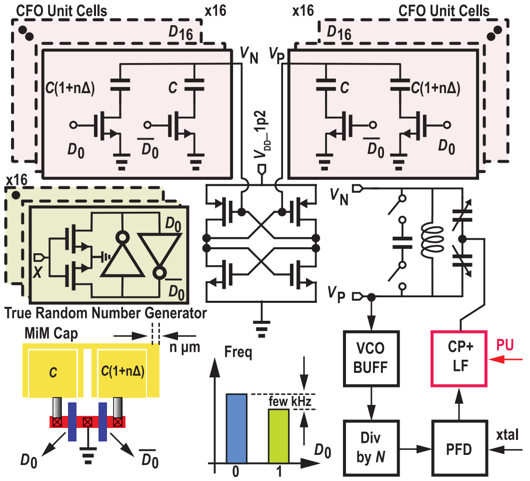


Fig. 4. CFO implementation by using semi-identical MIM caps.

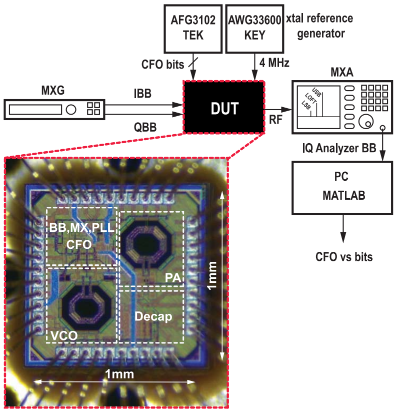


Fig. 5. Measurement setup and die photograph of the private BLE Tx.

amplifier and then an antenna. Fig. 3 also demonstrates two more circuit approaches that may be used in future work for enhanced security. The input I/Q data is fed through an I/Q imbalance block that weakly modulates the amplitude of the two paths and couples them together. Variables $|\epsilon| \ll 1$ and $\beta \ll 1$ can be randomized in a similar fashion to the proposed CFO approach. The chip can also generate LO I/Q offset that moves the center of the demodulated GFSK signal in the I/Q plane (i.e., the center of the demodulated GFSK circle would be randomized). The complete study of I/Q phase mismatch in transceivers is presented in [14] and the aforementioned hardware will be the topic of the future research.

As demonstrated in Fig. 2, smaller CFO LSB values are more desirable as they increase the effectiveness of the obfuscation. Hence, the design target was set as low as 1kHz. Further, the BLE standard mandates that no more than ±150kHz of CFO can be tolerated for decoding purposes, which sets an upper bound for the proposed CFO obfuscation circuit. Here, we set the VCO frequency to be randomly offset around the nominal channel frequency by +/− 80kHz, which is under the bound set by the standard. We did not increase this range further in order to leave a margin for the original CFO of the chipset (that is, the natural CFO of the oscillator without obfuscation), and also the CFO standard deviation added due to the open-loop design.

In this brief, a CFO of LSB of ~1kHz is achieved by toggling between two semi-identical custom MIM capacitors, as shown in Fig. 4. To generate a very small CFO (1kHz at advertisement channel 37 at 2402 MHz is roughly 415 ppb), semi-identical cap switching has been implemented. Adding a stripe of metal-8 to the MIM cap and doing post layout extraction, it has been optimized to achieve kHz range CFO [15], [16], [17]. A 16b Binary CDAC array is implemented alongside the 4.8 GHz LC VCO. Turning on and off every bit would result in an effective +/− ΔC seen by the tank. The ΔC should be in the aF region to result in a kHz frequency offset which would require proper layout for the capacitors and their routing. The number of bits for the CDAC is chosen in such a way to overcome the chip-to-chip mismatch.

Recent works have proposed different methods to implement a TRNG [18], [19], [20]. In this brief, an on-chip TRNG is implemented by using a back to back inverter memory cell that outputs a random bit each time the drain is pulled down using the CFO control pulse (refer to green block in Fig. 4). Nominally, this should occur every 15 minutes to correspond with a MAC address change at the network level. We hypothesize that the TRNG does not necessarily have to be perfect, as it only has to produce a new result every 15 minutes. It takes many samples of a TRNG to be able to gauge its entropy, and over a 28 hour period, only 112 random numbers will have been generated. Thus, our hypothesis is that even a poor random number generator will do just fine in such a situation. This, however, will be the subject of future investigation.

IV. Measurement Results

The fabricated privacy-preserving BLE transmitter alongside the employed measurement setup are shown in Fig. 5. The chip occupies 1mm^2 of total area and is implemented in a 65nm LPCMOS process.

