Experimental Demonstration of Electromagnetic Information Leakage from Modern Processor-Memory Systems

Alenka Zajić, *Senior Member, IEEE* and Milos Prvulovic, *Senior Member, IEEE*

***Abstract*— This paper shows that electromagnetic (EM) information leakage from modern laptops and desktops (with no peripherals attached) is indeed possible and is relatively easy to achieve. The experiments are performed on three laptop systems and one desktop system with different processors (Intel Centrino, Core 2, Core i7, and AMD Turion), and show that both active (program deliberately tries to cause emanations at a particular frequency) and passive (emanations at different frequencies happen as a result of system activity) EM side-channel attacks are possible on all the systems we tested. Furthermore, this paper shows that EM information leakage can reliably be received at distances that vary from tens of centimeters to several meters including the signals that have propagated through cubicle or structural walls. Finally, this paper shows how activity levels and data values used in accessing different parts of the memory subsystem (off-chip memory and each level of on-chip caches) affect the transmission distance.**

***Index Terms*— electromagnetic emanation security, electromagnetic information leakage, information security, security of modern professors, TEMPEST, side-channel attack, covert-channel attack.**

I. Introduction

Although it is well-known that electronic circuits used in modern computer systems are sources of electromagnetic emanations (which can be confirmed by placing any commercial radio receiver very close to a working computer system), little is known about whether these emanations exhibit any data-dependent behavior. The existence of side-channel electromagnetic (EM) radiation and the potential risk it poses on the computer security was reported in the open literature as early as 1966 [1], but without technical details on specific risks, eavesdropping techniques, or how to prevent such attacks. One of the first unclassified technical reports on analysis of the security risks of EM emanations from computer monitors have been reported in [2], [3]. Recently, it was shown that the EM emanations from computer keyboards also pose security risks [4]. To address these risks, several evaluation methods and countermeasure techniques have been proposed [5]-[9].

Research interest in compromising EM emanations further increased with the mass-market introduction of smart cards, i.e., cards with embedded microcontrollers. Typical smartcards have relatively simple 8-bit microcontrollers operating at low frequencies (<300 MHz), a few hundred bytes of RAM, and 5-30 kilobytes of ROM memory. Systematic investigation of leakage of compromising information via direct (unmodulated) and indirect (unintentionally modulated) EM emanations from smartcards has been reported in [10], [11]. Direct EM emanations are a result of intentional current flows within circuits, while indirect EM emanations are a result of small coupling among chip circuits. Agarwal et al. [10] and Gandolfi et al. [11] found that EM emanations consist of multiple compromising signals, each one leaking somewhat different information. These experiments also confirmed that each of these EM signals can individually contain enough leaked information to break cryptographic implementations even in the presence of significant countermeasures against other side-channels (e.g. power analysis). Finally, several countermeasures for protecting video displays and smartcards from EM leakage have been proposed [12]-[19], including low-cost shielding techniques (e.g. metal foil), use of asynchronous circuits, and changing the layout of circuitry.

Unfortunately, little work has been openly published that investigates EM emanations from more powerful processors and systems, such as server, desktop, laptop, and smartphone systems. These systems may be much more vulnerable than smartcards for many reasons, including 1) having more complex circuitry that can produce unintended coupling and modulation effects, 2) components (e.g. processors) in these systems have more external connections (e.g. pins) that can serve as unintended antennas for these signals to get out, 3) these systems expend far more power and use higher voltage levels (for higher performance), which may increase the range from which such leakage can be observed, 4) these processors use many more performance-enhancing mechanisms that can each potentially leak information, and 5) these systems operate at higher frequencies and process more data per second, which can potentially increase the rate at which information is leaked. Although there has been significant research and applied work on EM interference/compatibility (EMI/EMC) [20], [21], which is also concerned with EM emanations, that work is mostly focused on interference a computer system or its components can cause in other devices and in radio communications. Hence, the goal of this paper is to provide some insights into EM covert- and side-channel emanations from modern systems.

This paper shows that electromagnetic (EM) emanations from modern laptops and desktops (with no peripherals attached) do carry inforation about program activity. While previous work on EM emanations focuses mainly on deducing cryptography keys, we use a broader notion of a “side channel”, which includes e.g. passwords and other possibly sensitive data in regular programs. Other work has already used the term “side channel” in such a broader sense [22]-[24],

and it has also been shown that information about program activity, such as memory access patterns, can enable other attacks [24], compromise cryptographic keys [25], or reveal potentially sensitive information about the user's activity [23].

Furthermore, this paper shows that information embedded in the leaked EM signals can reliably be received at distances that vary from tens of centimeters to several meters including the signals that have propagated through cubicle or structural walls. Finally, this paper shows how activity levels and data values used in accessing different parts of the memory subsystem (off-chip memory and each level of on-chip caches) affect the transmission distance. Our results include measurements from different types of real systems (laptop/desktops), and several processor manufacturers and models.

The rest of the paper is organized as follows. Section II describes our experimental setup, Section III presents our measured results, Section IV discusses potential defenses against EM covert- and side-channel attacks, and Section V summarizes our conclusions.

II. Experimental Setup

Our experimental environment was chosen to resemble a possible attack environment - we carried out our experiments in a student office within the Klaus Advanced Computing Building at Georgia Tech, located in the heart of a major metropolitan area (Atlanta, Ga, USA), using a compact, low-cost, commercially available receiver.

