DIRAC: Dynamic-IRregular Ar Clustering Algorithm with Incremental Learning for RF-based Trust Augmentation in IoT Device Authentication

Md Faizul Bari, Baibhab Chatterjee, Shreyas Sen
Department of Electrical & Computer Engineering, Purdue University, West Lafayette, USA
Emails: {mbari, bchattle, shreyas}@purdue.edu

Abstract—Unlike traditional radio frequency device authentication which utilizes security keys in conjunction with a digital subsystem for verification, human voice communication involves probabilistic identification of a person based on his/her voice signatures and improves the detection probability over time. Inspired by voice-based human identification, we implement a novel method of augmenting trust during device detection and authentication, involving dynamic irregular clustering which exploits the unique nonidealities in IoT devices as physical signatures originated from Radio Frequency (RF) circuitry. The proposed method increases the confidence level of the classification as more data come in from a particular device, and is also able to detect new devices that do not fall into any of the previous clusters. Using 30 Xbee modules as transmitters, we show that our proposed method can detect a transmitter with > 95% sensitivity (~ 100% with optimum parameters) using only 0.2 milliseconds of test data which makes it suitable for a very low latency communication system. Also, the incremental learning feature of the proposed method renders a gradual increase in sensitivity as more data are available from the transmitter end. The proposed method can provide an additional security layer in conjunction with the existing methods without adding any additional burden, which is extremely important for resource-limited asymmetric IoT nodes.

Index Terms—Clustering, radio frequency, circuits and systems nonideality, voice-inspired, incremental learning, IoT, security, device authentication

I. INTRODUCTION

A. Motivation and Background

Traditional IoT security involves either a) symmetric-key cryptography that uses the same security key for encryption, b) asymmetric-key cryptography that uses a private and a public key for verification, or c) hash-based message authentication that utilizes one-way hash functions for verifying digital signatures. State-of-the-art security mechanisms involving mutual, multi-factor, and open authentication augment the confidentiality of the system but are still susceptible to cross-site request forgery (CSRF) [1], key-hacking, etc. as none of them uses any inherent device signatures and depend only on processing digital information which may require additional circuitry/data processing both at the transmitter and receiver ends.

Conversely, when humans communicate, we detect the unique voice signature of a person, the confidence on which improves over time as we hear more from that person. Taking inspiration from human voice communication, we present a novel method of augmenting/increasing trust during device identification, in addition to existing digital signature-based methods. This is enabled by the unique device-specific signatures that arise from manufacturing process variations during the fabrication of RF integrated circuits and manifest themselves as system-level nonidealities. These nonidealities from RF circuitry are rejected on the receiver side. However, we embrace them and utilize them as inherent signatures.

Just like humans, any new device in the system will have to introduce itself during the first time communication (device initialization) when an initial voice model of the new device is made. Later, whenever the device communicates, the embedded physical signature is used not only to improve our confidence in detection and identify spoofing attacks but also to improve the voice model. Thus, as we hear more from the device, our model and actual voice signature will match closely and confidence in the identification of the device will increase (note: identification is still happening with commonplace digital signatures, but the confidence that these signatures are not being spoofed is coming from the physical signatures and that confidence is increasing with time). This is shown in Fig. 1, wherein the similarity between human voice communication and the proposed method is presented, and the analogy of augmenting trust with every true assertion is shown. In this scenario, a true assertion is defined as the case when the classification using...
the proposed method matches the actual transmitter (a logical mapping can be done between the transmitter’s physical MAC address and the classification). “Trust” is directly proportional to the probability of correctness in future classifications.

B. Related Work

Radio Frequency (RF) fingerprinting [2]- [4] utilizes time and frequency domain properties of individual transmitters, extracted during power-on, to uniquely identify each device. However, the properties need to be known beforehand, and both time and frequency domain analysis have their limitations in the form of detecting the start and end of the transients, high oversampling ratios, and the need for fixed preambles to avoid data dependency. Recently, I-Q samples (either raw data or slightly processed) from the transmitter are used with complex deep neural network-based frameworks at the receiver side for device classification [5]- [12]. RF wireless data are contaminated with noise and interference that can negatively affect device identification. These methods require a large training data and pre-training before employment. Additionally, complex neural network structure and relatively large test data requirements contribute to significant inference time, making them unusable in low latency communication. Furthermore, in networks where new devices can continuously come in and old devices are thrown out (e.g. mobile networks), traditional clustering mechanisms do not work. Our proposed method addresses all these issues and provides fast, on-the-go device identification. Using data from 30 Xbee S2C modules, it is shown that > 95% sensitivity (> 99% accuracy) can be achieved using only 0.2 ms of test data, which reaches ~ 100% sensitivity with optimum design parameter values.

C. Our Contribution

1) A novel dynamic irregular clustering algorithm, DIRAC, has been proposed using RF circuits and system nonidealities and is verified using data collected from 30 commercial Xbee S2C modules as transmitters. The proposed algorithm forms discrete, well-defined clusters in (mean, standard deviation) space to precisely define the clusters related to original devices. It requires only 2 ms data for one-time device initialization and can provide > 95% sensitivity (reaching ~ 100% sensitivity with optimum parameters) using only 0.2 ms test data, making it suitable for on-the-go authentication in low latency, high-speed communication.

