PowerKey: Generating Secret Keys from Power Line Electromagnetic Interferences*

Fangfang Yang[1], Mohammad A. Islam[2], and Shaolei Ren[1]

[1] University of California, Riverside
[2] University of Texas at Arlington

Abstract. With the increasing adoption of Internet-of-Things devices, autonomously securing device-to-device communications with minimal human efforts has become mandated. While recent studies have leveraged ambient signals (i.e., amplitude of voltage harmonics) in a building's power networks to secure plugged IoT devices, a key limitation is that the exploited signals are consistent only among nearby outlets, thus resulting in a low key matching rate when devices are far from each other. In this paper, we propose PowerKey to generate secret keys for multiple plugged IoT devices in an electrical domain (e.g., a lab or an office suite). Concretely, PowerKey taps into ambient power line electromagnetic interferences (EMI): there exist multiple spatially unique EMI spikes whose frequencies vary randomly but also remain consistent at participating power outlets to which IoT devices are connected. We propose K-mean clustering to locate common EMI spikes offline at participating outlets and then dynamically extract secret keys at runtime. For evaluation, we conduct experiments in two different locations — one research lab and one suite with multiple rooms. We show that with PowerKey, multiple devices can successfully obtain symmetric secret keys in a robust and reasonably fast manner (i.e., 100% successful at a bit generation rate of up to 52.7 bits/sec).

1 Introduction

The fast growing adoption of inter-connected Internet-of-Things (IoT) devices, such as smart thermostats, WiFi access points and smart power sockets, has been dramatically changing the way we interact with our daily work and living environments. Meanwhile, demand for security as well as usability is also soaring. In particular, a crucial concern is how to quickly establish a shared secret key among various co-located IoT devices without users' manual efforts.

Today, authentication and security for many IoT devices are often delegated to mobile-based apps rather than performed on their own in an autonomous manner. This usually needs to be done for each IoT device through a separate mobile app, since IoT devices may not be using a unified interface provided by third-party vendors. Moreover, the current way to establish secure connections

* This work was supported in part by the NSF under grant ECCS-1610471.

is often *one-time* (during the initial setup) and the secret keys typically remain unchanged for a long time, which poses hidden security threats.

In recent years, exploiting ambient contexts to generate dynamically shared or symmetric secret keys has been emerging as a promising solution to device authentication [1–8]. The key idea is that two or more physically co-located devices can sense similar *ambient* signals, which can serve as a proof of device authenticity. For example, the prior literature has extensively exploited radio frequency signals such as WiFi [2,3,5,9], acoustic signals [10–12], body electric/movement signals (for wearable devices) [1,6,13,14], among many others. However, a major limitation of these techniques is that they are mainly suitable for devices that are very close to each other. For example, to leverage ambient WiFi signals (e.g., amplitude and phase) for key generation, two devices must be placed within half a wavelength (i.e., a few centimeters), since otherwise the WiFi signal's attributes can be dramatically different between the devices [3, 5]. While key generation based on wireless channel reciprocity (i.e., two communicating devices will experience similar channel conditions) can apply for a longer distance [15–17], channel reciprocity is limited to two participating devices. Moreover, it contains little entropy in the generated keys if the two devices are relatively stationary (which is the case for indoor plugged-in IoT devices) [8, 17].

More recently, [7,8] have considered securing IoT devices within an authenticated electrical domain (e.g., a residential house, or a company's office suite) and proposed to exploit the amplitudes of voltage harmonics in the power network for symmetric key generation. Nonetheless, as amplitudes of voltage harmonics are subject to wiring topologies and hence consistent only among nearby outlets, the key matching rate can decrease significantly (to below 90%) when the devices are a few meters away from each other. Thus, this cannot continuously secure IoT devices with a high successful rate.

Contributions. We address the limitation of unreliable key generation under the same setting considered in [8], and present PowerKey, which exploits the consistency of electromagnetic interference (EMI) spike frequencies among outlets within an authenticated electrical domain to secure plugged-in IoT devices. Concretely, multiple devices, even in different rooms connected within a shared electrical domain, can see *similar* EMIs generated by switching mode power supplies (SMPS). These power supplies are used by many electronic devices such as computers, printers and TVs, and create prominent frequency spikes in the $40 \sim 150$kHz range because of high-frequency switching operation [18–20]. Importantly, the frequencies of the EMI spikes vary randomly and, if detectable at participating outlets, will be the same at these outlets. Thus, they can be used as a *reliable* common source of randomness for symmetric key generation.

A key challenge is that most EMI spikes are limited to a small area due to very weak strengths and only a few spikes are detectable as common signals at participating outlets for legitimate devices. Thus, we propose K-means clustering as offline pre-processing to locate the frequency windows over which these common EMI spikes exist at participating outlets. At runtime, legitimate devices can extract secret key information from the selected EMI spikes.

To evaluate PowerKey, we conduct experiments in two locations — an office suite with multiple rooms and a research lab. We show that with PowerKey, devices can successfully generate symmetric secret keys in a robust and reasonably fast manner (i.e., 100% successful at a bit generation rate of up to 52.7 bits/sec). Moreover, even considering a strong attacker that knows all the details of PowerKey but collects voltage signals from an outside outlet, we show that the chance of an attacker obtaining the secret key is practically zero.

2 Preliminaries on Power Line EMI

Overview of EMR/EMI. Electromagnetic radiation (EMR) is generated when electromagnetic fields drive the movement of atomic particles, such as an electron. Another associated concept is electromagnetic interference (EMI), which occurs whenever electromagnetic fields are disturbed by an external source through induction, electrostatic coupling, or conduction [21]. EMI can be broadly classified as *radiated* EMI and *conducted* EMI: radiated EMI (typically > 300MHz) propagates in radio frequencies over the air, whereas conducted EMI (< 300MHz) traverses through power lines [22].

