PowerHammer: Exfiltrating Data From Air-Gapped Computers Through Power Lines

Mordechai Guri, Boris Zadov, Dima Bykhovsky, and Yuval Elovici

***Abstract*— In this article, we provide an implementation, evaluation, and analysis of PowerHammer - an attack that uses power lines to exfiltrate data from air-gapped computers. A malicious code running on a compromised computer intentionally controls the utilization of the CPU cores. The CPU utilization is electromagnetically conducted and propagated through the power lines in the form of a parasitic signal that is modulated, encoded, and transmitted on top of the current flow fluctuations. This electromagnetic phenomenon is known as '*conducted emission*'. In this attack, the attacker taps the indoor electrical power wiring that is connected to the electrical outlet of the compromised computer. The conducted electromagnetic emission of the compromised computer is analyzed and the exfiltrated data is decoded. The proposed attack is then experimentally evaluated and characterized. The communication performance is discussed and a set of defensive countermeasures is presented. A crucial aspect of the proposed covert communication scheme is that it fully conforms to civilian and military conductive emission standards.**

***Index Terms*— Network security, air gap, covert channel, exfiltration, power line communication.**

I. Introduction

INFORMATION is the most critical asset of modern organizations and, hence, is coveted by adversaries. Protecting IT networks from sophisticated cyber-attacks is a challenging task, involving host level and network level security layers. This includes updating protection software in the host computers, configuring firewalls and routers, managing access controls, using centralized credential systems, and more. Nevertheless, despite a high degree of protection, as long as the local area network has a connection with the outside world (e.g., the Internet), an innovative and persistent attacker will eventually find a way to breach the network, eavesdrop, and transmit sensitive data outwards (e.g., see the Vault 7 [1], Sony [2], and Yahoo [3] incidents).

When sensitive data is involved, an organization may resort to so-called 'air-gap' isolation. An air-gapped network is a secured computer network in which measures are taken to maintain both physical and logical separation from less secured networks. The air-gap separation is maintained by enforcing strict regulations, such as prohibiting connectivity to unauthorized equipment and hardening the workstations in the network. Today, air-gapped networks are used in military and defense systems, critical infrastructure, the finance sector, and other industries [4], [5]. Two examples of air-gapped networks are the United States Defense Intelligence Agency's NSANET and Joint Worldwide Intelligence Communications System (JWICS) classified networks [6]. However, even air-gapped networks are not immune to breaches. In the past decade, it has been shown that attackers can successfully penetrate air-gapped networks by using complex attack vectors, such as supply chain attacks, malicious insiders, and social engineering [7]–[9]. In 2017, WikiLeaks published a reference to a hacking tool dubbed 'Brutal Kangaroo' used to infiltrate air-gapped computers via USB drives [10]. This tool was used by attackers to infect the Internet workstations of an organization's employees and wait for an employee to insert the infected USB drive into an air-gapped computer. Using such tools, attackers can breach the network, bypassing security systems such as AVs, firewalls, intrusion detection and prevention systems (IDS/IPS), and others.

After deploying malware in the air-gapped network, the attacker may, at some point, wish to leak information - a behavior commonly seen in advanced persistent threats (APTs). In order to exfiltrate data from air-gapped networks the attacker must resort to special types of covert channels, commonly referred to as 'air-gap covert channels'. In this approach, the malware uses the emanations from different components of the computer to establish out-of-band covert communication with the outer world [11]. Over the years, various methods to leak data through air-gaps have been developed. For example, electromagnetic covert channels have been studied for more than twenty years. In this type of communication, malware modulates binary information over the electromagnetic waves radiating from computer components (e.g., LCD screens, communication cables, computer buses, and hardware peripherals [12]–[21]). Other types of air-gap covert channels based on magnetic [22], [23], acoustic [24]–[30], optical [31]–[38] and thermal [39] emissions have also been investigated.

Different kinds of attack through electrical network connections have long been a subject of considerable interest. For example, Kocher et al. [40] proposed a differential consumed power analysis attack that determines hidden patterns from power traces of cryptographic computations and in [41]

Manuscript received December 25, 2018; revised June 24, 2019 and August 16, 2019; accepted October 21, 2019. Date of publication November 7, 2019; date of current version January 27, 2020. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Eduard A. Jorswieck. *(Corresponding author: Mordechai Guri.)*

M. Guri, B. Zadov, and Y. Elovici are with the Department of Software and Information Systems Engineering, Ben-Gurion University of the Negev, Be'er Sheva 8400711, Israel (e-mail: gurim@post.bgu.ac.il).

D. Bykhovsky is with the Department of Software and Information Systems Engineering, Ben-Gurion University of the Negev, Be'er Sheva 8400711, Israel, and also with the Department of Electrical and Electronics Engineering, Sami Shamoon College of Engineering, Be'er Sheva 8410802, Israel.

Digital Object Identifier 10.1109/TIFS.2019.2952257

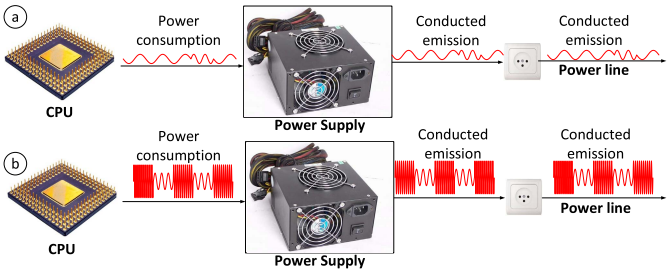

Fig. 1. Signal generation: (a) the conducted emission during usual workload of a computer and (b) the conducted emission during PowerHammer transmissions.

power-load attack was proposed. The particular application of an electrical network for air-gapped communication was also proposed [42]–[44].

In this paper, we present a new type of electric (current flow) covert channel. The method, which we named *PowerHammer*, enables attackers to exfiltrate information from air-gapped networks through AC power lines. We show that dedicated malware running on a computer can change the power consumption of the system by controlling the workload of the CPU. Binary data can be modulated on the changes of the current flow; the resulting emission is conducted to the power lines and propagates through them (Fig. 1). Using a non-invasive tap, the attacker measures the emission conducted on the power cables. Based on the signal received, the transmitted data is demodulated and decoded back into binary form. The main advantage over [42], [43] is the use of conductive emission rather than power consumption, which enables operation with a substantially higher bit-rate. In [44], preliminary results with significantly lower modulation frequencies and slower communication bit-rates, but a similar approach, were presented. An illustration of the proposed attack is presented in Fig. 1.

The rest of this paper is structured as follows. Technical background is provided in Section II. The adversarial attack model, communication and modulation are discussed in Section III. Section IV presents the experimental evaluation results. Communication performance is discussed in Section V. Undetectability and countermeasures are discussed in Section VI, and Section VII concludes.

II. Technical Background

A. Electricity Network

A typical indoor electrical supply network is comprised of a distribution board that divides electrical power into subsidiary phases. Each phase feeds several to dozens of circuits. The distribution to three different phases prevents internal wall wires from overheating and makes it possible to run large loads. Electricity distribution in buildings is managed in a main, centralized electrical service panel (Fig. 2). The power cords are connected to the floor panel with circuit breakers.

B. Switch-Mode Power Supplies & Conducted Emission

Computers consume power by using modern switch-mode power supplies (SMPSs). SMPSs are used in many types

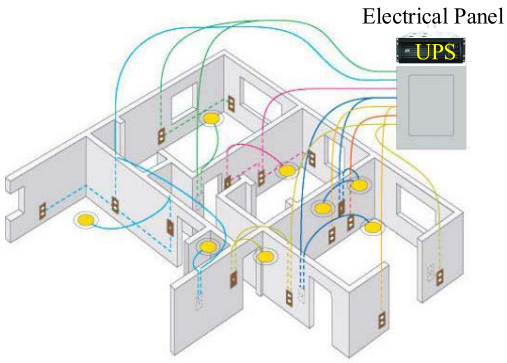

Fig. 2. Indoor power distribution.

of electronic equipment, including computers, TVs, printers, and cell phone chargers. One of the key features related to SMPSs in the context of this paper, is that they emanate electromagnetic radiation due to the fast switching current and voltage they generate during normal operation [45], [46]. The radiated noise is also leaked to the power lines, due to electromagnetic conductive processes known as *conducted emission* [47].

There are various regulations on the permitted levels of the radiated and conducted emissions from electronic devices. Commercially available SMPSs are designed to meet these regulations. For example, the FCC Part 15 standard requires that the conducted emission be controlled at the frequency bands of 450kHz-30MHz [48], [49].