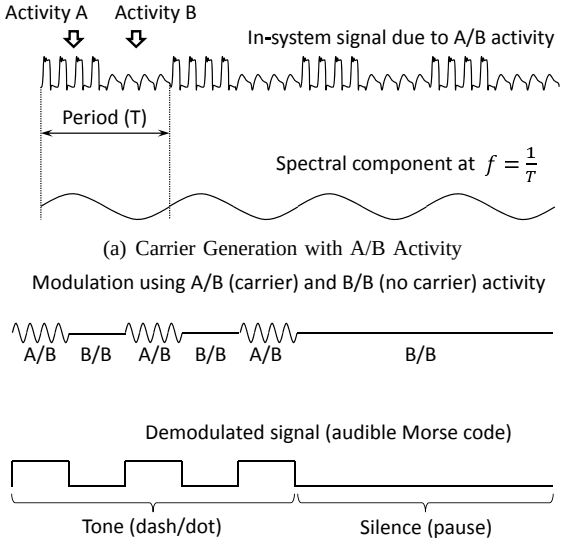
The goal of this paper is to demonstrate *program-activity-dependent* EM emanations from modern systems, identify how far they propagate, and understand how these emanations are related to architectural components in the system. For this, we need an experimental setup that will not only receive EM emanations in an environment similar to one where attacks might occur, but also conclusively establish that these emanations leak information about program activity that exercises a specific part of the computer architecture. This requirement for *activity-dependent* emanations poses a significant challenge - the EM emanations cover an enormous range in the radio-frequency (RF) spectrum, but it is a very difficult task to find which frequencies have any information about program activity embedded (modulated) in the signal. As a result, even when activity-revealing signals are present in a particular frequency range, commercial receivers may not be able to demodulate such signals in a way that provides us with evidence that such emanations are activity-dependent.

In light of this, we chose not to carry out a potentially inconclusive search for the frequencies and modulations that unintentionally carry data. Instead, we deliberately cause activity within the system that, if such activity produces emanations at all, should produce emanations at a frequency of our choice and with a modulation of our choice. In other words, instead of trying to find passive data-carrying emanations, we attempt to deliberately cause data-carrying emanations that, if they do occur, will be easy to recognize as such. Later in the paper, we will show that passive emanations that leak data also occur.

The method we use to produce these controllable emanations is to create repetitive variations in activity. We choose T, the period (duration) of each repetition, two types of activity (A and B), and write a small software code (i.e. microbenchmark) that in each period does activity A in the first half and B in the second half of the period. The intuition behind this is that, if activity A and activity B result in non-identical EM fields around the processor or the system, repetition of this A-then-B pattern will create oscillations (with period T) in this EM field, i.e. it will result in a "carrier" RF signal at frequency $1/T$. The period T will be selected to correspond to a specific frequency, e.g. to produce a radio signal at 1 MHz (near the middle of the commercial AM band) we should set $T = 1\mu s$. This carrier-generation approach is illustrated in Figure 1(a).

Unfortunately, transmission of only a carrier signal is not sufficient to provide proof of *data-carrying* transmissions - as indicated earlier, all systems we have tested produce *some* signal at every frequency we tried to receive, so when the receiver indicates that a signal is present at the intended frequency, it might be that this signal is completely unrelated to our microbenchmark. Therefore, we modified our microbenchmark to *modulate* an audible signal into the transmitted carrier signal, using amplitude modulation that can be demodulated by simple radio receiver. In our setup, we use an inexpensive (<100 USD) handheld radio receiver, Tecsun PL-660, that supports the commercial AM range from 520 kHz to $1,610$ kHz and the long-wave (100 kHz to 520 kHz) band, as well as the shortwave and commercial FM radio bands.

Now, our AM modulation is achieved by inserting intervals during which only activity B is performed in both half-periods - any carrier signal produced by differences between A and B should be absent when only B is used, resulting in the simplest form of AM modulation (on-off keying).



(a) Carrier Generation with A/B Activity

(b) Modulation of the Carrier Signal

Fig. 1. Our microbenchmark attempts to induce emanations at a specific radio frequency by alternating half-periods of A and B activity (left) and then modulating an audible Morse code signal into this carrier signal (right).

Even an idle computer system produces RF signals that, after AM demodulation, result in clicking, whining, and other sounds. To distinguish our signals from those, we modulate our

transmitted signal the A5 note (880 Hz), and turn this tone on and off to transmit Morse code for "All your data is belong to us." This modulation approach is illustrated in Figure 1(b). Because it is virtually impossible for such a message to be accidentally produced by some other mechanism, we base our "reception distance" results on successful reception of this message from a given distance.

Any two types of activity that can happen under program control can be used for A and B, e.g. if activity A is "memory accesses that cause level-three (L3) cache misses" and B is "empty-loop", successful reception of the resulting signal indicates that EM emanations can allow the attacker to tell the difference between L3 misses and "do-nothing" activity. From this presence or absence of L3 misses, the attacker may potentially glean information about the data in the program. Successful reception of the signal in this experiment would allow an even stronger conclusion to be drawn about covert-channel transmissions - a malicious program can clandestinely transmit data over the EM side-channel by creating L3 misses.

There can be many kinds of activity in the computer system, so we had to select the A and B activity carefully. For activity A, we chose 32-bit loads (memory reads) from a pseudo-random element of an array filled with all-one values (every 32-bit word is 0xFFFFFFFF). We chose memory loads because they happen frequently in real programs and are relatively easy to control - by varying the size of the array, the same microbenchmark can produce L3 misses, level-two (L2) misses (but L3 hits), L1 misses (but L2 hits), and level-one (L1) hits, so we can study how these different levels of the memory hierarchy affect the RF emanations. Activity B in most of our experiments is a simple empty loop, so any reception of Morse-code signals at our receiver is due to differences between what happens when activity in a given part of the memory hierarchy is present and when it is absent. We also perform some *value-based* experiments (see Section III-E), where activity B is identical to activity A, except that is accesses memory that holds different values than that accessed in A. Any Morse-code signals received in these value-based experiments are caused by actual data values being fetched from the memory subsystem.