2) As more data are available from the RF transmitters, the proposed clustering method dynamically expands cluster size and incrementally learns the inherent device signature, gradually increasing in trust during classification.

3) The effect of design parameters on RF nonideality-based device detection has been explored, with discussions on sensitivity saturation and optimum threshold limit.

II. DATA COLLECTION AND ANALYSIS

A. Experimental Setup

Fig. 2 shows the physical setup with a block diagram in the inset. Thirty Xbee S2C modules were used as transmitters (TX). A 31-bit pseudo-random bit sequence (PRBS) was generated using MATLAB and was fed to each TX which transmitted this data for 60 s to a receiver module (RX) with QPSK modulation (2.465 GHz carrier). Simultaneously, transmitted data were also captured by a ‘HackRF one’ software-defined radio (SDR), connected via SMA cable to the TX. GNU Radio was used to sample the received data at 6 MSp/s and store them. The captured data are divided into several frames, each containing 1200 samples (0.2 ms data). It is found that some frames are unusable (contain no data, only noise) as Xbee transmits intermittently. These noisy frames (~ 15% of total data) were discarded, and 2-step (coarse and fine) frequency compensation was performed in MATLAB. Subsequently, carrier frequency offset (CFO) was calculated from the filtered frames.

B. Data Analysis and Intuition behind Dynamic Clustering

Our analysis begins with the plotting of the mean (μ) and standard deviation (σ) of CFO from all 30 Xbee modules in a 2D (μ, σ)-space, as shown in Fig. 3(a). (μ, σ) points for each TX form a linear shape instead of a point, which shows that they are evolving over time. Also, some TX form clusters that do not overlap with one another, while some show slight overlapping (just like humans having a similar voice, if not quite the same). Fig. 3(b) shows that traditional clustering methods (e.g. k-means clustering, density-based clustering, etc.) either fail to include all the intended regions in the cluster and/or include extra regions that can potentially belong to rogue devices. Discrete and irregular clusters which are tightly defined around the intended domain and that grow dynamically to adjust the cluster with new data is the much better solution for well-defined clusters in this scenario, which is the basis of the proposed DIRAC method.

III. PROPOSED METHOD

A. Initialization and Irregular Cluster formation

Similar to introducing oneself to an unknown audience, when a new TX enters a system, it has to declare its presence. This introduction can be done when the TX is being verified by the RX using key/hash-based authentication. We do not need any key for DIRAC as we do not utilize the message content. Rather the physical signature embedded in the transmitted RF signal is extracted. As shown in Fig. 3(c) (Initialization stage), let’s assume that for transmitter \( T_i \), initially \( n \) data points are captured, labeled as \( D_1 \) to \( D_n \), and block length is \( b(< n) \).
Then $D_1$ to $D_b$ will be block 1, $D_2$ to $D_{b+1}$ will be block 2, and so on (total $n - b + 1$ number of blocks). From each block, $(\mu, \sigma)$ point is calculated to get $(n - b + 1)$ number of $(\mu, \sigma)$ pairs labeled as $C_{p,p=1}^{,=(n-b+1)}$. The very last $(n-b+1)-th$ block remains as the "current block" of transmitter $T_i$. To sum it up, at the end of the initialization step, each device will have several center points $C_p$ and a "current block" containing the last block of $b$ data points.

**B. Dynamic Irregular Clustering**

If any new data point $D_{n+1}$ is now received at the RX end and it claims to be coming from $T_i$, we verify its claim based on two conditions. Firstly, if the Euclidean distance between $C_x$ (corresponding to data point $D_{n+1}$) and any of the existing centers of $T_i$ is less than or equal to a threshold value, that is $||C_x - C_p|| \leq \text{threshold}$, $C_p \subset T_i$. Secondly, if the Euclidean distance between $C_x$ and any of the existing centers of all other transmitters $T_{j, j \neq i}$ is greater than the threshold, that is $||C_x - C_p|| \geq \text{threshold}$, $C_p \subset T_{j, j \neq i}$. If $C_x$ fails to satisfy any or both of the conditions, we reject its claim of pertaining to $T_i$. So in effect, a cluster for $T_i$ is formed by combining all the circles centered at $C_p$ and with a radius of the threshold. That gives us an irregularly shaped cluster matching the pattern of the transmitter data.

To find $C_x$ of data point $D_{n+1}$, we form a “test block” by taking the last $(b-1)$ data points of the “current block” (for the first condition test, it’s the current block of $T_i$). For the second condition test, it’s the current block of all other transmitters $T_{j, j \neq i}$ in the system, tested one after another) and appending $D_{n+1} - th$ datum at the end. Fig. 3(c) (Testing $D_{n+1}$ datum) shows this process for 3 transmitters in the system. Then we calculate $(\mu, \sigma)$ of the “test block” to find out $C_x$. The inclusion of $(b-1)$ previous data points with the new point helps reducing catastrophic forgetting, a phenomenon quite common in online learning. When any new point $C_x$ satisfies both of the above-mentioned conditions, we put our trust in its claim and assume that it is indeed from $T_i$. The first data point is excluded from the current block of $T_i$ and $D_{n+1} - th$ datum is appended at the end to update the current block. Also, $C_x$ is now included in the set of $C_p$ and $p = p + 1$. With this simple technique, the clusters grow dynamically with new and verified data.