III. Attack Model

A. Proposed Concept

The proposed method *exploits* the conducted emission to modulate digital information on the power lines. We show that a malicious code can influence the momentary power consumption of the computer, generating data-modulated conduction on the power lines in the low frequency band. The generated noise travels along the input power lines and can be measured by an attacker probing the power cables. The proposed method works with computers that are fully compliant with conduction emission regulations, as discussed below in Section VI-A.

The adversarial attack model consists of three main steps: (1) system infection and data gathering, (2) transceiver implantation, and (3) data exfiltration.

1) System Infection & Data Gathering: The system infection step is common for all air-gap covert channel attacks, since it requires running a malicious code in the targeted air-gapped computer [50]. In the incursion phase, the attacker infects the target system or network with malware. Infecting highly secure and even air-gapped networks has been proven feasible in recent years. Note that several APTs discovered in the last decade are capable of infecting air-gapped networks [51], e.g., Turla [52], RedOctober [53], and Fanny [54]. As a part of the targeted attack, the adversary may infiltrate the air-gapped networks using social engineering, supply chain attacks, or malicious insiders.

Having a foothold in the system, the malware starts retrieving interesting data for the attacker. The data might be files, encryption keys, credential tokens, or passwords. The data gathered is stored on a computer, usually the PC workstation or server which contains the sensitive data to leak.

2) Transceiver Implantation: The transceiver is based on a non-invasive probe connected to a small computer. The probe is placed near the power line feeding the computer or at the main electric panel (see also Fig. 6 below). It measures the conducted emission, processes the modulated signals, decodes the data and sends it to the attacker (e.g., with a Wi-Fi transceiver).

The required hardware implant is similar to that required for a broad kind of differential power analysis and related attacks [40]. Another example of such a hardware implant was presented in Snowden's leaked documents. In that case, the component, known as COTTONMOUTH [55], was a USB connector with a hidden RF transceiver that attackers used to connect with air-gapped networks by sending and receiving data to/from a long haul relay subsystem.

3) Data Exfiltration: In the last phase of the attack, the malware starts the data leak by encoding the data and transmitting it via signals injected to the power lines. The signals are generated by changing the workload on the CPU cores. The transmissions may take place at predefined times or in response to some trigger infiltrated by the attacker. The signal is received by the power line probe and delivered to the attacker (e.g., via Wi-Fi).

Note that although the described attack model is complicated, it is not beyond the ability of motivated and capable attackers. In the last decade, APTs coupled with sophisticated attack vectors such as supply chain attacks and targeted social engineering have been shown to be feasible. As a reward for these efforts, the attacker can get his/her hands on very valuable and secured information, which is out of reach of other types of covert channels.

B. Signal Generation (Transmitter)

As described above (Sec. II-B), changes in the power consumption of a computer are conducted to the power lines through the power supply. Since modern CPUs are energy efficient, the momentary workload of the CPU directly affects the dynamic changes in its power consumption [56]. By intentionally starting and stopping the CPU workload, it is possible to govern its power consumption, and to conduct a parasitic signal on the electrical network lines at a specified frequency and to modulate binary data over it (Fig. 1). We developed a fine-grained approach, in which we control the workload of each of the CPU cores independently from the other cores. Regulating the workload of each core separately enables greater control of the momentary power consumption. This approach has two main advantages:

1) Choosing which cores to operate on at a given time allows us to use only the currently available cores, that is, cores which are not utilized by other processes. This way, the transmission activity will not interrupt other active processes in the system. This is important for the usability of the computer and the stealth of the covert channel.
2) By using different numbers of cores for the transmission, we can control the current consumption (e.g., fewer cores consume less power), and hence the amplitude of the carrier wave. This allows us to employ amplitude-based modulations in which data is encoded on the amplitude level of the signal.

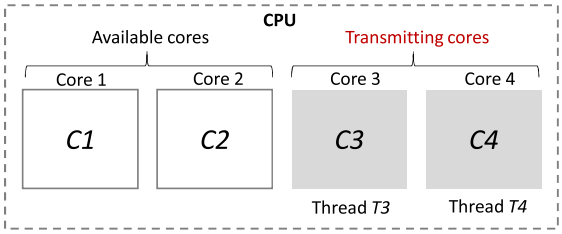

Fig. 3. A CPU with two transmitting threads.

Algorithm 1 WorkerThread ($iCore$, $freq$, $nCycles0$, $nCycles1$)

```
1: bindThreadToCore(iCore)
2: half_cycle_ms ← 0.5 * 1000/freq
3: while (!endTransmission()) do
4:    if (data[i] = 0) then
5:       sleep(nCycles0*half_cycle_ms*2)
6:    else
7:       for j ← 0 to nCycles1 do
8:          T1 ← getCurrentTime()
9:          while (getCurrentTime() − T1 < half_cycle_ms) do ;
10:         end while
11:         sleep(half_cycle_ms)
12:      end for
13:   end if
14: end while
```

To generate a carrier wave at frequency f_c in one or more cores, we control the utilization of the CPU at a frequency related to f_c. To that end, n worker threads are created, where each thread is bound to a specific core. To generate the carrier wave, each worker thread overloads its core at a frequency f_c repeatedly – alternating between applying a continuous workload on its core for a time period of $1/2f_c$ (full power consumption) and putting its core in an idle state for a time period of $1/2f_c$ (no power consumption).

This operation is illustrated in Fig. 3, which depicts a system with two worker threads. Threads $T3$ and $T4$ are bound to cores $C3$ and $C4$, respectively. Note that cores $C1$ and $C2$ do not participate in this transmission. When the worker threads $T3$ and $T4$ start, they receive the required carrier frequency f_c and the stream of bits to transmit. The basic operation of a worker thread is described in Algorithm 1.

Each worker thread receives the core to be bound to ($iCore$) and the carrier frequency ($freq$). It also receives the number of cycles for the modulation of logical '0' ($nCycles0$) and the number of cycles for the modulation of logical '1' ($nCycles1$).

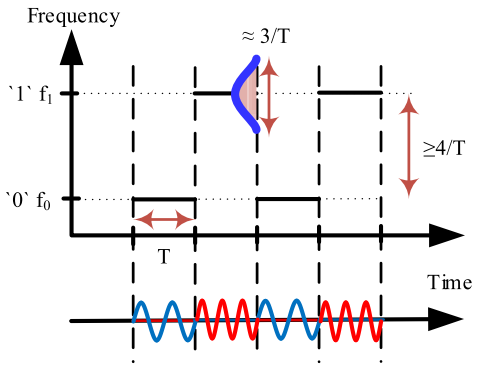

Fig. 4. Illustration of BFSK modulation.

Note that the cycle time is derived from the frequency of the carrier wave (line 2). The thread's main function iterates on the array of bits to transmit. In the case of logical '0' it sleeps for $nCycles0$ cycles (line 5). In the case of logical '1' it repeatedly starts and stops the workload of the core at the carrier frequency $freq$ for $nCycles1$ cycles (lines 7-12). We overload the core using the busy waiting technique. This causes full utilization of the core for the time period and returns.

Based on the algorithm above, we implemented a transmitter for Linux Ubuntu (version 16.04, 64 bit). We used the `sched_setaffinity` system call to bind each thread to a CPU core. The affinity is the thread level attribute that is configured independently for each worker thread. To synchronize the initiation and termination of the worker threads, we used the thread mutex objects with `pthread_mutex_lock()` and `pthread_mutex_unlock()` [57]. For the thread sleeping we used the `nanosleep()` system call [58]. Note that the precision of `nanosleep()` is in nanoseconds, and it is sufficient given the frequencies of the carrier waves, which are at 24kHz or lower.

C. Modulation

Recall that the transmitting code can determine the frequency of the signal by setting the cycle time in the signal generation algorithm. In the current research we used frequency shift keying (FSK) for data modulation. In FSK, the data is represented by a change in the frequency of a carrier wave. For the evaluation, we mainly used the binary-FSK (BFSK) modulation [59], which is outlined in Fig. 4. In this modulation, only two different carrier frequencies are employed. Each frequency is amplitude modulated, such that '1' and '0' are two possible symbol combinations. In FSK, the length of each symbol is T, and the spectral width is approximately $\pm 1.5/T$. Theoretically, this modulation requires at least $4/T$ frequency spacing between different carrier frequencies.