Here, CFO was measured by locking the VCO via the PLL, unlocking the PLL by turning off the charge pump, and observing the mean of the VCO's frequency over the course of a single 400 μsec packet. These results are presented for three different chip samples from the same wafer/lot (additional samples were not available) in Fig. 6(a). Without obfuscation (red data points), the mean CFO for each chip was distinct, and had a tight distribution with a standard deviation of 237Hz when measured across 5000 packets. When obfuscation is enabled, additional bits are pseudo-randomly added to the VCO via the on-chip TRNG after the PLL turns off, and results in Fig. 6(a) reveal a much wider distribution of CFO's: standard deviations of 11, 165, and 70kHz for each chip, respectively. Fig. 6(b) shows the measured CFO data histogram of a chipset that demonstrates the average offset of 14kHz with standard deviation of 11kHz.

Fig. 7 presents the measured phase noise of the transmitter at advertising channel-37, demonstrating a −110 dBc/Hz spot noise at 1MHz offset frequency.

Finally, to demonstrate the effect of the open-loop VCO on obfuscation, the CFO generated by 3 instances of the chip is measured for thousands of packets, and the packet-to-packet CFO variation of ESP32 chipsets with this data is estimated. Fig. 8 shows how much time (or equivalently the number of

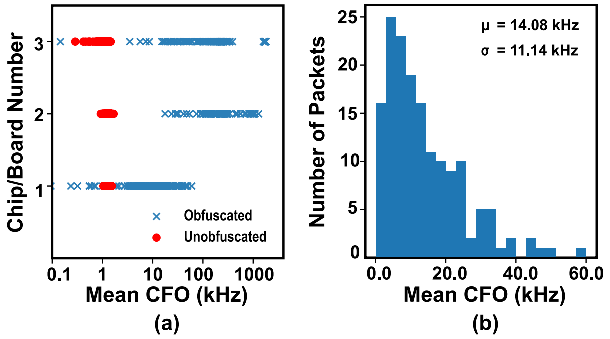
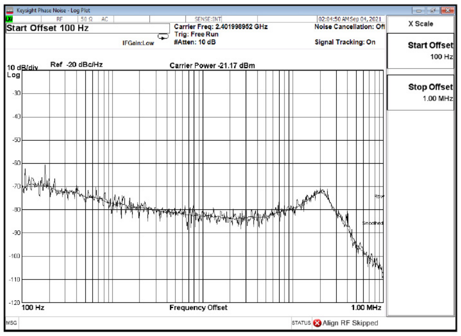


Fig. 6. Measured TRNG enabled, (a) mean frequency CFO and (b) the histogram while transmitting BLE packets (for chip/board #1).


Keysight Phase Noise - Log Plot

RF 50 Ω AC SENSE:INT 02:04:50 AM Sep 04, 2021

Start Offset 100 Hz

Carrier Freq: 2.401998952 GHz
Trig: Free Run
#Atten: 10 dB

IFGain:Low

Noise Cancellation: Off
Signal Tracking: On

Ref -20 dBc/Hz

Carrier Power -21.17 dBm

MSG STATUS Align RF Skipped

Fig. 7. Measured phase noise of the advertising channel-37 of BLE, with −110 dBc/Hz spot noise at 1 MHz offset and 0.1m rad IPN (100Hz to 1MHz).

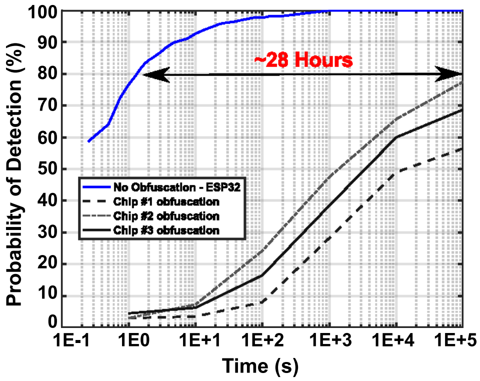


Fig. 8. Obfuscation comparison using the proposed chip (measured chip performances) compared with COTS Tx.

packet) the attacker would need to achieve the same probability of identification when the unlocked PLL architecture with the random CFO obfuscation is used (black curves) compared to when there is no CFO obfuscation (blue curves). Here it can be seen that the length of time needed to identify each chip with 80% confidence increases from less than 1 second to more than 28 hours.

V. Conclusion

The proposed BLE privacy enabled test-chip measured over multiple samples demonstrates the functionality of the proposed idea to enhance the location privacy of BLE transmitter by intentionally varying the CFO. With the proposed methodology, the detection of users can be delayed by multiple orders of magnitude, thanks to frequency uncertainty of an unlocked VCO and the proposed randomization technique. Further improvement to the system can be achieved by intentionally adding I/Q offset and I/Q imbalance randomization.

References

[1] H. Givehchian et al., "Evaluating physical-layer BLE location tracking attacks on mobile devices," in *Proc. Symp. Security Privacy (SP)*, 2022, pp. 1690–1704.