Existing research on exploiting EMR/EMI. EMR signals are good indicators of the system power consumption for power attacks [23]. Electronic devices plugged into power outlets also generate noises (i.e., conducted EMI) propagating through power lines [22, 24]. The prior literature has tapped into power line EMI for simple gesture recognition by sensing its EMI-induced electrical potential [25]. Also, conducted EMI strengths can be extracted to infer a television's content [26] and stealthy data exfiltration from computers [27]. Other studies include exploiting power line EMI for detecting appliance on/off activities in a smart home [22, 28], for estimating data center-level power usage information to launch load injection attacks [29], among others. In addition, the consistent deviation in power grid's nominal 50/60Hz frequency has also been leveraged for wide-area (e.g., city-scale) clock synchronization [30, 31]. By contrast, we exploit switching-induced EMI spikes in $40 \sim 150$kHz for a new and important purpose — key generation to secure IoT device communications.[3]

3 Problem and Threat Model

3.1 Problem Statement

Considering the same setting as in [7, 8], multiple IoT devices are plugged into a power network (e.g., smart thermostats and wireless access points) and need to agree on symmetric secret keys for authenticated communications.

[3] Given a power network and a time window, the frequencies of switching-induced EMI spikes are unique (i.e., spatial-temporal uniqueness) and hence can be exploited for purposes other than key generation. For example, *proof of location*: when a computer is stolen and used elsewhere, the frequency statistics/patterns of EMI spikes will differ, which can prompt additional security measures such as passwords.

Trust domain. In [7, 8], the concept of authenticated electrical domain is introduced, which is also referred to as a *trust domain* and can be a small single-tenant commercial building or a tenant in a large commercial building with restricted physical accesses. Fig. 1 illustrates a building's power network with a standard design [32]. Each panel box delivers electricity to multiple nearby rooms/outlets through parallel branch circuits protected by individual circuit breakers. In reality, each panel box often serves a small commercial building, a residential house, or a tenant (i.e., company) in an office complex, which is an authenticated electrical domain [8].

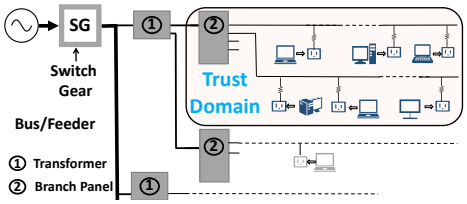


Fig. 1. Overview of a trust domain (i.e., authenticated electrical domain in [8]).

Legitimate devices. A legitimate device can be any plugged-in device, such as smart light bulb and WiFi access point, that is physically located within a trust domain. Thus, the same as in [8], being physically in a trust domain also equals to authenticity. Legitimate devices are synchronized with a granularity of 100ms, which is not restrictive since device-to-device (wireless) communications require even better synchronization [33]. All legitimate devices can sample the voltage signals from the outlets they are plugged in [8].

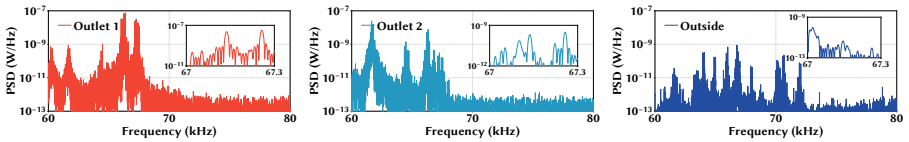
3.2 Threat Model

Following the threat model in [3, 8, 11, 12], attackers cannot forcibly enter the trust domain to acquire the voltage signals or obtain secret keys. The attacker is able to decode all message exchanged between any parties during key generation process. Thus, it knows all the details of PowerKey. The attacker can plug a voltage sensor into a power outlet to directly detect EMI spike frequencies. But, it can only do so outside the trust domain.

4 An In-depth Look at High-Frequency EMI Spikes

All power outlets over a large area beyond a single trust domain share the same fundamental frequency as well as harmonics (i.e., multiples of 50/60Hz) [30]. Thus, the low frequency information does not meet confidentiality requirement for key generation, motivating us to explore high-frequency EMI spikes.

Sources for high-frequency EMI spikes. Many electronic appliances (e.g., computers, televisions, compact fluorescent lights) employ switching-mode power supplies (SMPS), a crucial part of which is the high-frequency switching circuit. Moreover, a power factor correction (PFC) circuit is mandated by international regulations to improve power quality for devices with a rating of more than 75W, which applies to all desktop computers (including certain laptops) and many other appliances [18]. The core of a PFC circuit also relies on the high-frequency

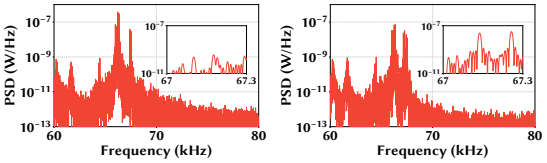


(a) (b) (c)

Fig. 3. PSD of voltage signals. (a) Outlet 1 in the lab. (b) Outlet 2 in the lab. (c) Outside the lab (i.e., outside trusted rooms).

switching operation (typically between 40 $\sim$ 150kHz) [18]. Consequently, the rapid switching operation in PFC and SMPS produces high-frequency conducted EMI, which has been extensively reported by prior studies [22, 28].