The main motivation for choosing FSK modulation is that it can be used with non-coherent receivers. This means that it does not require frequency and phase recovery of the exact f_0 and f_1 signals, but only symbol timing synchronization. The typical block-scheme of such a receiver is presented in Fig. 5. It includes band-pass filters (BPFs) BPF_0 and BPF_1 that are located around frequencies f_0 and f_1, respectively. The energy

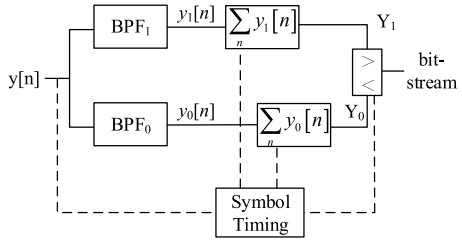

Fig. 5. Block-scheme of non-coherent FSK receiver.

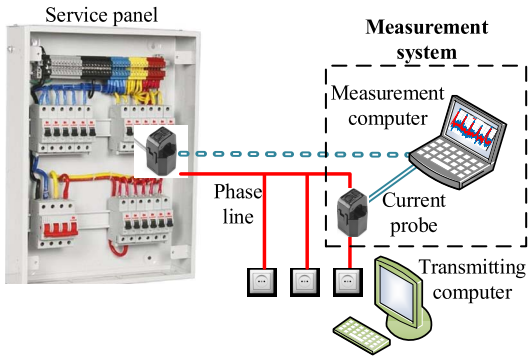

Fig. 6. The setup for electrical network measurements.

of $y_0[n]$ and $y_1[n]$ signals is compared over a time period of T and the received bit is determined by the highest energy value.

D. Bit-Framing

The data packets are transmitted in frames. Like most air-gap covert channels, the unidirectional communication means that the receiver cannot establish a handshake with the transmitter, and hence cannot identify the beginning of a transmission and determine the physical-layer channel parameters. To solve this, a preamble header is transmitted at the beginning of every packet. After the preamble, a payload is transmitted. Finally, a cyclic redundancy check (CRC) is applied.

IV. Experimental Validation

In this section we provide experimental evaluation of the PowerHammer attack and discuss the measurements and the results.

A. Measurement Setup

Our measurement setup consists of a current probe connected to a measurement system (Fig. 6). For measuring the current, we used a type of split core current transformer. This is a non-invasive probe that is clamped around the power line and measures the amount of current passing through it (Fig. 7). The non-invasive probe behaves like an inductor that responds to the magnetic field around a current-carrying cable (Fig. 7b). The amount of current in the coil is correlated with the amount of current flowing in the conductor. For our experiments we used SparkFun's split core current transformer ECS1030-L72 [60]. The current sensor is connected through a 3.5mm audio cable to the audio input jack of a laptop computer

TABLE I
THE COMPUTERS USED FOR THE TRANSMISSION EXPERIMENTS

#	Name	Model	Motherboard	CPU	PSU (SMPS)
1	**PC**	Silverstone Desktop	Gigabyte H87M-D3H	Intel Core i7-4770 CPU@ 3.4GHz, 4 cores (8 threads)	FSP300-50HMN 300W
2	**Server**	IBM System x3500 M4	Intel C602J	Intel Xeon CPUE5-2620, 12 cores (24 threads)	DPS-750AB-1 A 750Wx2
3	**Low power device (IoT)**	Raspberry Pi 3	Raspberry Pi 3, Model B V1.2	Quad Core Broadcom BCM2837, 64-bit, ARMv8, processor Cortex A53	Stontronics DSA-13PFC-05, 5V 2.5A

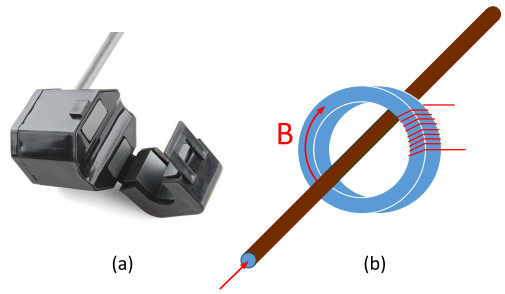

Fig. 7. (a) Current probe. (b) Split core current transformer principle.

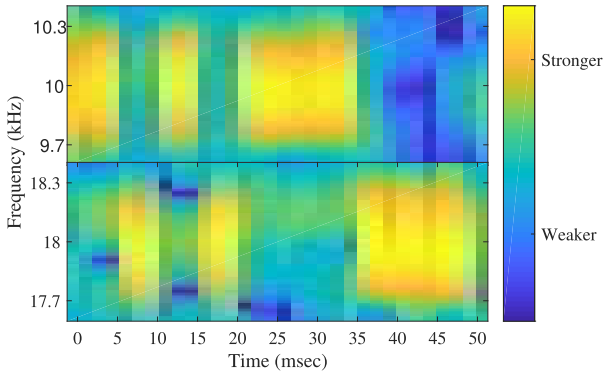

Fig. 8. Spectrogram of BFSK transmission at 200 bit/sec and communication frequencies $f_0 = 10$kHz and $f_1 = 18$kHz; the transmitted signal is the '0101000111' sequence.

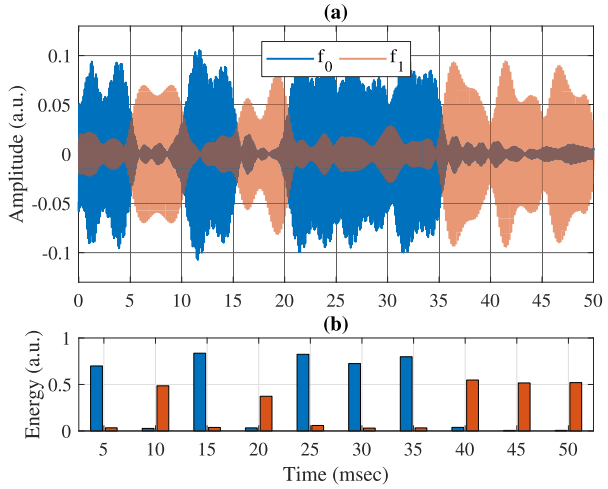

Fig. 9. (a) Time domain plot of the signal in Fig. 8. (b) Corresponding signal energies.

(Dell Latitude E7450, Windows 10 64-bit). The sampling rate was set to 48kHz with a 16-bit resolution. A sampled signal was further processed and analyzed by Matlab. For real-time demodulation we used a custom-developed receiver coded in Visual C++.

B. Transmitting Computers

Throughout the experiments, the following three types of computers were addressed: a desktop PC, a server, and a low power device representing an IoT. A list of the computers with their specifications is provided in Table I. With the low power device (Raspberry Pi 3), we checked the feasibility of transmitting from IoT devices that only have low power consumption. The PC was running a Linux Ubuntu OS version 16.04.4 LTS 64-bit. The server was running a Linux Ubuntu OS version 16.04.1 LTS 64-bit (kernel version 4.4.0). The Raspberry Pi was running the Raspbian Strech OS (kernel version 4.9). The transmitter presented in Section III-B was compiled with GCC and executed on each of the three computers.

C. Modulation

An example of a spectrogram (also termed a short-time Fourier transform (STFT)) of the '0101000111' sequence signal is presented in Fig. 8, where $T = 5$ msec, which yields a bit rate of 200 bit/sec, and communication frequencies $f_0 = 10$kHz and $f_1 = 18$kHz. The time spacing of the spectrogram was chosen to match a symbol length of two segments per symbol. The spectral width of the signals matches the theoretical bandwidth of $\cong 3/T = 600$Hz. The presented signal is fully congruent with BFSK signal theory.

An example of receiver signals $y_0[n]$ and $y_1[n]$ (outlined in Fig. 4 above) is presented in Fig. 9(a) for the same part of the signal as in Fig. 8. For each symbol, corresponding energies of $y_0[n]$ and $y_1[n]$ are presented in Fig. 9(b).

An example of a BFSK signal is presented in Fig. 9. The signal (measured from the power line) was generated using the transmitting algorithm shown above. In this transmission (a sequence of '0101000111') the symbol time-length is $T = 5$ msec, which yields a bit rate of 200 bit/sec. For each symbol, energies of $y_0[n]$ and $y_1[n]$ are compared, as outlined in Fig. 9(b). The theoretical analysis of the corresponding bit-error rate (BER) is presented in the Appendix.

D. Signal Power

1) Number of Cores: The number of cores used in the transmission directly influences the power consumption of the

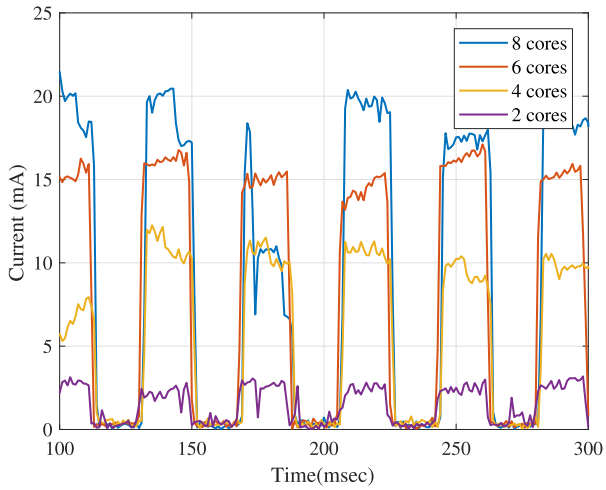

Fig. 10. The waveform of a transmission with different numbers of cores.