Finally, please note that, as indicated earlier, the sound and the Morse-code are used in our experiments so that we can recognize our signal when it is received by a commercial radio receiver. Other types of signals (e.g. outside of the normal hearing range) can be embedded into the carrier signal, and other modulations (e.g. frequency modulation or even some non-standard modulation) can just as easily be used to create a truly covert transmission that can still be received by a customized receiver/demodulator.

III. Experimental Results

A. Reception at Intended RF Frequencies

In our initial set of experiments, we run our memory-activity microbenchmark on a latest-generation laptop with an Intel i7-2620M processor (introduced in February 2011, 32 nm Sandy Bridge microarchitecture), which has two cores, 2 threads per core, runs at 2.7 GHz, and has a 4 MB L3 cache shared by both cores. Each core also has a 256 kB L2 cache and a 32 kB L1 data cache. Note that the microbenchmark we use is a single-threaded 32-bit application running in user mode under a 64-bit version of Windows 7. There are no other compute-intensive or memory-intensive applications running while the microbenchmark is executing, but other than that we do not do anything to affect the normal operation of the system, i.e. the system is still performing its normal I/O, memory, and processor management, which can interfere with the microbenchmarks ability to create "clean" activity patterns with a stable frequency. We will discuss the effect of this system interference as the need arises, but the reader should keep in mind that the reception distances we report are measured in the presence of some noise and timing instability caused by normal system activity.

Our first experiment is to determine the reception distance when we vary the footprint of the microbenchmark's memory accesses and the direction from which the emanations are being received. The microbenchmark was set to self-adjust so that its A/B period corresponds to a frequency of 200 kHz, and the receiver is tuned to receive at the same (200 kHz) frequency. The results of this experiment are shown in Figure 2.

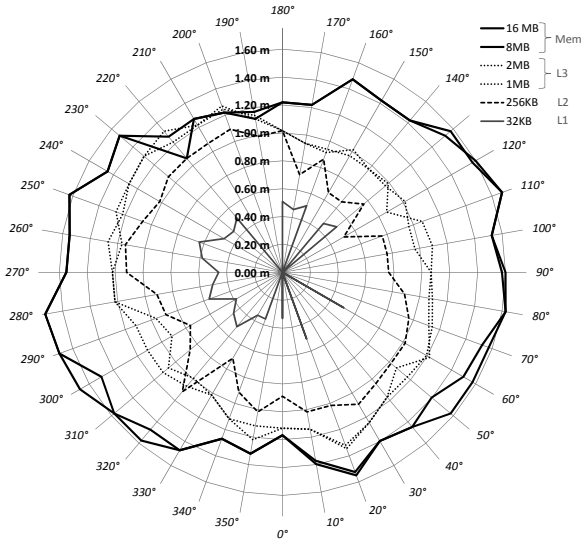


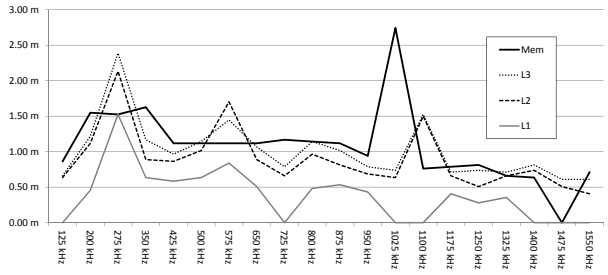
Fig. 2. Data reception distance from the i7-based laptop, for different directions and activity, when the radio receiver is tuned to the same frequency at which the microbenchmark is trying to create emanations.

We observe that the reception distance is significantly affected by different memory activity - L3 misses (off-chip main memory accesses) result in data reception from up to 1.73 m, L3 hits (L2 misses) result in reception from up to 1.35 m, L2 hits (L1 misses) result in reception from up to 1.14 m, and L1 hits result in reception from only up to 0.64 m. Several considerations can help explain this result. First, accesses to lower parts of the memory hierarchy (e.g. memory and larger caches) include misses in higher parts of the hierarchy (e.g. L1 cache), so accesses to lower parts of the memory hierarchy can be expected to transmit signals to at least the same distance as higher parts of the memory hierarchy. Additionally, lower parts of the memory hierarchy consume more energy per access, so they create larger currents

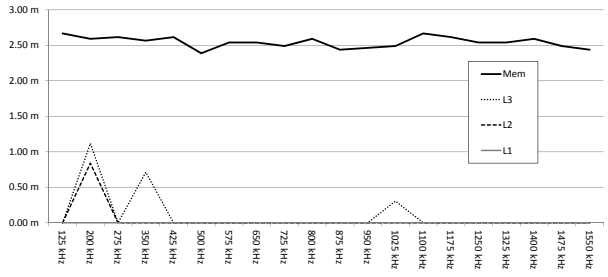
and voltage swings in (longer) power supply lines that may also act as a transmission mechanism.

We also observe that the reception distance changes somewhat with the direction. Interestingly, main memory activity results in longer-distance reception on both sides than it does at the front and back of this laptop, L3 and L2 activity has longer-distance reception in back-right and front-left directions, and L1 activity tends to only create discernible signals on the right side of the laptop. One possible explanation for this lies in the physical location of the processor and memory, and in the shape and distribution of metal sheets (case) in this particular laptop. The main memory is located near the middle of the case, the processor is on the right side and toward the back, and metal sheets almost entirely encase the electronics of the laptop with only some (narrow) openings in the left-back corner (air exhaust from the CPU cooling fan) and along the right side (where the CD/DVD drive is).