To demonstrate EMI spikes, we show in Fig. 2(a) the power spectral density (PSD) of voltage signals collected from a power outlet in our lab. Then, we turn on an additional desktop computer and show the new PSD in Fig. 2(b), which clearly demonstrates the creation of two new EMI spikes (as well as a few weaker spikes) centered around 67.2kHz.



(a) (b)

Fig. 2. Frequency analysis of voltage signals. (a) Without the additional computer; (b) With the additional computer.

Characteristics of EMI spikes. While the amplitudes of EMI spikes can vary significantly depending on the measurement point [8], their frequencies exhibit the following characteristics: they vary rapidly over time, and some of them can remain consistent among multiple power outlets within a trust domain. We perform fast Fourier transform (FFT) on voltage signals to examine the frequency characteristics (detailed experiment setup in Section 6).

Varying randomly. The switching frequency of each SMPS unit can vary randomly within a certain range, depending on the instantaneous load and random drifting [18]. Fig. 9 in the appendix presents the probability distributions of eight EMI frequencies. Note that, due to frequency orthogonality, power line communication does not interfere with switching-induced EMI spikes [34].

Some EMI spikes are consistent for nearby power outlets. While most EMI spikes have weak strengths, we see in Figs. 3(a) and 3(b) that two different outlets in our lab still have consistent EMI spikes around 67.2kHz. The consistent EMI spikes depend on the locations of the outlets: when the set of outlets changes, the set of common EMI spikes also change.

Undetectable from outside the trust domain. Most EMI spikes are localized to nearby outlets due to, e.g., fading over long wires. Moreover, because of physical isolation in different panel boxes, EMI spikes generated within a trusted domain typically vanish and become undetectable from outside the trusted domain. To see this, we collect voltage signals simultaneously both from

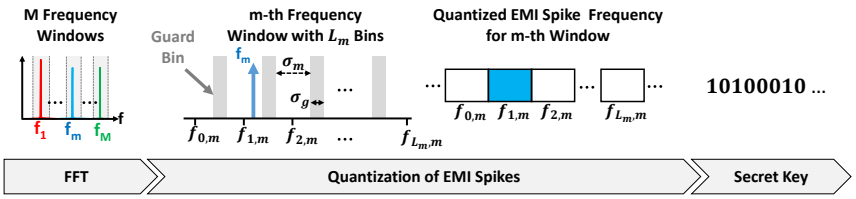


Fig. 4. The design overview of PowerKey.

outlets in our lab and from an outlet in a different electrical domain next to our lab. From Fig. 3(c), we see that the outside outlet has dramatically different frequency patterns than the outlets in our lab. Actually, even for two outlets both in our lab, their voltage signals' frequency patterns shown in Figs. 3(a) and 3(b) are different, despite the similarity over certain frequency bands.

Even though a strong attacker outside the trust domain might detect some leaked EMI spikes from within the trust domain, it is very unlikely that the attacker can detect *all* the common EMI spikes used by legitimate devices for key generation because of the spatial uniqueness of conducted EMI signals [8].

5 The Design of PowerKey

PowerKey is built inside the power supply unit of plugged-in IoT devices. It consists of a high-pass filter (to filter out the dominant 50/60Hz component), an analog-to-digital circuit (ADC), a data communication interface, plus a microcontroller unit. PowerKey is mainly responsible for sending digitized voltage signals to the IoT device, which runs our algorithms. The total hardware cost at scale is below US$5 [8]. Note that sampling voltage signals with 300kHz or higher (to recover signals of up to 150kHz) is not restrictive, as a simple SMPS is already controlled to sample and quantize the voltage signals at a high frequency. We refer to [7] for the detailed implementation. The key difference between PowerKey and VoltKey in [8] is that PowerKey runs FFT, whereas VoltKey leverages the amplitudes of voltages harmonics. Next, we describe PowerKey in detail.

5.1 Offline Pre-Processing

Among numerous (mostly weak) spikes, PowerKey first identifies a set of EMI spikes, whose frequencies vary independently from each other (for more entropy) and are detectable among the participating devices.

• **Step 1.** Each device collects voltage signals for T seconds synchronously as training data and then divides the signal into $N = \frac{T}{\Delta t}$ non-overlapping segments with equal duration Δt.

• **Step 2.** The devices perform FFT analysis on each segment of their own collected voltage signals and pick up EMI spikes over the 40 $\sim$ 150kHz band. For the i-th segment, the devices exchange the frequencies of their own EMI

Algorithm 1 Identify Freq. Windows for Common EMI Spikes

1: Collect voltage signals from devices' outlets for T seconds and divide their own signals into $N = \frac{T}{\Delta t}$ segments each with a duration of Δt seconds.
2: For the i-th segment ($i = 1, 2, \cdots, N$), compare the voltage signals of all devices and find the set of common EMI spike frequencies $\{f_1^i, f_2^i \cdots f_{M_i}^i\}$.
3: Based on the common EMI spike frequencies, run K-means clustering [35] to find $K = \max\{M_1, M_2, \cdots M_N\}$ clusters, each corresponding to one EMI spike.
4: Calculate the correlation coefficient matrix of the EMI spike frequencies. Only one EMI spike is kept if multiple spikes have strongly correlated frequencies.
5: Return M frequency windows $[f_{m,L}, f_{m,R}]$ for $m = 1, 2, \cdots M$

spikes (i.e., local maxima of frequencies) and find the common ones, denoted by the set $\{f_1^i, f_2^i \cdots f_{M_i}^i\}$. Repeat this operation for all the N segments. Here, if the frequencies of an EMI spike at two devices have a difference no more than a threshold η, the two devices are said to have a common EMI spike.