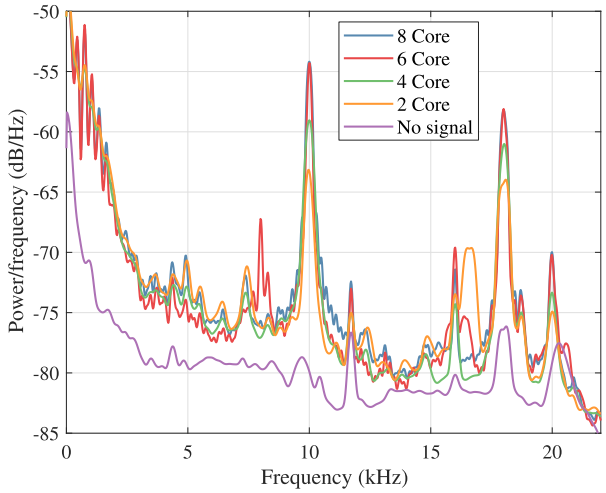

Fig. 11. PSD of a transmission with different number of cores at two different frequencies, $f_0 = 10$kHz and $f_1 = 18$kHz.

computer (e.g., more transmitting threads result in greater power consumption), and, hence, the amount of conductive emission. The measurements in Fig. 10 show the waveform of an alternating signal ('101010101010') transmitted from the PC at a modulation frequency of 10kHz. The transmissions in two, four, six, and eight cores yield signal levels of 2.5mA, 12mA, 15mA, and 19mA, respectively. As can be seen, the number of cores used for the transmissions is correlated with the conducted emission measured. However, it is important to note that although using more cores yields a stronger signal, the attacker may use only some of the available cores for the transmissions; using all the cores will significantly affect the workload on the system and might reveal the malicious activity.

An illustration of the power spectral density (PSD) of the modulated signal, estimated using the Welch periodogram method, is presented in Fig. 11. This graph shows the approximately linear dependency of the communication signal power on the number of transmitted cores at two different frequencies, $f_0 = 10$kHz and $f_1 = 18$kHz.

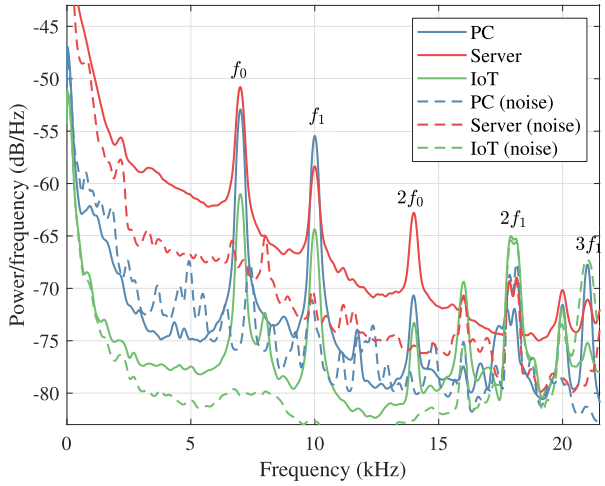

Fig. 12. Comparison of the PSD of the communication signal at $f_0 = 7$kHz and $f_1 = 10$kHz for the computers in Table I.

2) Frequency Dependence & Transmitting Computer Dependence: The difference between the PSDs of communication signals for different computers (Table I) is presented in Fig. 12. The "noise" is defined as the signal level at the communication frequencies when no signal is transmitted at the particular frequency at that moment. As can be seen in Fig. 12, each signal produces harmonics at the frequencies that are multiples of the signal frequency. This aspect is important for the selection of FSK communication frequencies. Note, the signal measured from the server is relatively weak (f_1 frequency) compared to an ordinary workstation due to the high-end ('smart') power supplies with improved load efficiency and smart voltage regulation, which results in lower conducted emission.

Figs. 11 & 12 also explain the differences in symbol energies in Fig. 9(b), since the transmitted signal power/energy depends on frequency. Moreover, this difference has to be further compensated at the receiver in order to improve communication performance.

E. Bit-Error Rate

The theoretical analysis of BER performance is based on a *noise margin* that is defined as the difference between signal and noise levels. For the applied scheme (non-coherent BFSK), the theory requires a noise margin level of 11 dB for a BER of 10^{-3} and 8 dB for a BER of 10^{-2} [59]. An illustration of noise margins for the signals in Fig. 12 for the PC and IoT devices is presented in Fig. 13. In the graphs, the noise margin is about 13 dB for 200 bps communication, which is congruent with our measured 0% BER for this configuration.

V. Communication Performance

The goal of this section is to discuss communication performance measures of the reported attack.

A. Bit-Rate

We note that the reported communication performance may be significantly increased by applying *M*ary-FSK (M-FSK)

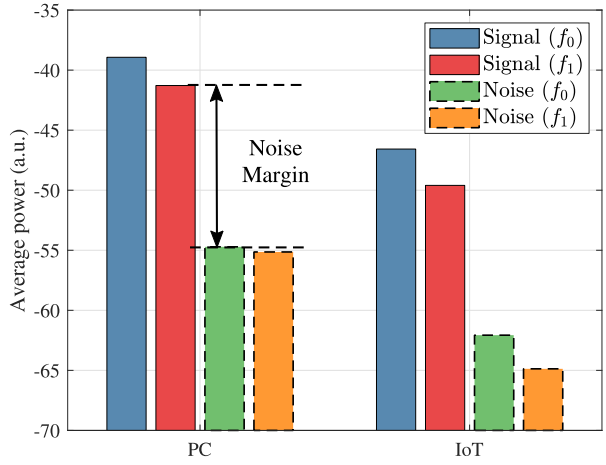

Fig. 13. Noise margin evaluated for the scenario in Fig. 12, for two different transmitting computers.

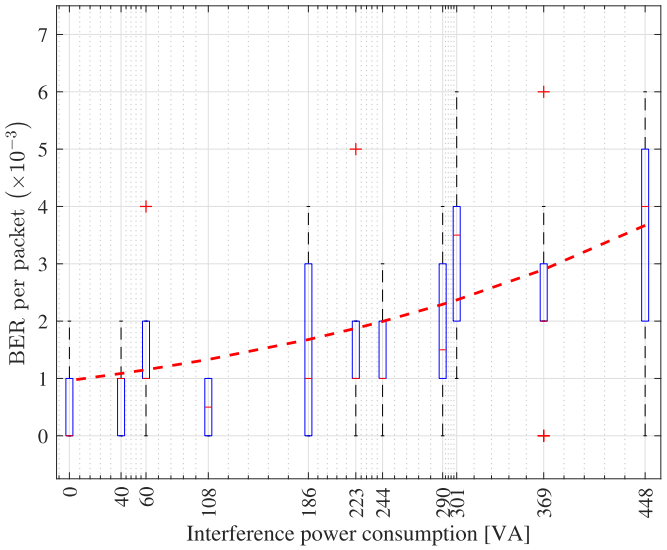

Fig. 14. Boxplot of errors per packet as a function of interference power consumption. The trend-line is of the form $f(x) = \frac{1}{2}\exp(-(ax+b)/2)$, where $a = 12.5$ is the SNR without interfering computers and $b \approx -6 \times 10^{-3}$.

modulation instead of BFSK. In this modulation, M different carrier frequencies are used instead of the two carriers in the case of BFSK. The maximum bit-rate is limited by two factors: the minimum achievable symbol duration, T, and the available communication bandwidth, which are related by

$$B = M\Delta f + B_c \cong \frac{4M+3}{T}, \tag{1}$$

where $B_c = 3/T$ is the bandwidth of each modulated carrier and $\Delta f = 4/T$ is the minimum spacing between modulation frequencies. Note, the bandwidth is limited not only by the ability of the transmitter to generate the required frequencies, but also by the harmonics limitation mentioned above (Sec. IV-D.2). The corresponding BER is dictated by the frequency band with the lowest noise margin. In our experiments, we successfully generated symbols with $T = 0.001$ seconds duration that correspond to a symbol rate of $R_s = 1000$ symbol/sec and a 8-FSK bit-rate of 3 kbps, under optimal communication conditions.