Finally, we observe that the metal cladding in this laptop, in spite of nearly entirely enveloping the processor and memory, only suppresses the emanations somewhat but does not prevent them. This is not a surprising result, because thin metal enclosures are known to provide poor shielding at frequencies below a few MHz [20], [21]. While metal cases are present in all of our experiements, for additional insight we assess the emanation-suppression effect of a metal case in Section III-C.



(a) Reception at "transmit" frequency



(b) Reception at 1024 kHz

Fig. 3. Data reception distance (vertical axis) from the i7-based laptop from the 270°direction (right side), with the microbenchmark self-tuned for different frequencies (horizontal axis). Reception distance is shown at the microbenchmark's frequency (upper) and at 1024 kHz (lower).

To confirm that we can tune our microbenchmark to any frequency, we performed an experiment in which we vary the transmission/reception frequency from 125 kHz to 1550 kHz with a 75 kHz step, with the receiver in the 270°direction, i.e. pointed at the right side of the laptop. The results of this "frequency sweep" experiment are shown in Figure 3(a). From these results we make three observations. First, data reception can be achieved at nearly every frequency in this range, which has an important implication on security of real applications - the frequency of "transmission" depends on the period of looping behavior in the application, and this shows that data-carrying emanations may be present for any specific period for repetitive behavior, i.e. emanations do not only occur at specific "resonant" frequencies. Second, in these results, there is a general trend toward shorter reception distances as the frequency increases. We have not yet been able to experimentally determine a specific reason for this, but two likely hypotheses are that it is related to the antenna construction in our receiver[1] or to the physical dimensions of the laptop and its components. Our final observation from these "frequency sweep" results is that reception distance for memory-activity experiments sharply spikes at 1024 kHz, while in cache-activity experiments the reception distance sharply spikes at 275 kHz (with smaller spikes at 575 kHz and 1100 kHz). The explanation for these spikes and for the results in Figure 3(b) is in Section III-B.

B. Reception at Other (Unintended) Frequencies

In the course of our prior experiments with the i7-based laptop, we noted that memory activity causes much stronger signals at 1024 kHz than at other frequencies. We investigated this and discovered that, when the microbenchmark is set to cause emanations at a particular frequency, the same modulated signal (Morse code with A5 note) can also be received at several other frequencies. Most of those other frequencies were multiples (harmonics) of the intended frequency, where the presence of these signals is easy to explain: our A/B activity pattern causes periodic "carrier" activity that is not purely sinusoidal, so it creates signals at frequencies that are harmonics of the intended one. For example, when our A/B activity is timed for 200 kHz, it results in reception not only at 200 kHz but also at 400 kHz, 600 kHz, etc. We note that the reception of harmonics may be due to the receiver or its ferrite antenna, so their reception is not conclusive evidence that they are actually transmitted.

The signal at 1024 kHz was *stronger* than the one at 200 kHz, even though the microbenchmark was set for 200 kHz. To investigate, we performed measurements identical to those in Section III-A, but this time with the receiver always set to receive at 1024 kHz, regardless of what frequency the microbenchmark is self-tuned for.

Figure 4 shows these results when the microbenchmark is self-tuned for 200 kHz. The maximum reception distance for memory activity emanations is now 2.64 m, which is 50% farther than when the receiver is tuned to the intended transmission frequency. Interestingly, the reception distances for L3 and L2 activity are not similarly increased when receiving at 1024 kHz (they are very similar to what we received at 200 kHz), and there was no discernible reception of signals from L1 activity. For on-chip activity (L1, L2, and L3) the reception distances are consistent with this being a harmonic of the "transmission" frequency. For off-chip activity, however, there appears to be an additional mechanism that results in

[1]The same ferrite antenna is used for signals in this entire frequency range, and a ferrite antenna tends to work better for lower frequencies in its range

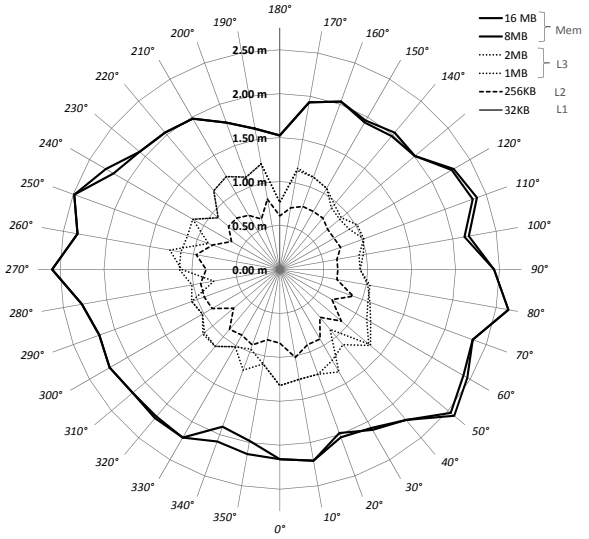


Fig. 4. Data reception distance from the i7-based laptop at 1024 kHz, for different directions and activity, while the microbenchmark is trying to create emanations at 200 kHz frequency.

significantly stronger emanations. Adopting the terminology from [19], the Source (processor activity in this case) is the same for both 200 kHz and 1024 kHz signals, but the Path and/or Antenna components for the 1024 kHz signal result in much stronger EM emanations.