• **Step 3.** Based on N sets of common EMI spikes, we run K-means clustering [35] to find frequency clusters. Then, we perform correlation analysis to remove strongly-correlated EMI spikes and find EMI spikes with little correlation. For each of the remaining M common EMI spikes, we identify its frequency window $[f_{m,L}, f_{m,R}]$, where $f_{m,L}$ and $f_{m,R}$ represent the lower and upper bounds of the m-th EMI spike frequency window. Later, the devices use the detected frequency windows to find EMI spike frequencies at runtime.

The pseudo code is described in Algorithm 1. The K-means algorithm and correlation analysis can be run by a leading device, which then sends back the results to other devices. Re-execution of Algorithm 1 is needed only when the power network environment significantly changes (e.g., some common EMI spikes disappear). Note that the actual EMI frequency, not the range identified offline, is needed to extract keys at runtime.

5.2 Quantize Frequencies of EMI Spikes

At runtime, within a certain frequency window, the common EMI spike can result in slightly different frequencies at different devices due to measurement errors. Thus, we quantize EMI spike frequencies into discrete bins. In this paper, if the frequency difference is no more than σ Hz for 80% of the time, then σ is chosen as the default quantization step size. To further mitigate the frequency discrepancies, we insert a guard frequency band of size σ_g between two valid quantized frequency bins. Fig. 4 provides an illustration of the frequency quantization. For example, a device detects a EMI spike frequency of f within a frequency window $[f_L, f_R]$ and the chosen quantization step size is σ. Then, the frequency is quantized into a bin with index of $\lfloor \frac{f - f_L}{\sigma + \sigma_g} \rfloor$.

5.3 Extract Secret Keys

For key generation, participating devices convert indexes of valid EMI frequencies into binary bits using, e.g., Grey codes. Then, the devices shall exchange

the information to remove invalid EMI spikes whose frequencies fall into guard bins. Finally, they perform reconciliation and privacy amplification.

Converting frequency index into binary bits. If the EMI spike frequency at any participating device falls into an invalid frequency guard band, then it becomes less certain to decide its corresponding frequency bin. Thus, the corresponding EMI spike window is discarded to avoid secret key discrepancies. The devices first find their own invalid windows (if any) and exchange this information with other participating devices. For the remaining valid EMI spike windows, the indexes of their frequency bins will be converted into binary bits.

Reconciliation. For better presentation, we focus on two legitimate devices, i.e., Alice and Bob, while it can also be extended to more than two devices [7,11]. Based on the valid EMI spike frequency windows and indexes, Alice and Bob each end up with a n-bit sequence, denoted by $\widetilde{K}_a$ and $\widetilde{K}_b$, respectively. While it is rare to have different $\widetilde{K}_a$ and $\widetilde{K}_b$, it can still occur in practice.

To improve the key matching rate between Alice and Bob, we apply a crucial step — reconciliation process [4, 6], which uses error correction coding to fix the bit differences/errors at the expense of slowing down bit generation rate. Specifically, the key idea is that both Alice's n-bit sequence $\widetilde{K}_a$ and Bob's n-bit sequence $\widetilde{K}_b$ can actually be viewed as error-corrupted versions of a shared symmetric key, and errors can be fixable using error correction coding. Consider an (n, k, r) error correction code scheme $\mathcal{C}$, which maps any k-bit sequence into a n-bit codewords ($n > k$) through a one-to-one encoding function and can correct up to r error bits. Meanwhile, there exists a many-to-one decoding function that maps any n-bit string into one of the 2^k valid codewords. Let $g_e(\cdot)$ and $g_d(\cdot)$ be the encoding and decoding functions of $\mathcal{C}$, respectively. First, Alice can first decode its n-bit string $\widetilde{K}_a$ and then produces the codeword $g_e(g_d(\widetilde{K}_a))$ that is the closest to $\widetilde{K}_a$. Then, Alice computes the bit-wise error string $\Delta\widetilde{K} = \widetilde{K}_a - g_e(g_d(\widetilde{K}_a))$ and sends it to Bob, which can be in cleartext without encryption. Then, if the bit error rate is roughly estimated and the number of error bits is no more than r, Bob can obtain Alice's n-bit sequence $\widetilde{K}_a$ with a high probability based on $\Delta\widetilde{K} + g_e(g_d(\widetilde{K}_b - \Delta\widetilde{K}))$.

To sum up, if $\widetilde{K}_a$ and $\widetilde{K}_b$ generated from Alice's and Bob's respective quantized EMI spike frequencies differ in no more than r bits, the reconciliation process using the coding scheme $\mathcal{C}$ can ensure that both Alice and Bob eventually possess the same n-bit string.

Privacy amplification. During the reconciliation process, Alice's bit-wise error string $\Delta\widetilde{K} = \widetilde{K}_a - g_e(g_d(\widetilde{K}_a))$, which contains partial information of its n-bit string $\widetilde{K}_a$, is communicated to Bob and meanwhile also possibly leaked to attackers. To address the leakage of partial information about the keys, privacy amplification can be applied: instead of using all the n-bit strings to generate their keys, Alice and Bob can shrink their n-bit strings by $(n-k)$ bits to properly

Table 1. Frequency Quantization Schemes.

Quantization Scheme	**Q1**	**Q2**	**Q3**	**Q4**	**Q5**
Valid Frequency Bin Size (Hz)	σ	σ	σ	σ+1	σ+1
Guard Bin Size (Hz)	0	σ-1	σ	σ-1	σ

create k-bit strings, thus preventing attackers from acquiring partial information about the k-bit strings [4, 6].