B. Distance

An important characteristic of a communication channel is the distance-related power attenuation of the received signal. In order to quantify the influence of the communication distance, we used the following experimental setup. The transmitter and the receiver were placed at different distances from one another, between 2 and 110 meters, and were interconnected by a single network-connected cable. For all the measurements, the communication speed was set to 1 kbps and 10 packets of 1 kilobit were transmitted. The measured per-packet number of errors was between 0 and 5, resulting in an average BER of $\lesssim 2 \times 10^{-3}$, and the influence of the separation distance was found to be negligible. These results are, in fact, not surprising, based on the previously published research in the fields of ultra-narrowband and narrowband power-line communication (UNB-PLC and NB-PLC) [61], [62]. However, PLC theory also predicts that the communication performance may degrade as a result of branching cables, impedance mismatches and other factors. Nevertheless, an in-depth discussion of this issue is beyond the scope of this paper and may be the subject of future research.

C. Interference

Each common power-line shares emissions conducted from all the same-line-connected electronic devices, such as computers, TVs, lights, etc. While these emissions do not include any particular information, they interfere with the communication signal emission of interest. In order to quantify the influence of the interfering devices, we used the following experimental setup. Up to 10 devices (5 PCs and monitors) with SMPS power supplies were connected on the same electrical network line with the transmitter. The power consumption of these devices was evaluated by voltage and current measurements (NI 9225 voltmeter, NI 9227 ammeter, sample frequency of 25 kHz). For different power consumption levels, 10 packets of 1 kilobit were transmitted at 1 kbps. The BER performance during the experimental evaluation is summarized in Fig. 14 as a box-plot of errors per packet as a function of interference power consumption. Since the approximated BER is given by the relation [59]

$$p_e \approx \exp\left(-\frac{\text{SNR}}{2}\right), \tag{2}$$

where SNR is the signal-to-noise ratio, we have chosen the trend-line of the form

$$f(x) = \frac{1}{2}\exp\left(-\frac{a+bx}{2}\right), \tag{3}$$

where a is the SNR without interfering computers and $a + bx$ is the SNR for the given interference power consumption. The resulting fit values for the presented trend-line are $a = 12.5, b \approx -6 \times 10^{-3}$.

For higher levels of interference, better performance can be achieved by the common interference mitigation techniques, such as integration/summation over a longer symbol time,

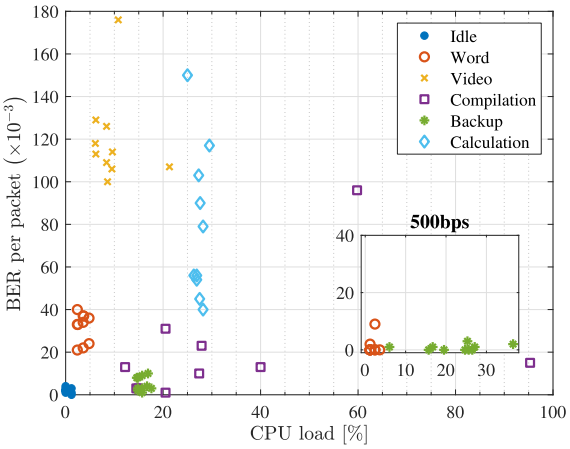

Fig. 15. Number of errors per packet for different CPU workloads conditions. The inset illustrates the BER improvement for bit-rate drop from 1 kpbs to 500 bps.

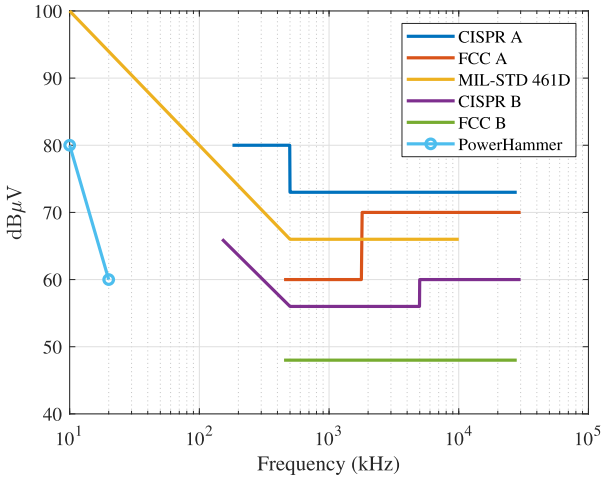

Fig. 16. Comparison of PowerHammer with conducted emission standards.

spreading sequences, and others [59], resulting in a lower bit-rates.

D. CPU Load

Since the transmitting process (and its theads) shares the CPU with other processes in the operating system, we examined whether the activity of other processes interferes with the signal generation. For the evaluation, we tested the following workloads on the system: (1) The system is idle and only the default processes are running in the background; (2) The LibreOffice Writer [63] is open, and the user is typing a document; (3) The VLC media player [64] is playing an HD video clip; (4) The Linux `rsync` [65] command is performing a backup to the HDD; (5) The Linux `matho-primes` [66] is performing calculations of big prime numbers.

For different workloads, 10 packets of 1 kilobit at 1 kbps were transmitted. The BER performance during the experimental evaluation is summarized in Fig. 15, where each point represents a different packet. The results show a dramatic influence of CPU load on communication performance. The particular case of the VLC media player is most probably related to variable load on the GPU rather than on the CPU. In general, the performance degradation is related to alternating bursts of CPU load that are produced by workloads. However, a drop in bit-rate may significantly improve the communication performance, as outlined in the inset for a 500 bps bit-rate. A further drop in bit-rate may further change the BER for any required workload conditions. From the theoretical point of view, this improvement is related to the fact that the SNR is proportional to the symbol length, i.e. an increase of symbol length by a factor of two (from 1 msec to 2 msec) is expected to provide twofold improvement in the SNR [59].

VI. Undetectability & Countermeasures

A. Conducted Emission Standards

There are civilian and military standards for the permitted levels of emission conducted from electronic devices to power lines. These standards specify the strength of the electric field (dBμV) allowed in different frequency bands. Fig. 16 shows the conducted emission limits defined by the three international standards: (1) FCC Part 15 (the American standard), (2) CISPR 22 (the European standard), and (3) MIL-STD 461 (the military standard), as well as the strength of the electric fields generated with PowerHammer transmissions. Note that the FCC and CISPR standards have two classes, defined as class A and class B; the conducted emission limits are more strict for class B. The FCC Part 15 standard states that the conducted emissions must be controlled in the 0.45MHz-30MHz frequency band. The CISPR 22 standard states that the conducted emissions must be controlled in the 0.15MHz-30MHz frequency band. The PowerHammer transmissions are carried out in the band of 0-0.024MHz (0-24kHz), which is significantly below the FCC and CISPR limits in terms of frequency. The MIL-STD 461 standard states that the conducted emissions must be controlled in the 30Hz-10kHz frequency band; in the range of 0-24kHz the emissions must be kept below the level of 100-95dBμV. The strength of the electric fields generated in our covert channel is significantly lower than the maximum defined by MIL-STD 461 (ranges from 20 to 30dBμV), and hence is compliant with this standard.

Note that the civilian and military standards for conducted emission are aimed at preventing harmful interference between devices. We exploit the conducted emission by using low strength electric fields that do not cause standard-prohibited interference, but can still be used to carry information.

B. Conductive Emission Filtering

Typical civilian and military electrical network filters are designed for limiting the conducted emission for frequencies starting at about 20 kHz e.g., [67]. As shown in Section VI-A above, the generated signals are at frequencies lower than 20 kHz, and hence bypass most of the common filters.

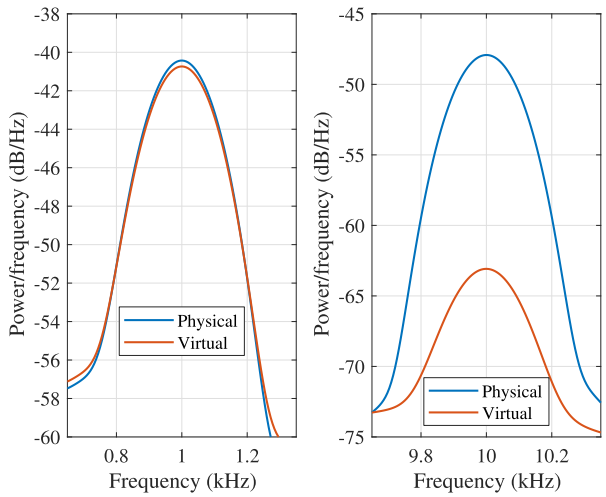

Fig. 17. The signals generated from physical and virtual machines at 1kHz and 10kHz communication frequencies.