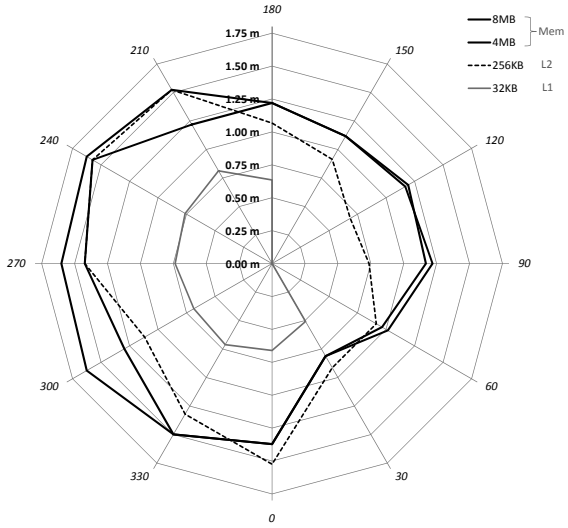
We performed additional experiments to investigate this, and found that memory activity results in audible signals at 1024 kHz, both in our microbenchmark and in other applications. Figure 3(b) shows the reception distance at 1024 kHz with our microbenchmark, while varying the frequency for which the microbenchmark is set. For memory activity, this distance stays around 2.5 m, while for on-chip activity this distance depends on the microbenchmark's frequency.

We also received 1024 kHz signals without our microbenchmark, while running other (real) applications, and observed that application start-up and other phases of intense memory activity can easily be heard as distinct clicking and scratching sounds when the AM receiver is tuned to 1024 kHz and placed at a distance of about 2.5 m from the system.

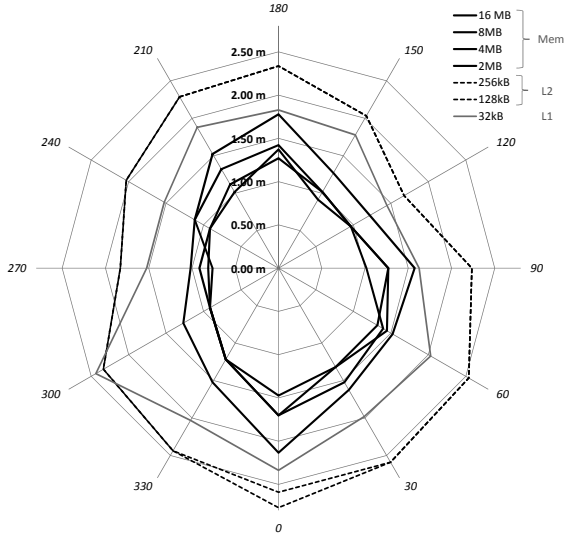
We observe similar unintended emanations at 275 kHz (and several of its harmonic frequencies) for on-chip activity. We confirm this by performing another "frequency sweep" experiment, but with the receiver tuned to 275 kHz. In this experiment, we observe reception from up to 0.42 m for memory-activity, and stronger emanations for on-chip activity - reception from 2.3 m for L3, 2 m for L2, and 1.5 m for L1 activity. These reception distances vary very little when we change the frequency at which the microbenchmark is set to cause emanations. We conclude that these 275 kHz emanations are likely caused by on-chip caches (likely L1 and L2 caches). To corroborate this conclusion, we note that, in the course of this experiment, we occasionally (every few seconds) heard short "whistle" sounds on top of the microbenchmark-induced output from our receiver, and we found that the same "whistle" sounds occur while running other applications (without our microbenchmark). We investigated this by monitoring system activity and found that these sounds coincide with process migration, i.e. moving a process (e.g. our microbenchmark) from one core to the other. Because the signals we are receiving at 275 kHz appear to be related to on-chip cache activity, we believe that the "whistle" sound is caused by a burst of L1/L2 cache misses that occurs when a process is moved to another core.

C. Experiments with Other Systems

All results we have reported up to this point were collected on a single laptop system. To allow us to draw more general conclusions, we performed similar experiments on two other laptop systems and one workstation system. On each of these systems, our microbenchmark was self-tuned to create emanations with a carrier frequency of 200 kHz. Figure 5(a) shows the results collected for a laptop based on the Intel Core Duo T2600 mobile processor (introduced in January 2006, 65 nm Yonah microarchitecture) with two cores operating at



(a) Core Duo Laptop



(b) Turion X2 Ultra Laptop

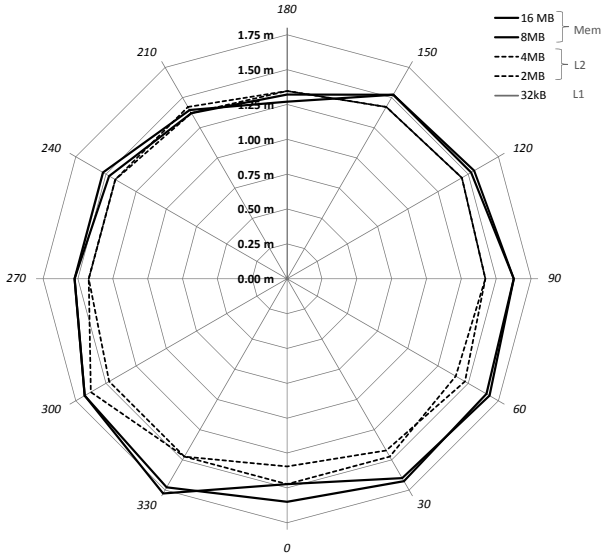
Fig. 5. Data reception distance at 200 kHz for two additional laptop systems.