6 Evaluation Methodology

Experiment setup. We conduct experiments in two different trust domains — an office suite with multiple individual rooms and a research lab, as shown in the appendix. The office suite is shared by multiple faculty members while the lab has more than 20 workstations. We use the office suite as our default location with multiple faculty offices accessible through a corridor.

Voltage signal collection and processing. For proof of concept, we use a Rigol 1074Z oscilloscope as a proxy ADC to collect voltage signals from the power outlets that are then transferred to a laptop for processing, while one can also follow the design in [7, 8] and insert an additional FFT module.

Error correction coding. We use the following commonly-used error correction coding (ECC) schemes with varying degrees of error tolerance [36]. (i) *Hamming Code*, a linear perfect error correction scheme that encodes every 4 bits of data with 3 parity bits and can withstand 1-bit error in the data. (ii) *Golay Code*, another linear code which encodes 12 bits data into 23 bits and can correct up to 3 error bits. (iii) *Reed-Solomon Code (RS)*, a non-linear cyclic code that can detect and correct multiple errors: an $RS(n, k)$ encoding can correct up to $\lfloor \frac{n-k}{2} \rfloor$ bit errors. In our evaluation, we use three variations of the RS code — $RS(7, 3)$, $RS(15, 5)$, and $RS(15, 3)$.

Frequency quantization and guard bin size. We set σ as the step size if the frequency difference between any two outlets is no greater than σ for 80% of the time. As shown in Table 1, we test five different quantization schemes with varying step sizes and guard bands, denoted as $Q1, Q2 \cdots, Q5$.

Experiment durations. We first collect 500 seconds of voltage data simultaneously from the chosen power outlets to identify the common EMI spike windows offline (Section 5.1), and determine the quantization scheme. We use $\Delta t = 100$ms as the length of each voltage signal segment. For online evaluation, we use the same segment length and run the experiments for 60 minutes.

Evaluation metrics. We consider the following standard metrics.

• **Bit generation rate.** It is the number of secret bits generated per unit time. Consider a segment size of Δt seconds and M common EMI spikes with frequency windows $[f_{m,L}, f_{m,R}]$, quantization step size σ_m and frequency guard band size $\sigma_{g,m}$, for $m = 1, 2, \cdots M$. The bit generation rate (BGR) in bits per second with ECC $\mathcal{C}(n, k, r)$ is given by $\text{BGR} = \frac{k}{n} \cdot \frac{1}{\Delta t} \sum_{m=1}^{M} \log_2 \lfloor \frac{f_{m,R} - f_{m,L}}{\sigma_m + \sigma_{g,m}} \rfloor$.

• **Bit Error Rate.** It indicates the probability of differences between secret keys extracted by two or more devices. A low bit error rate (BER) is desirable.

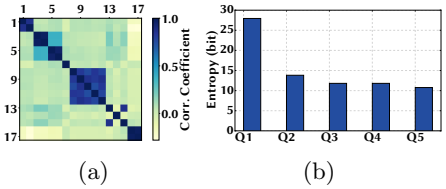
• **Key Matching Rate.** This indicates, on average, the percentage of keys generated by PowerKey can be used as a valid shared secret key. We use the standard AES 128-bit key as the length requirement [37].

In addition, we also consider *Entropy* and *Mutual Information*. Entropy measures the amount of information contained in the random variable we generate from the EMI spike frequencies. Mutual information quantifies the amount of dependency between two random variables and we use this to measure the information possibly obtained by an attacker.

7 Evaluation Results

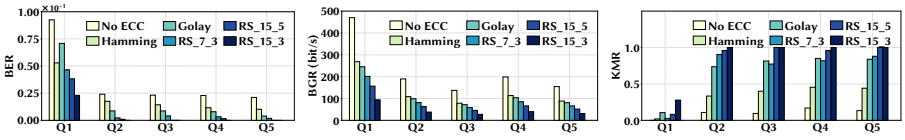
In this section, we present our evaluation results in the office suite, while the results in the lab are deferred to the appendix. Our results demonstrate that with the design of PowerKey, multiple devices can successfully generate symmetric secret keys in a robust and fast manner (i.e., with a 100% key matching rate at a bit generation rate of 52.7 bits/sec).

Analysis of EMI spike frequencies. By pre-processing the voltage signals in the office suite, we identify a total of 17 common EMI spikes out of hundreds of spikes. As shown in Fig. 5(a), only 8 of the 17 spikes are uncorrelated, while the remaining spikes are redundant and need to be removed. We also show the histograms of the 8 independent EMI spike frequencies and the frequency differences at the two outlets in Fig. 9 and Fig. 10 in the appendix, respectively. It can be seen that each of the 8 EMI spike frequencies varies within a narrow window. We also run randomness test on frequencies of the 8 EMI spikes in Matlab using runstest(·). The results are all positive, verifying the randomness of EMI spike frequencies with a 95% significance level [38].



(a) (b)

Fig. 5. (a) Correlation coefficients of EMI spike frequencies in the office. (b) Entropy with different quantizations.



(a) Bit error rate (b) Bit generation rate (c) Key matching rate

Fig. 6. Performance of PowerKey in the office suite.

Performance of PowerKey. We now examine the performance of PowerKey.

Entropy of EMI spike frequencies. Fig. 5(b) shows the impact of our quantization configurations on the overall entropy of the 8 EMI spike frequencies. Naturally, when the EMI spike frequency is mapped to fewer bins, the amount of entropy also decreases but still is better than some of the existing literature whose ambient signals can only have $1 \sim 2$bits [4,6].