C. Virtual Machines (VMs)

Virtualization technologies are widely used in modern IT environments. In the context of security, one of the advantages of virtualization is the resource isolation it provides. Virtual machine monitors (VMMs) provide a layer of abstraction between the virtual machine and the physical hardware (CPU and peripherals). Since the signal generated is tightly related to the CPU timing and hardware performance, we examined whether the virtualization layer caused interruptions or delays that may affect the signal quality.

In order to evaluate the effect of the VM, we compared the signals generated from a physical machine (host) and the signals generated from a VM (guest). In these measurements, the host and the guest were running on a Linux Ubuntu 16.04 64-bit. We used VMWare Workstation Player 14.0 and configured the guest to support four processors. For the comparisons, an identical sweep signal in the range of 0-24kHz was transmitted from both the host and the guest. The analysis of the signals received showed that while the host was able to generate signals in the whole inspected frequency range of 0-24kHz, the guest could only generate signals up to 7kHz. Fig. 17 shows the PSD of transmissions made by the host and guest machines with two different frequencies (1kHz and 10kHz). As can be seen, the 1kHz signal is similar in the host and guest (with levels of -41dB). The 10kHz guest signal (-63dB) is significantly weaker than the host signal (-47dB).

We found that the 7kHz limitation shown above is a result of the interruptions occurring in the transmitting threads when executed in a VM. Technically, the VMM initiates a periodical context switch, which suspends the transmitting process (and its threads), in order to transfer the control to the host machine. These interruptions effectively limit the operational frequency of the transmitting threads to 7kHz.

D. Stealth & Intrusion Detection

The transmitting program only leaves a small footprint in the memory, making its presence easier to hide from intrusion detection systems. At the OS level, the transmitting program requires no special or elevated privileges (e.g., root or admin), and hence can be initiated from an ordinary user space process. The transmitting code consists mainly of basic CPU operations such as busy loops, which do not expose malicious behaviors, making it somewhat evasive from automated analysis tools.

Host-based intrusion detection systems (HIDS) may continuously trace the activities of running processes in order to detect suspicious behavior; in our case, a group of threads that abnormally regulates the CPU workloads would be reported and inspected [68]. However, such a detection approach may suffer from false alarms, since many legitimate processes use CPU-intensive calculations that affect the processor's workload [69]. Moreover, since only non-privileged CPU instructions are involved, monitoring such a process may degrade performance [16]. Nevertheless, the malware has to inject the transmitting threads into a legitimate, trusted process to bypass the security mechanisms [70].

E. Uninterruptible Power Supply

While an uninterruptible power supply (UPS) that is connected between a computer and power sockets may be an effective countermeasure, typical UPS installation is at the main electrical panel (Fig. 2). The most effective countermeasure of this kind is a special online double conversion UPS that isolates the power supply from the electricity mains [71].

F. Power Lines Monitoring

A primary defensive countermeasure in the field of covert channels is monitoring the channel in order to detect the presence of covert communication [72], [73]. In our case, it is possible to detect the covert transmissions by monitoring the current flow on the power lines. The most effective dedicated countermeasure to the proposed attack is power line monitoring. The recent progress in machine learning enables commercially-available identification and profiling of almost each network-connected device [74]. An appropriate anomaly detection algorithm is expected to identify power-line communication attempts. Beyond the possibility of detection of unwanted communication, sufficiently effective monitoring is expected to identify new network-connected devices and to detect abnormal consumption patterns. Moreover, it can simultaneously serve as a power consumption analyzer that drives electrical energy savings [75]. The main disadvantage of this method is that power-monitoring signals are confidential information that may be further used for different attacks e.g., [40], [76].

G. Wireless Monitoring

One of the possible countermeasures is intercepting and monitoring wireless signals. Since the implanted transceiver has to use some kind of wireless communication, its presence may be detected. While effective monitoring methods were proposed for standard communication protocols

(e.g., [77]), using special secure communication techniques such as spread-spectrum or frequency-hopping may significantly burden the detection of the illegitimate communications.

VII. Summary & Conclusion

In this paper we presented PowerHammer, a type of attack that uses *conducted emission* through electrical network lines to exfiltrate data from air-gapped computers. Data is modulated by a malicious code that controls the CPU load and is then conducted from the power supply to the power lines. The attack is based on placing a probe on the network cable feeding the computer. The attacker measures the emitted signal, processes it and decodes the transmitted binary information.

The implemented communication method provides a bit-rate of up to a few kbps under optimal communication conditions. The communication performance may be influenced by workloads of the communication computer and a power-consuming devices on the same communicating power line. In this case, lower communication bit-rates are recommended to preserve a reasonable BER.

Appendix
BER Analysis

The error probability of the received signal is described using a comparison of energies, $E_0 \lessgtr E_1$. In order to evaluate the expected BER, the values of these energies may be represented by random variables. These variables are conditioned on particular transmitted binary signals at each frequency i.e., $E_{i|j}$, were $i \in \{0,1\}$ is a transmission frequency and $j \in \{0,1\}$ is a transmitted value at that frequency (see Figs. 4 and 9). Following the central limit theorem, these variables may be assumed to be Gaussian, such that

$$E_{0|0} \sim N\left(\mu_{0|0}, \sigma^2_{0|0}\right), \quad E_{1|1} \sim N\left(\mu_{1|1}, \sigma^2_{1|1}\right),$$
$$E_{0|1} \sim N\left(\mu_{0|1}, \sigma^2_{0|1}\right), \quad E_{1|0} \sim N\left(\mu_{1|0}, \sigma^2_{1|0}\right), \tag{4}$$

where $\mu_{0|0}, \mu_{0|1}, \mu_{1|0}, \mu_{1|1}$ and $\sigma^2_{0|0}, \sigma^2_{0|1}, \sigma^2_{1|0}, \sigma^2_{1|1}$ represent received signals and their variance due to accompanying noise.

For example, when '0' and '1' are encoded by symbols [0 1] and [1 0], the corresponding error probabilities are given by

$$p_{e_0} = \mathcal{P}(E_{0|0} > E_{1|1}), \tag{5a}$$
$$p_{e_1} = \mathcal{P}(E_{0|1} < E_{1|0}). \tag{5b}$$

An illustration of $p_{e_{10}}$ is presented in Fig. 18.

Error probability p_{e_0} is given by [78]

$$p_{e_0} = 1 - \frac{1}{\sigma_{0|0}\sqrt{2\pi}} \int_{-\infty}^{\infty} Q\left(-\frac{v - \mu_{1|1}}{\sigma_{1|1}}\right) \times \exp\left[-\frac{\left(v - \mu_{0|0}\right)^2}{2\sigma^2_{0|0}}\right] dv \tag{6}$$

and the corresponding p_{e_1} value may be easily derived by changing the indices of μ and σ. Finally, using the equal probability of '0's and '1's, $\mathcal{P}('0') = \mathcal{P}('1') = 1/2$, the average BER is given by

$$p_e = \frac{1}{2} p_{e_{01}} + \frac{1}{2} p_{e_{10}}. \tag{7}$$

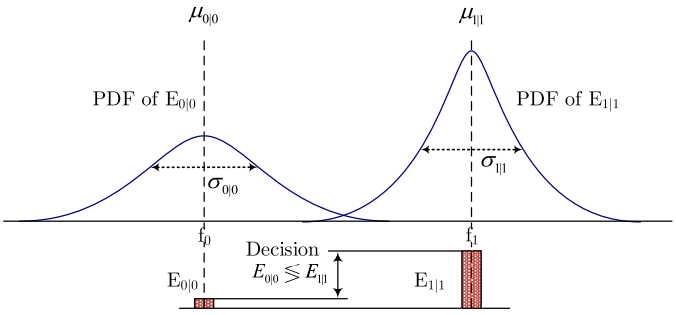

Fig. 18. Illustration of p_{e_0} error probability evaluation for [0 1] symbol as defined by Eq. (5a).

When multiple devices are connected to the same electrical network line, they produce excessive interference in the communication channel. In this case the interference energies $E_{0|0}$ and $E_{1|0}$ have to be updated accordingly.

References

[1] E. MacAskill, S. Thielman, and P. Oltermann, "WikiLeaks publishes 'biggest ever leak of secret CIA documents,"' The Guardian, 2017, vol. 26.

[2] K. Zetter, "Sony got hacked hard: What we know and don't know so far," Wired, Jan. 2014.

[3] S. Thielman. (Dec. 2016). *Yahoo Hack: 1bn Accounts Compromised by Biggest Data Breach in History*. Accessed: Dec. 3, 2019. [Online]. Available: https://www.theguardian.com/technology/2016/dec/14/yahoo-hack-security-of-one-billion-accounts-breached

[4] M. Guri, M. Monitz, and Y. Elovici, "Bridging the air gap between isolated networks and mobile phones in a practical cyber-attack," *ACM Trans. Intell. Syst. Technol.*, vol. 8, no. 4, p. 50, 2017.