**IV. RESULTS**

**A. Sensitivity, Not Accuracy**

In the proposed method, any newly received test data point will claim that it has come from a certain TX in the system. For testing that claim, we define the claimed/labeled TX as the positive class (so, it’s always 1) and all other TX as the negative class (29 for our case). We see that we have a strong class imbalance and in such a scenario sensitivity ($\frac{TP}{TP+FP}$) or specificity ($\frac{TN}{TN+FN}$) parameters are used instead of accuracy ($\frac{TP+TN}{TP+FP+TN+FN}$), where TP, FP, TN, FN means True Positive, False Positive, True Negative, and False Negative count respectively. Here, a large negative class renders large TN and so, large accuracy (> 99%) can always be obtained. To show the performance more accurately, we choose sensitivity (which does not depend on TN) as our performance indicator.

**B. Effect of Initial Data Size**

Initial data size is an important parameter because it determines the initial clustering and any error made here progressively moves forward. Fig. 3(d) shows the importance of initial center points. As the number of initial center points $(n-b+1)$ increases, the initial cluster becomes more accurate.

While increasing $n$ should increase center points, hence accuracy, increasing $b$ will decrease the number of center points. But higher $b$ provides more accurate center point positions in $(\mu, \sigma)$ space. If $n$ is sufficiently large, increasing $b$ slightly will have a minimal effect on the number of initial center points, yet this higher $b$ value will render their positions more accurately. This analysis is supported by the trend shown in Fig. 4(a) for clusters of non-overlapping nature. Fig. 4(b) shows a similar trend for two overlapping clusters. However, sensitivity saturation is observed after a certain value. DIRAC allocates the overlapping portion to both clusters. So, test data for that region fail in the
In all cases, 1 datapoint = 1 frame = 1200 sample = 0.2 ms

(a) Sensitivity plot (threshold=0.2 kHz) shows that large initial data (n) can offset the effect of the block size, b, and render a good result. (b) Sensitivity saturates in overlapping TX because there is always a certain false negative value that can’t be compensated by increasing initial data. (c) 100% initial sensitivity at an optimum threshold. Increasing threshold inflates the clusters and after a certain value, neighboring clusters start to overlap, leading to a performance drop.

Fig. 5. Sensitivity plot (block length=5 and initial data length=50) shows that the proposed method learns dynamically with more data.

C. Effect of threshold

Fig. 4(c) shows that after an optimum threshold point, both accuracy and sensitivity drop sharply. As the threshold value increases, cluster size increases which, at a certain point, starts to overlap with nearby clusters. This causes some false negative counts and a decrease in sensitivity and accuracy.

D. Incremental Learning

Fig. 5 shows that the performance of our model gradually increases as it tests more and more data because clusters grow dynamically in our method. In voice communication the more we hear from a person, the more our trust grows in his/her voice signature. At some point, we can detect his/her voice even in a noisy crowd. Similarly, our model puts more and more trust in the “voice” of a device by listening to it more and more.

V. POSSIBLE ATTACK MODELS

One possible attack model for the DIRAC algorithm is “replay attack”, which is explained in Fig. 6. The adversary needs to regenerate and replay a message that contains the same physical signature in the transmitted RF signal as the victim TX. Let’s assume that the physical signature of the TX, S, goes through transformation $T_{TX}$ at the TX and $T_{RX}$ at the RX and a malicious device is trying to impersonate. The transformations in the attacker are $T_A$, $T_P$, and $T_D$ respectively. The adversary must produce $T_A^{-1}$ and $T_D^{-1}$ with very high precision to completely nullify their effects, which requires nearly infinite resolution ADC/DAC (practically they are 8/16-bit). Also, doing this in real-time requires an extremely powerful processor. The resolution, computation, and bandwidth limitations combined make this attack practically impossible.

VI. CONCLUSION

In this work, we have shown that the physical signatures from RF circuits and systems can be used to augment trust during digital signature-based device authentication. Using data from 30 commercial Xbee devices, a new dynamic irregular clustering algorithm is proposed which requires only 0.2 ms of test data to provide > 95% sensitivity, and reaches ~ 100% with optimum thresholding. The design space of the proposed algorithm has been explored and the effects of initial data size and threshold have been analyzed in detail. Also, one possible attack model and the robustness of the DIRAC algorithm against it have been discussed. Akin to human voice recognition, the proposed method learns more and more about device signatures in an incremental fashion. This method can enable an additional security stage to the existing key-based authentication techniques without putting extra burden on the existing framework and render the overall message authentication system more secure.
REFERENCES