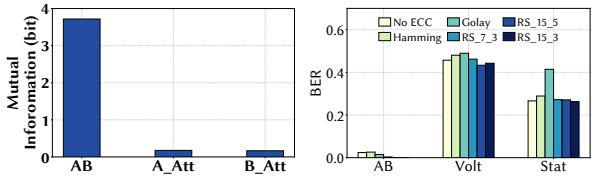
Bit error rate. We now look at the bit error rate under different quantization and ECC schemes and show the results in Fig. 6(a). We see that either quantizations or ECC alone cannot achieve a low bit error rate. By combining quantization with an appropriate ECC scheme (e.g., $RS(15,5)$ or $RS(15,3)$), PowerKey essentially achieves a zero bit error rate in practice.

Bit generation rate. We show the bit generation rate in Fig. 6(b). As in the prior literature [4,6], the bit generation rate only considers how many secret key bits Alice and Bob can generate, without accounting for possible errors. Clearly, both quantization and ECC reduce the bit generation rate, but they are needed to achieve a high key matching rate as we show next.

Key matching rate. Next, we show the key matching rate (KMR) between Alice and Bob in Fig. 6(c) for the standard AES 128-bit key [37]. We see that ECC plays a vital role to correct mismatched bits between Alice and Bob. Specifically, the RS codes perform the best, achieving nearly 100% key matching rate when combined with quantization. By contrast, when using amplitudes of voltage harmonics for key generation for devices 18m (approx. 60ft) away, the key matching rate reduces to below 90% [8].

Security analysis of PowerKey. We consider an attacker that can collect voltage signals from outside the trust domain, be synchronized with Alice/Bob, and knows all the details of PowerKey (including the common EMI spike frequency windows located offline). In our experiment, we choose an outlet next to the entrance to our office suite. We assume that the attacker uses its most prominent EMI spikes, or estimates the EMI spike frequencies based on their probability distribution, within each valid EMI frequency window. Thus, the attacker is assumed to follow the same procedure as a legitimate device, except for that it extracts EMI spike frequencies from outside the trust domain.

We first calculate the mutual information between two parties in Fig. 7(a). We see that the mutual information between the attacker and Alice/Bob is much lower compared to that between Alice and Bob, thus showing that the attacker's signal contains little information about Alice's/Bob's. Next, we show the bit error rate in Fig. 7(b) for quantization scheme Q4 (Table 1) and see that, under various strategies, the attacker's bit error rate is significantly higher than that of Alice/Bob, resulting in almost random bits. Further, it achieves a practically zero key matching rate, and hence we omit the result. The reason that the attacker is not able to acquire the secret key is that the common EMI spikes located offline are spatially unique to the power outlets to which legitimate devices are connected.



(a) (b)

Fig. 7. (a) Mutual information: "AB" (Alice-Bob), "A-Att" (Alice-Attacker), and "B-Att" (Bob-Attacker). (b) Bit error rate. "AB" means Alice/Bob; "Volt" means the attacker uses the highest EMI spike for each window from its collected signals; "Stat" means estimating the EMI spike frequencies based on their probability distributions.

8 Related Works

For key generation, the prior research has exploited radio frequency signals such as WiFi [2, 3, 5, 9], acoustic signals [10–12, 15], body electric/movement signals (for wearable devices) [1,6,13,14], among many others. Nonetheless, the existing approaches can suffer from a limited distance [2,3,5,9], low key matching rate [6], and/or low bit generation rate [4, 5, 9]. While key generation based on wireless channel reciprocity can apply for a longer distance [15–17], channel reciprocity often needs time-division multiplexing and is limited to two participating devices each time. Moreover, it contains little entropy in the generated keys if the two devices are relatively stationary [17]. Other studies [11, 12] look at secret key generation within a *single* room by utilizing ambient acoustic/luminous characteristics, but they require long-term statistics of the ambient signals and hence take several minutes or even longer to produce a valid key.

The recent study [8] considers key generation for plugged-in IoT devices under the same setting as ours, but it leverages amplitudes of voltage harmonics that are consistent only among nearby outlets. Thus, when the inter-device distance increases (e.g., 10m), the key matching rate can significantly decrease.

Finally, our work is also relevant to studies that exploit conducted EMI for side channel inference/attacks [26,27,39]. Nonetheless, PowerKey is novel in that it exploits EMI spike frequencies for an orthogonal and important goal — secret key generation.

9 Conclusion

In this paper, we proposed a novel key generation approach, called PowerKey, based on EMI spikes in an authenticated electrical domain. PowerKey includes an offline pre-processing stage to identify common EMI spikes as well as run-time extraction of EMI spike frequency for key generation. For evaluation, we conducted real experiments in two different locations — one research lab and one suite with multiple offices. Our results demonstrated that PowerKey can successfully generate secret keys in a robust and reasonably fast manner (i.e., with 100% key matching rate at a bit generation rate of up to 52.7 bits/sec).