[5] E. Byres, "The air gap: Scada's enduring security myth," *Commun. ACM*, vol. 56, no. 8, pp. 29–31, 2013.

[6] *Classified United States Website—Wikipedia*. Accessed: Dec. 3, 2017. [Online]. Available: https://en.wikipedia.org/wiki/Classified_United_States_website

[7] M. Maybury *et al.*, "Analysis and detection of malicious insiders," in *Proc. Int. Conf. Intell. Anal.*, McLean, VA, USA, May 2005. [Online]. Available: https://www.mitre.org/sites/default/files/pdf/05_0207.pdf

[8] E. Osnos, D. Remnick, and J. Yaffa. *Trump, Putin, and the New Cold War—The New Yorker*. Accessed: Dec. 3, 2017. [Online]. Available: https://www.newyorker.com/magazine/2017/03/06/trump-putin-and-the-new-cold-war

[9] S. Abraham and I. Chengalur-Smith, "An overview of social engineering malware: Trends, tactics, and implications," *Technol. Soc.*, vol. 32, no. 3, pp. 183–196, 2010.

[10] Wikileaks. *CIA Uses 'Brutal Kangaroo' Toolkit to Hack Air-Gapped Networks.* Accessed: Dec. 13, 2017. [Online]. Available: https://www.theinquirer.net/inquirer/news/3012499/-wikileaks-cia-uses-brutal-kangaroo-toolkit-to-hack-air-gapped-networks

[11] H. O. Hundley and R. H. Anderson, "Emerging challenge: Security and safety in cyberspace," *IEEE Technol. Soc. Mag.*, vol. 14, no. 4, pp. 19–28, Winter 1995.

[12] M. Guri, G. Kedma, A. Kachlon, and Y. Elovici, "AirHopper: Bridging the air-gap between isolated networks and mobile phones using radio frequencies," in *Proc. 9th Int. Conf. Malicious Unwanted Softw., Amer.*, Oct. 2014, pp. 58–67.

[13] M. G. Kuhn and R. J. Anderson, "Soft tempest: Hidden data transmission using electromagnetic emanations," in *Information Hiding*, D. Aucsmith, Ed. Berlin, Germany: Springer, 1998, pp. 124–142.

[14] M. G. Kuhn, "Compromising emanations: Eavesdropping risks of computer displays," Comput. Lab., Univ. Cambridge, Cambridge, U.K., Tech. Rep. 577, Dec. 2003. [Online]. Available: https://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-577.pdf

[15] M. Vuagnoux and S. Pasini, "Compromising electromagnetic emanations of wired and wireless keyboards," in *Proc. USENIX Secur. Symp.*, 2009, pp. 1–16.

[16] M. Guri, A. Kachlon, O. Hasson, G. Kedma, Y. Mirsky, and Y. Elovici, "GSMem: Data exfiltration from air-gapped computers over GSM frequencies," in *Proc. 24th USENIX Secur. Symp.*, Aug. 2015, pp. 849–864.

[17] N. Matyunin, J. Szefer, S. Biedermann, and S. Katzenbeisser, "Covert channels using mobile device's magnetic field sensors," in *Proc. 21st Asia South Pacific Design Automat. Conf.*, Jan. 2016, pp. 525–532.

[18] (2015). *Funtenna · GitHub*. Accessed: Dec. 3, 2017. [Online]. Available: https://github.com/funtenna

[19] C. Kasmi, J. L. Esteves, and P. Valembois, "Air-gap limitations and bypass techniques:'Command and control' using smart electromagnetic interferences," *J. Cybercrime Digit. Investigations*, vol. 1, no. 1, pp. 1–7, 2016.

[20] Z. Yang, Q. Huang, and Q. Zhang, "NICScatter: Backscatter as a covert channel in mobile devices," in *Proc. 23rd Annu. Int. Conf. Mobile Comput. Netw.*, New York, NY, USA, 2017, pp. 356–367.

[21] Z. Zhou, W. Zhang, and N. Yu, "Data exfiltration via multipurpose RFID cards and countermeasures," Feb. 2019, *arXiv:1902.00676*. [Online]. Available: https://arxiv.org/abs/1902.00676

[22] M. Guri, B. Zadov, and Y. Elovici, "ODINI: Escaping sensitive data from faraday-caged, air-gapped computers via magnetic fields," *IEEE Trans. Inf. Forensics Security*, to be published.

[23] M. Guri, A. Daidakulov, and Y. Elovici, "MAGNETO: Covert channel between air-gapped systems and nearby smartphones via cpu-generated magnetic fields," 2018, *arXiv:1802.02317*. [Online]. Available: https://arxiv.org/abs/1802.02317

[24] A. Madhavapeddy, R. Sharp, D. Scott, and A. Tse, "Audio networking: The forgotten wireless technology," *IEEE Pervasive Comput.*, vol. 4, no. 3, pp. 55–60, Jul. 2005.

[25] L. Deshotels, "Inaudible sound as a covert channel in mobile devices," in *Proc. 8th USENIX Conf. Offensive Technol.*, 2014, p. 16.

[26] M. Hanspach and M. Goetz, "On covert acoustical mesh networks in air," 2014, *arXiv:1406.1213*. [Online]. Available: https://arxiv.org/abs/1406.1213

[27] B. Carrara and C. Adams, "On acoustic covert channels between air-gapped systems," in *Proc. Int. Symp. Found. Pract. Secur.* Cham, Switzerland: Springer, 2014, pp. 3–16.

[28] E. Lee, H. Kim, and J. W. Yoon, "Various threat models to circumvent air-gapped systems for preventing network attack," in *Proc. Int. Workshop Inf. Secur. Appl.*, vol. 9503. Cham, Switzerland: Springer, 2015, pp. 187–199.

[29] M. Guri, Y. Solewicz, A. Daidakulov, and Y. Elovici, "Acoustic data exfiltration from speakerless air-gapped computers via covert hard-drive noise ('DiskFiltration')," in *Proc. Eur. Symp. Res. Comput. Secur.* Cham, Switzerland: Springer, 2017, pp. 98–115.

[30] P. Krishnamurthy, F. Khorrami, R. Karri, D. Paul-Pena, and H. Salehghaffari, "Process-aware covert channels using physical instrumentation in cyber-physical systems," *IEEE Trans. Inf. Forensics Security*, vol. 13, no. 11, pp. 2761–2771, Nov. 2018.

[31] J. Loughry and D. A. Umphress, "Information leakage from optical emanations," *ACM Trans. Inf. Syst. Secur.*, vol. 5, no. 3, pp. 262–289, 2002.

[32] A. C. Lopes and D. F. Aranha, "Platform-agnostic low-intrusion optical data exfiltration," in *Proc. ICISSP*, 2017, pp. 474–480.

[33] M. Guri, B. Zadov, and Y. Elovici, *LED-it-GO: Leaking (A Lot of) Data from Air-Gapped Computers via the (Small) Hard Drive LED*. Cham, Switzerland: Springer, 2017, pp. 161–184, doi: 10.1007/978-3-319-60876-1_8.

[34] J. Loughry, "Optical TEMPEST," in *Proc. Int. Symp. Electromagn. Compat.*, Aug. 2018, pp. 172–177.

[35] D. Bak, P. Mazurek, and D. Oszutowska–Mazurek, "Optimization of demodulation for air–gap data transmission based on backlight modulation of screen," in *Computational Science* (Lecture Notes in Computer Science). Springer, 2019, pp. 71–80.

[36] Z. Zhou, W. Zhang, S. Li, and N. Yu, "Potential risk of IoT device supporting IR remote control," *Comput. Netw.*, vol. 148, pp. 307–317, Jan. 2019.

[37] Z. Zhou, W. Zhang, Z. Yang, and N. Yu, "Optical exfiltration of data via keyboard led status indicators to IP cameras," *IEEE Internet Things J.*, vol. 6, no. 2, pp. 1541–1550, Apr. 2019.

[38] M. Guri and D. Bykhovsky, "aIR-Jumper: Covert air-gap exfiltration/infiltration via security cameras & infrared (IR)," *Comput. Secur.*, vol. 82, pp. 15–29, May 2019.

[39] M. Guri, M. Monitz, Y. Mirski, and Y. Elovici, "BitWhisper: Covert signaling channel between air-gapped computers using thermal manipulations," in *Proc. IEEE 28th Comput. Secur. Found. Symp.*, Jul. 2015, pp. 276–289.