2.16 GHz, with a shared 2 MB L2 cache and per-core 32 kB L1 data caches. In this system we observe a slightly more biased radiation pattern - reception is significantly stronger from the left side, which is where the CPU and the main memory reside. We also observe more pronounced emanations from L1 activity - in this system, L1 activity transmits data to a distance of up to 0.81 m. Finally, we observe that L2 (last-level cache) and memory activity experiments result in emanations of similar strength, with the same caveat as in Section III-A - in a given interval more accesses are performed with L2 activity alone than with main memory activity.

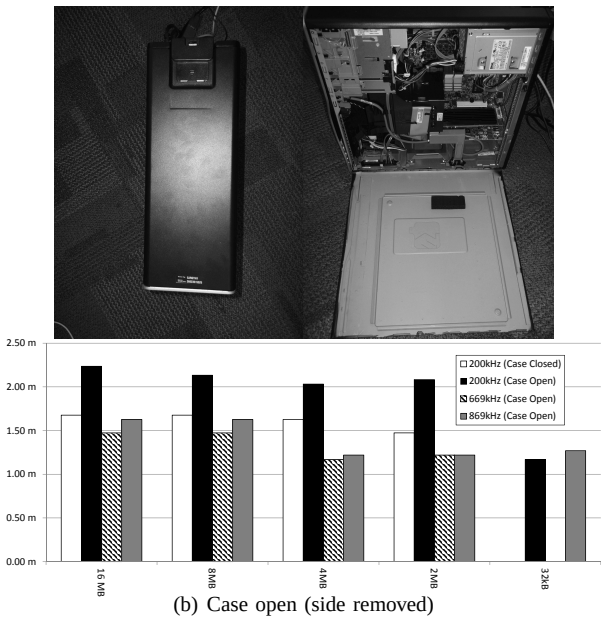
Figure 5(b) shows the results collected for a laptop based on the AMD Turion X2 Ultra ZM-80 mobile processor (Introduced in June 2008, 65 nm Lion microarchitecture), with two cores operating at 2.10 GHz, each with a 1 MB L2 cache and a 64 kB L2 data cache. For this system, we observe much stronger emanations from L1 cache activity - the data reception distance for L1 activity is 2.44 m. The emanations from L2 (last-level cache) activity also propagate farther than in prior experiments - the data reception distance for these emanations is up to 2.77 m. Longer reception distances in this laptop might be due to the geometry of the system (where the processor is, distribution of metal in the laptop's case, etc.).

Finally, Figure 6(a) shows the results collected for a workstation based on the Intel Core 2 Extreme X9650 desktop processor (introduced in November 2007, 45 nm Yorkfield microarchitecture), with four cores operating at 3 GHz, with a 6 MB L2 cache shared by each pair of cores, and a 32 kB data cache in each core. In this system we observe a radiating pattern that favors the front of the system. This is not surprising, considering that the entire system is nearly completely encased in metal, and that the only significant openings are two empty drive bays located at the front of the case. In fact, we expected not to get any data reception from this system, especially on the sides where there is a two-layer seamless metal sheet (see photos in Figure 6(b)) between the system's components and the receiver.

This system presented us with a nearly ideal setup to investigate the effect of the metal case - unlike laptop systems, where removal of the metal cladding would require complete disassembly of a tightly-packed system, the case of this desktop system can be opened easily by unlatching and removing the side of the case. We then set our microbenchmark to cause emanations at 200 kHz, and recorded emanations from the 300°direction (facing the open side, but towards the front of the system where the processor is). We set the receiver at 200 kHz (to compare against the results collected with the case closed), and also at 669 kHz and 869 kHz (where we found strong signals regardless of which frequency the microbenchmark was set for). The results of this experiment are show in the chart in Figure 6(b). We observe that the metal case does suppress emanations, but that effect is limited - the seamless metal sheet (side of the case) reduces the reception distance (at least at the frequencies that we measured) by only 20-30%. The only exception is the L1 cache activity, where reception went from 1.17 m when the case is open to no reception when the case is closed. As in all other experiments, we note that the "zero reception distance" means that the



(a) Receiver at 200 kHz



(b) Case open (side removed)

Fig. 6. Data reception distance for a desktop system based on the Core 2 Extreme processor with the microbenchmark self-tuned for 200 kHz. We show results for (a) 200 kHz receiver frequency from different directions and (b) for three receiver frequencies (without changing the microbenchmark's frequency) with the open case (see photos above chart).

receiver did not output "readable" Morse code. However, note that a weak Morse code signal may be present, but is buried in this noise, very close to the system (up to about 0.4 m) the receiver outputs very loud noise at nearly all frequencies. Still, this result indicates that, at least in this system, the presence of the metal case does prevent L1 activity from being used to create RF side-channel transmissions, at least in the frequency range that our LW/AM receiver can receive.

D. Effect of Walls

Having found that a metal case only has a limited effect on the reception distance, we performed several experiments to determine how the data propagation distance is affected by

walls. Because these experiments required significant portability, they were performed with the same laptop used in Sections III-A and III-B. We have found that glass walls (about 20 cm thick, such as those often used in "showcase" server rooms) and cubicle partitions create no difference (within our measurement error) on reception distance - with glass or cubicle walls, we observe reception distance that vary less than 5 cm from those recorded without any obstructions. We also performed an experiment through a structural wall that has vertical steel beams (about 0.1 m on each side), that are separated by about 0.2 m of insulation-filled space, and that are finished with drywall (plaster board) on both sides. The entire wall is about 0.15 m thick. We found that this wall reduces the reception distance by 0.5-0.6 m. For example, signals that had a reception distance of 2.8 m and 1.6 m, the reception distance through this wall was 2.06 m and 1.12 m, respectively.