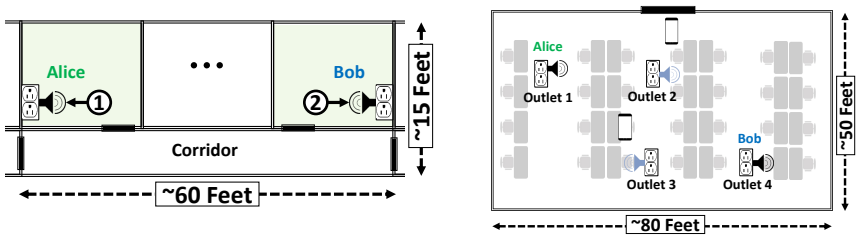
References

1. Z. Yan, Q. Song, R. Tan, Y. Li, and A. W. K. Kong, "Towards touch-to-access device authentication using induced body electric potentials," in *MobiCom*, 2019.
2. S. Jana, S. N. Premnath, M. Clark, S. K. Kasera, N. Patwari, and S. V. Krishnamurthy, "On the effectiveness of secret key extraction from wireless signal strength in real environments," in *MobiCom*, 2009.
3. W. Xi, C. Qian, J. Han, K. Zhao, S. Zhong, X.-Y. Li, and J. Zhao, "Instant and robust authentication and key agreement among mobile devices," in *CCS*, 2016.
4. S. Mathur, R. Miller, A. Varshavsky, W. Trappe, and N. Mandayam, "Proximate: Proximity-based secure pairing using ambient wireless signals," in *MobiSys*, 2011.
5. A. Varshavsky, A. Scannell, A. LaMarca, and E. de Lara, "Amigo: Proximity-based authentication of mobile devices," in *UbiComp*, 2007.
6. L. Yang, W. Wang, and Q. Zhang, "Secret from muscle: Enabling secure pairing with electromyography," in *SenSys*, 2016.
7. J. West, T. VoNguyen, I. Ahlgren, I. Motyashok, G. K. Thiruvathukal, N. Klingensmith, K. Lee, D. He, Y. Kim, and S. Banerjee, "Demo abstract: VoltKey: Continuous secret key generation based on power line noise for zero-involvement pairing and authentication," in *IPSN*, 2019.
8. K. Lee, N. Klingensmith, S. Banerjee, and Y. Kim, "Voltkey: Continuous secret key generation based on power line noise for zero-involvement pairing and authentication," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 3, Sept. 2019.
9. S. Mathur, W. Trappe, N. Mandayam, C. Ye, and A. Reznik, "Radio-telepathy: Extracting a secret key from an unauthenticated wireless channel," in *MobiCom*, 2008.
10. P. Xie, J. Feng, Z. Cao, and J. Wang, "Genewave: Fast authentication and key agreement on commodity mobile devices," *IEEE/ACM Trans. Netw.*, vol. 26, pp. 1688–1700, Aug. 2018.
11. M. Miettinen, N. Asokan, T. D. Nguyen, A.-R. Sadeghi, and M. Sobhani, "Context-based zero-interaction pairing and key evolution for advanced personal devices," in *CCS*, 2014.
12. M. Miettinen, T. D. Nguyen, A.-R. Sadeghi, and N. Asokan, "Revisiting context-based authentication in iot," in *DAC*, 2018.
13. W. Wang, L. Yang, Q. Zhang, and T. Jiang, "Securing on-body iot devices by exploiting creeping wave propagation," *IEEE Journal on Selected Areas in Communications*, vol. 36, pp. 696–703, April 2018.
14. Z. Luo, W. Wang, J. Xiao, Q. Huang, T. jiang, and Q. Zhang, "Authenticating on-body backscatter by exploiting propagation signatures," *Proc. ACM Interact. Mob. Wearable Ubiquitous Technol.*, vol. 2, pp. 123:1–123:22, Sept. 2018.
15. Y. Lu, F. Wu, S. Tang, L. Kong, and G. Chen, "Free: A fast and robust key extraction mechanism via inaudible acoustic signal," in *MobiHoc*, 2019.
16. J. Zhang, T. Q. Duong, A. Marshall, and R. Woods, "Key generation from wireless channels: A review," *IEEE Access*, vol. 4, pp. 614–626, 2016.
17. J. Wan, A. B. Lopez, and M. A. Al Faruque, "Exploiting wireless channel randomness to generate keys for automotive cyber-physical system security," in *ICCPS*, 2016.
18. On Semiconductor, "Power factor correction (PFC) handbook," `http://www.onsemi.com/pub/Collateral/HBD853-D.PDF`.