[2] V. Brik, S. Banerjee, M. Gruteser, and S. Oh, "Wireless device identification with radiometric signatures," in *Proc. 14th ACM Int. Conf. Mobile Comput. Netw.*, 2008, pp. 116–127.

[3] T. D. Vo-Huu, T. D. Vo-Huu, and G. Noubir, "Fingerprinting Wi-Fi devices using software defined radios," in *Proc. ACM Conf. Security Privacy Wireless Mobile Netw. (WiSec)*, 2016, pp. 3–14.

[4] J. Hua, H. Sun, Z. Shen, Z. Qian, and S. Zhong, "Accurate and efficient wireless device fingerprinting using channel state information," in *Proc. IEEE INFOCOM*, 2018, pp. 1700–1708.

[5] G. Shen, J. Zhang, A. Marshall, L. Peng, and X. Wang, "Radio frequency fingerprint identification for lora using deep learning," *IEEE J. Sel. Areas Commun.*, vol. 39, no. 8, pp. 2604–2616, Aug. 2021.

[6] Z. Zhu, H. Leung, and X. Huang, "Challenges in reconfigurable radio transceivers and application of nonlinear signal processing for RF impairment mitigation," *IEEE Circuits Syst. Mag.*, vol. 13, no. 1, pp. 44–65, 1st Quart., 2013.

[7] W. Wang, Z. Sun, S. Piao, B. Zhu, and K. Ren, "Wireless physical-layer identification: Modeling and validation," *IEEE Trans. Inf. Forensics Security*, vol. 11, pp. 2091–2106, 2016.

[8] K. Sankhe et al., "No radio left behind: Radio fingerprinting through deep learning of physical-layer hard-ware impairments," *IEEE Trans. Cogn. Commun. Netw.*, vol. 6, no. 1, pp. 165–178, 2020.

[9] A. M. V. V. Sai and Y. Li, "A survey on privacy issues in mobile social networks," *IEEE Access*, vol. 8, pp. 130906–130921, 2020, doi: 10.1109/ACCESS.2020.3009691.

[10] A. Barua, M. A. Al Alamin, M. S. Hossain, and E. Hossain, "Security and privacy threats for Bluetooth low energy in IoT and wearable devices: A comprehensive survey," *IEEE Open J. Commun. Soc.*, vol. 3, pp. 251–281, 2022.

[11] J. K. Becker, D. Li, and D. Starobinski, "Tracking anonymized Bluetooth devices," in *Proc. Privacy Enhanc. Technol.*, 2019, pp. 1–17.

[12] W. Sun, J. Paek, and S. Choi, "CV-Track: Leveraging carrier frequency offset variation for BLE signal detection," in *Proc. 4th ACM Workshop Hot Topics Wireless*, 2017, pp. 1–5.

[13] A. Nikoofard and P. P. Mercier, "A 900MHz GFSK and 16-FSK TX achieving up to 63% TX efficiency and 76% PA efficiency via a DC-DC-powered class-D VCO and a class-E PA," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 69, no. 9, pp. 3739–3743, Sep. 2022.

[14] A. Nikoofard, S. Kananian, and A. Fotowat-Ahmady, "A fully analog calibration technique for phase and gain mismatches in image-reject receivers," *AEU Int. J. Electron. Commun.*, vol. 69, no. 5, pp. 823–835, 2015.

[15] L. Vercesi, L. Fanori, F. De Bernardinis, A. Liscidini, and R. Castello, "A dither-less all digital PLL for cellular transmitters," *IEEE J. Solid-State Circuits*, vol. 47, no. 8, pp. 1908–1920, Aug. 2012.

[16] C. Venerus and I. Galton, "A TDC-free mostly-digital FDC-PLL frequency synthesizer with a 2.8-3.5 GHz DCO," *IEEE J. Solid-State Circuits*, vol. 50, no. 2, pp. 450–463, Feb. 2015.

[17] I. Bashir, R. B. Staszewski, and P. T. Balsara, "A digitally controlled injection-locked oscillator with fine frequency resolution," *IEEE J. Solid-State Circuits*, vol. 51, no. 6, pp. 1347–1360, Jun. 2016.

[18] Y. Luo, W. Wang, S. Best, Y. Wang, and X. Xu, "A high-performance and secure TRNG based on chaotic cellular automata topology," *IEEE Trans. Circuits Syst. I, Reg. Papers*, vol. 67, no. 12, pp. 4970–4983, Dec. 2020.

[19] R. D. Sala, D. Bellizia, and G. Scotti, "A novel ultra-compact FPGA-compatible TRNG architecture exploiting latched ring oscillators," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 69, no. 3, pp. 1672–1676, Mar. 2022.

[20] K. Yang, D. Blaauw, and D. Sylvester, "Hardware designs for security in ultra-low-power IoT systems: An overview and survey," *IEEE Micro*, vol. 37, no. 6, pp. 72–89, Nov./Dec. 2017.