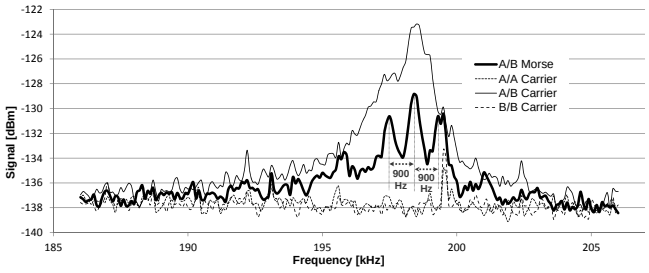
E. Signals Created by Differences in Data Values Only

In all our previous experiments, the signals were created by differences in the type of activity - activity "A" consisted of repeated memory accesses, and activity "B" consisted of an empty loop. Therefore, those experiments only prove that side-channel transmission of data is possible if that data affects the pattern of memory accesses, either in terms of time (whether accesses happen, how often, etc.) or in terms of which level of the memory subsystem they use. Encouraged by relatively large reception distances that we measured in those experiments, we also performed experiments where activity B is identical to activity A in terms of the type and pattern of accesses, and only differs in data values that are being loaded. In these experiments, activity A consists of a loop that loads a pseudo-random element from an array populated by values whose bits are all ones (0xFFFFFFFF), and activity B consists of executing the same loop (the same static instructions, to avoid any differences due to compiler optimizations or fetching instructions from different places) with the same element indices, but using a separate array that is populated by zero values. Both arrays are identically aligned, their virtual addresses differ in only one bit. Also,we use array sizes that ensure that both arrays together fit in the desired level of the memory hierarchy - e.g. to create hits in a 32 kB L1 cache, each of these these two arrays is 8 kB in size.

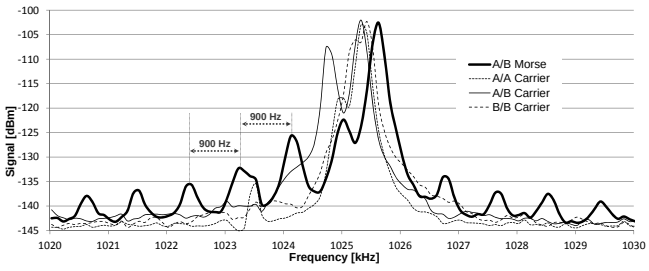
The results of this experiment are largely negative - on the Core i7 laptop (the only system we measured this way), we were not able to receive signals "transmitted" by such a value-based microbenchmark, except in one case - the 275 kHz parasitic emanations from on-chip activity. At 275 kHz, we were able to get clear reception from distances up to 0.3 m when the microbenchmark was set to generate L1 activity, and up to 0.5 m with L2 or L3 activity, regardless of the actual frequency that the microbenchmark was self-tuned to use. Although this is the only experiment (so far) in which we found that actual values create measurable transmissions, the implications for system security from side-channel attacks are significant - when the same value is used in a repetitive fashion, attackers may be able to wirelessly extract at least some information about that value (in our experiments, whether the value has all zeroes or all ones). Furthermore, this reception of values is possible even when the accesses to these values occur entirely in on-chip caches (even L1 hits), and from distances that permit at least some realistic attacks, e.g. hiding a receiver under a coffee-shop table, placing a briefcase next to the target system during a meeting, etc.

F. Validation in a Controlled Setting

The experiments reported in this paper use low-cost receivers, an unconventional metric for signal strength, and an uncontrolled experimental setting. While such experiments are more representative of the circumstances under which actual attacks might occur, we performed additional experiments to confirm that the signals we receive are present, and that they are "modulated" by our microbenchmark's activity. These additional experiments place the i7-based laptop from Sections III-A and III-B in a shielded chamber, together with an AOR LA-390 Loop Antenna placed 0.5 m away in the 270°direction. The antenna is conected to an Agilent N9020A MXA Signal Analyzer, located outside the shielded chamber.



(a) Spectrum at 200 kHz (Intended)



(b) Spectrum at 1025 kHz (Unintended)

Fig. 7. Received spectra for the i7-based laptop when the microbenchmark is trying to create emanations at 200 kHz

In Figure 7(a), the "A/B Morse" spectrum corresponds to experiments from Section III-A, and closely matches what would be expected in a signal where the carrier is modulated using an 880 Hz tone. Note that the actual carrier frequency is 198.5 kHz, not the 200 kHz that the microbenchmark is trying to achieve. This can be expected - the duration of each half-perior must be a whole number of iterations in our A or B activity loop, so the program cannot precisely match the target frequency. We also show the spectrum for just the A/B Carrier signal, without any attempt at modulation, which shows a strong carrier-like signal at 198.5 kHz. Finally, we also show the spectra for our carrier generation code when the same activity is used in both half-periods (A/A Carrier and B/B Carrier), and observe that such activity does not create

a carrier signal. This confirms that the signals we observe in Section III-A are indeed caused by by microbenchmark described in Section II.

In Figure 7(b) we show the spectrum around 1025 kHz for the exact same experiments (including "carrier generation" at 200 kHz). This corresponds to signals reported in Section III-B. Here we observe a strong signal around 1025.5 kHz regardless of program activity. However, the A/B Carrier activity results in an additional signal about 600 Hz below the "main" one, and the A/B Morse activity additionally results in numerous ripple signals at 900 Hz intervals. This does not directly match the spectrum for traditional AM modulation, but the "main" frequency and the first "ripple" on each side, when AM-demodulated, do result in audible Morse-code signals.