[40] P. Kocher, J. Jaffe, and B. Jun, "Differential power analysis," in *Proc. Annu. Int. Cryptol. Conf.* Berlin, Germany: Springer, 1999, pp. 388–397.

[41] M. A. Islam and S. Ren, "Ohm's law in data centers: A voltage side channel for timing power attacks," in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Toronto, QC, Canada, 2018, pp. 146–162.

[42] D. C. Cox and T. Clark, "Software-based data ex-filtration via simple power analysis," M.S. thesis, Dept. Comput. Sci., Naval Post Graduate School, Monterey, CA, USA, 2009.

[43] S. K. Khatamifard, L. Wang, A. Das, S. Kose, and U. R. Karpuzcu, "POWERT channels: A novel class of covert communicationexploiting power management vulnerabilities," in *Proc. IEEE Int. Symp. High Perform. Comput. Archit.*, Feb. 2019, pp. 291–303.

[44] D. Imhoff. (Jul. 2017). *Data Exfiltration From Air Gapped Systems Using Power Line Communication*. Accessed: Apr. 12, 2018. [Online]. Available: https://pushstack.wordpress.com/2017/07/24/data-exfiltration-from-air-gapped-systems-using-power-line-communication/

[45] K. H. Billings and T. Morey, *Switchmode Power Supply Handbook*. New York, NY, USA: McGraw-Hill, 2011.

[46] S. Ye, W. Eberle, and Y.-F. Liu, "A novel EMI filter design method for switching power supplies," *IEEE Trans. Power Electron.*, vol. 19, no. 6, pp. 1668–1678, Nov. 2004.

[47] F. Fiori and F. Musolino, "Comparison of IC conducted emission measurement methods," *IEEE Trans. Instrum. Meas.*, vol. 52, no. 3, pp. 839–845, Jun. 2003.

[48] *FCC Part 15B*. Accessed: Mar. 19, 2018. [Online]. Available: https://cdn-shop.adafruit.com/datasheets/SIM800_FCC_Part15.pdf

[49] D. Fung. (Apr. 2015). *Introduction to Conducted Emission*. Accessed: Mar. 19, 2018. [Online]. Available: http://www.ee.cityu.edu.hk/~emc/20150418P1.pdf

[50] M. Bellare, D. Kane, and P. Rogaway, "Big-key symmetric encryption: Resisting key exfiltration," in *Advances in Cryptology*, M. Robshaw and J. Katz, Eds. Berlin, Germany: Springer, 2016, pp. 373–402.

[51] *Industrial Defence in-Depth*. Kaspersky Lab. Accessed: Dec. 3, 2017. [Online]. Available: https://www.sans.org/summit-archives/file/summit-archive-1493412875.pdf

[52] *The Epic Turla (Snake/Uroburos) Attacks | Virus Definition | Kaspersky Lab*. Accessed: Dec. 3, 2017. [Online]. Available: https://www.kaspersky.com/resource-center/threats/epic-turla-snake-malware-attacks

[53] GReAT. (Jan. 2013). *Red October'Diplomatic Cyber Attacks Investigation*. Accessed: May 30, 2019. [Online]. Available: https://securelist.com/red-october-diplomatic-cyber-attacks-investigation/36740/

[54] *A Fanny Equation: I am Your Father, Stuxnet—Securelist*. Accessed: Dec. 3, 2017. [Online]. Available: https://securelist.com/a-fanny-equation-i-am-your-father-stuxnet/68787/

[55] *COTTONMOUTH-III: NSA Exploit of the Day–Schneier on Security*. Accessed: Dec. 3, 2017. [Online]. Available: https://www.schneier.com/blog/archives/2014/03/cottonmouth-iii.html

[56] J. Von Kistowski, H. Block, J. Beckett, C. Spradling, K.-D. Lange, and S. Kounev, "Variations in CPU power consumption," in *Proc. 7th ACM/SPEC Int. Conf. Perform. Eng.*, Mar. 2016, pp. 147–158.

[57] *Pthread_Mutex_Lock(3): Lock/Unlock Mutex—Linux Manual Page*. Accessed: Dec. 3, 2017. [Online]. Available: https://linux.die.net/man/3/pthread_mutex_lock

[58] *Sleep(3)–Linux Manual Page*. Accessed: Dec. 3, 2017. [Online]. Available: http://man7.org/linux/man-pages/man3/sleep.3.html

[59] J. Proakis and M. Salehi, *Digital Communications*, 5th ed. New York, NY, USA: McGraw-Hill, 2008.

[60] *Ecs1030-l72-Spec-Sparkfun-082411*. Accessed: Mar. 14, 2018. [Online]. Available: https://cdn.sparkfun.com/datasheets/Sensors/Current/ECS1030-L72-SPEC.pdf

[61] T. A. Papadopoulos, C. G. Kaloudas, A. I. Chrysochos, and G. K. Papagiannis, "Application of narrowband power-line communication in medium-voltage smart distribution grids," *IEEE Trans. Power Del.*, vol. 28, no. 2, pp. 981–988, Apr. 2013.

[62] A. Sendin, I. Peña, and P. Angueira, "Strategies for power line communications smart metering network deployment," *Energies*, vol. 7, no. 4, pp. 2377–2420, 2014.

[63] *Home | Libreoffice–Free Office Suite–Fun Project–Fantastic People*. Accessed: Jan. 11, 2018. [Online]. Available: https://www.libreoffice.org/

[64] *Official Download of VLC Media Player, the Best Open Source Player–VideoLAN*. Accessed: Jan. 11, 2018. [Online]. Available: https://www.videolan.org/vlc/index.html
[65] *Rsync(1)–Linux Manual Page*. Accessed on Jan. 14, 2018. [Online]. Available: https://linux.die.net/man/1/rsync
[66] *Ubuntu Manpage: Matho-Primes—Generate Consecutive Prime Numbers*. Accessed: Jan. 11, 2018. [Online]. Available: http://manpages.ubuntu.com/manpages/zesty/man1/matho-primes.1.html
[67] Curtis Industries. *Tempest Line Filters*. Accessed: Feb. 3, 2019. [Online]. Available: https://www.curtisind.com/products/military/
[68] I. Agadakos *et al.*, "Jumping the air gap: Modeling cyber-physical attack paths in the Internet-of-Things," in *Proc. Workshop Cyber-Phys. Syst. Secur. PrivaCy*, New York, NY, USA, 2017, pp. 37–48, doi: 10.1145/3140241.3140252.
[69] B. Carrara and C. Adams, "A survey and taxonomy aimed at the detection and measurement of covert channels," in *Proc. 4th ACM Workshop Inf. Hiding Multimedia Secur.*, 2016, pp. 115–126.
[70] A. A. Cárdenas, S. Amin, Z.-S. Lin, Y.-L. Huang, C.-Y. Huang, and S. Sastry, "Attacks against process control systems: Risk assessment, detection, and response," in *Proc. 6th ACM Symp. Inf., Comput. Commun. Secur.*, Mar. 2011, pp. 355–366.
[71] M. S. Racine, J. D. Parham, and M. Rashid, "An overview of uninterruptible power supplies," in *Proc. 37th Annu. North Amer. Power Symp.*, Oct. 2005, pp. 159–164.
[72] S. Zander, G. Armitage, and P. Branch, "A survey of covert channels and countermeasures in computer network protocols," *IEEE Commun. Surveys Tuts.*, vol. 9, no. 3, pp. 44–57, 3rd Quart., 2007.
[73] B. A. Bash, D. Goeckel, D. Towsley, and S. Guha, "Hiding information in noise: Fundamental limits of covert wireless communication," *IEEE Commun. Mag.*, vol. 53, no. 12, pp. 26–31, Dec. 2015.
[74] Sense. *Sense Home Energy Monitor*. Accessed: Mar. 3, 2019. [Online]. Available: https://sense.com/
[75] M. Migliardi, A. Merlo, and L. Caviglione, "A survey of green, energy-aware security and some of its recent developments in networking and mobile computing," in *Proc. 8th Int. Conf. Innov. Mobile Internet Services Ubiquitous Comput.*, Jul. 2014, pp. 241–246.
[76] Q. Do, B. Martini, and K.-K. R. Choo, "Cyber-physical systems information gathering: A smart home case study," *Comput. Netw.*, vol. 138, pp. 1–12, Jun. 2018.
[77] 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.
[78] E. Weiss, D. Bykhovsky, and S. Arnon, "Symbol error rate model for communication using femtosecond pulses for space applications," *IEEE Photon. Technol. Lett.*, vol. 28, no. 12, pp. 1286–1289, Jun. 15, 2016.