19. On Semiconductor, "Switch-mode power supply reference manual," https://www.onsemi.com/pub/Collateral/SMPSRM-D.PDF.
20. A. Pressman, *Switching Power Supply Design*. McGraw-Hill, Inc., 2 ed., 1998.
21. Wikipedia, "Electromagnetic interference," https://en.wikipedia.org/wiki/Electromagnetic_interference.
22. M. Gulati, S. S. Ram, and A. Singh, "An in depth study into using emi signatures for appliance identification," in *BuildSys*, 2014.
23. R. Callan, A. Zajić, and M. Prvulovic, "A practical methodology for measuring the side-channel signal available to the attacker for instruction-level events," in *MICRO*, 2014.
24. Electronic Code of U.S. Federal Regulations, "Unintentional radiators, section 15.107 — conducted limits," 2018.
25. G. Cohn, D. Morris, S. N. Patel, and D. S. Tan, "Your noise is my command: Sensing gestures using the body as an antenna," in *CHI*, 2011.
26. M. Enev, S. Gupta, T. Kohno, and S. N. Patel, "Televisions, video privacy, and powerline electromagnetic interference," in *CCS*, 2011.
27. Z. Shao, M. A. Islam, and S. Ren, "Your noise, my signal: Exploiting switching noise for stealthy data exfiltration from desktop computers," *Proc. ACM Meas. Anal. Comput. Syst.*, vol. 4, May 2020.
28. S. Gupta, M. S. Reynolds, and S. N. Patel, "Electrisense: Single-point sensing using emi for electrical event detection and classification in the home," in *UbiComp*, 2010.
29. M. A. Islam and S. Ren, "Ohm's law in data centers: A voltage side channel for timing power attacks," in *CCS*, 2018.
30. S. Viswanathan, R. Tan, and D. K. Y. Yau, "Exploiting electrical grid for accurate and secure clock synchronization," *ACM Trans. Sen. Netw.*, vol. 14, pp. 12:1–12:32, May 2018.
31. Y. Li, R. Tan, and D. K. Y. Yau, "Natural timestamps in powerline electromagnetic radiation," *ACM Trans. Sen. Netw.*, vol. 14, pp. 13:1–13:30, June 2018.
32. Arch Toolbox, "Electrical power systems in buildings," https://www.archtoolbox.com/materials-systems/electrical/electrical-power-systems.html.
33. A. Goldsmith, *Wireless Communications*. Cambridge University Press, 2005.
34. IEEE Standards Association, "IEEE draft standard for broadband over power line networks: Medium access control and physical layer specifications amendment: Enhancement for internet of things applications," 2018, https://standards.ieee.org/project/1901a.html.
35. T. Kanungo, D. M. Mount, N. S. Netanyahu, C. D. Piatko, R. Silverman, and A. Y. Wu, "An efficient k-means clustering algorithm: Analysis and implementation," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, pp. 881–892, July 2002.
36. G. C. Clark and J. B. Cain, *Error-Correction Coding for Digital Communications*. Springer Publishing Company, Incorporated, 1st ed., 2013.
37. Wikipedia, "Advanced encryption standard," https://en.wikipedia.org/wiki/Advanced_Encryption_Standard.
38. MathWorks, "Run test for randomness," https://www.mathworks.com/help/stats/runstest.html.
39. Q. Pu, S. Gupta, S. Gollakota, and S. Patel, "Whole-home gesture recognition using wireless signals," in *MobiCom*, 2013.

Appendix

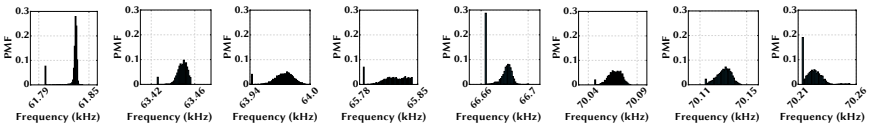
Experiment Setup. We conduct experiments in two different trust domains — an office suite with multiple individual rooms (Fig. 8(a)) and a research lab (Fig. 8(b)).



(a) Office (b) Lab

Fig. 8. (a) Layout of the office. (b) Layout of the lab.

Analysis of EMI spike frequencies in the office suite. We show the histograms of the 8 independent EMI spike frequencies and the frequency differences at two outlets in Fig. 9 and Fig. 10, respectively. We see that the two outlets share certain time-varying EMI spike frequencies with only minor differences.



(a) (b) (c) (d) (e) (f) (g) (h)

Fig. 9. Histogram of 8 different EMI spike frequencies in the office suite.

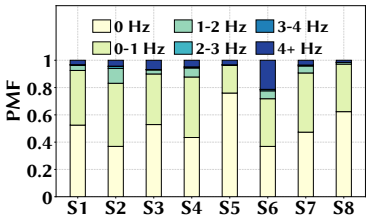
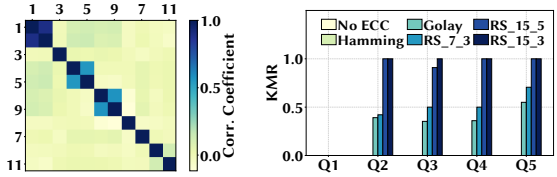


Fig. 10. Distribution of frequency differences between two outlets for 8 different EMI spike frequencies in the office suite. "S-n" means the n-the EMI spike. $\sigma = 1, 1, 1, 1, 1, 4, 1, 1$Hz for the 8 EMI spike windows, respectively.

Results for Key Generation in the Lab We now run experiments in a lab with 20+ desktops shown in Fig. 8(b).

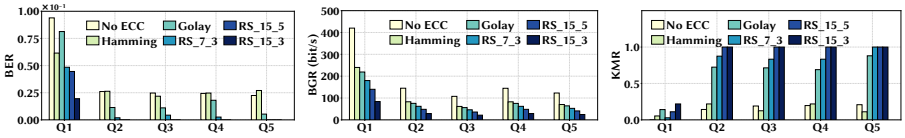
Analysis of EMI spike frequencies. After offline pre-processing, PowerKey identifies a total of 11 EMI spikes for the lab. Then, as shown in correlation analysis in Fig. 11(a), 8 of the 11 spikes are uncorrelated, while the remaining ones are redundant and need to be removed.



(a) (b)

Fig. 11. (a) Correlation coefficients of EMI spike frequencies in the lab. (b) Key matching rate for four devices in the lab.

Key generation performance. We show the key generation performance for the lab. The main results are deferred to Fig. 12. We can see that in terms of all the evaluation metrics, the performance of PowerKey is consistent with that in the office setting. Likewise, the attacker can barely obtain secret keys successfully, with a high bit error rate and practically zero key matching rate.



(a) Bit error rate (b) Bit generation rate (c) Key matching rate

Fig. 12. Performance of PowerKey in the lab.

Multiple devices. Next, we consider four devices associated with four outlets in Fig. 8(b). Our results in Fig. 11(b) show that with an appropriate quantization and ECC scheme, PowerKey can still generate secret keys with a negligible bit error rate and almost 100% key matching rate, demonstrating its reliable key generation.