IV. Discussion

The results from these initial experiments show that EM covert-channel attacks are possible on each of the four systems we used in our measurements. In each system, we were able to generate (purely in software, by manipulating the timing of memory access activity) modulated RF signals that were successfully received at distances that enable many realistic attack scenarios. A scenario that we have conclusively proven to be possible is intentional covert transmission of data from a program, using only processor (or memory) activity patterns that are unlikely to raise suspicious even under close scrutiny. Such attacks can include, for example, injection of transmission code into a new version of popular software.

Our experiments also indicate that attacks are likely to be possible without any injection of code - transmission will occur "naturally" with most repetitive behavior, wherever the pattern of behavior varies depending on the data. This may likely affect all applications, because loops with if-then-else statements that depend on the data are often found in most code. Furthermore, we found that data-carrying emanations can in some cases be caused even without time-varying or activity-varying behavior - changes in data values alone can result in emanations that reveal at least some information about these values.

Several defensive approaches against EM emanation attacks are possible. Metal shielding has often been used to protect smartcards and other devices for which RF emanation attacks have been reported. However, for laptop and desktop system shielding provides only limited benefits - these systems already have substantial metal shielding and it provides only a limited benefit in terms of reception distance. A major increase in shielding is likely to detract from functionality, especially for laptops (where weight is a major factor in functionality). Another type of defense would be to disrupt the repetitive patterns of behavior, e.g. by introducing random delays or clock frequency variation. This defense poses significant performance challenges, especially if the introduced delays or clock variation is large enough to be effective - note that timing variation spreads the power of emanated signals over a wider frequency spectrum, and this spread needs to be large enough to "bury" the signal into the noise deeply enough that it cannot easily be recovered using statistical approaches (averaging, correlation, etc.). A complementary defense strategy would be to prevent repetitive behavior from being data-dependent, e.g. by preventing any data-dependent timing variation. This approach is also likely to carry a performance penalty - e.g. the faster path in an if-then-else hammock would need a delay to match the timing of the slower path. We plan to investigate these and other defensive approaches in our future work, together with experimental approaches that will allow us to measure their benefit.

V. Conclusions

To the best of our knowledge, this paper is the first to show that electromagnetic (EM) information leakage from modern laptops and desktops (with no peripherals attached) is indeed possible and is relatively easy to achieve. The experiments were performed on three laptop systems and one desktop system with different processors (Intel Centrino, Core 2, Core i7, and AMD Turion), and show that both active (program deliberately tries to cause emanations at a particular frequency) and passive (emanations at different frequencies happen as a result of system activity) EM side-channel attacks are possible on all the systems we tested. Furthermore, this paper showed that EM information leakage can reliably be received at distances that vary from tens of centimeters to several meters including the signals that have propagated through cubicle or structural walls. Finally, this paper showed how activity levels and data values used in accessing different parts of the memory subsystem (off-chip memory and each level of on-chip caches) affect the transmission distance.

References

[1] H. J. Highland. Electromagnetic radiation revisited. *Computers and Security*, pp. 85–93, 1986.

[2] W. van Eck. Electromagnetic radiation from video display units: an eavesdropping risk? *Computers and Security*, pp. 269–286, 1985.

[3] M. G. Khun, Compromising emanations: eavesdropping risks of computer displays. *The complete unofficial TEMPEST web page: http://www.eskimo.com/ joelm/tempest.html*, 2003.

[4] M. Vuagnoux and S. Pasini, "An improed technique to discover compromising electronic emanations," *Proc. IEEE Int. Symp. Electromagnetic Compatibility* pp. 121–126, 2010.

[5] M. G. Khun, *Filtered-Tempest Fonts* (2005) [Online], Available: http://www.cl.cam.ac.uk/ mgk25/ emsec/ softtempest-faq.html, 2005.

[6] H. Sekiguchi and S. Seto, "Measurement of radiated computer RGB signals," *Progress in Electromagnetic Research C*, pp, 1–12, 2009.

[7] Y. Suzuki and Y. Akiyama, "Jaming technique to prevent information leakage caused by unintentional emissions of PC video signals," *Proc. IEEE Int. Symp. Electromagnetic Compatibility*, pp. 138–142, 2010.

[8] H. Sekiguchi," Novel information leakage threat for input operations on touch screen mnitors caused by electromagnetic noise and its countermeasure method," *Progress in Electromagnetic Research B*, pp. 399–419, 2012.

[9] M. G. Khun, " Compromising Emanations of LCD TV Sets," *IEEE Transactions on Electromagnetic Compatibility*, vol. 55, pp. 564–570, June 2013.

[10] D. Agrawal, B. Archambeult, J. R. Rao, and P. Rohatgi. The EM side-channel(s). In *Proc. of Cryptographic Hardware and Embedded Systems - CHES 2002*, pages 29–45, 2002.

[11] K. Gandolfi, C. Mourtel, and F. Olivier. Electromagnetic analysis: concrete results. In *Proc. of Cryptographic Hardware and Embedded Systems - CHES 2001*, pages 251–261, 2001.
[12] J. J. A. Fournier, S. Moore, H. Li, R. Mullins, and G. Taylor. Security evaluation of asynchronous circuits. In *Proc. of Cryptographic Hardware and Embedded Systems - CHES 2003*, pages 137–151, 2003.
[13] T. Plos, M. Hutter, and C. Herbst. Enhancing side-channel analysis with low-cost shielding techniques. In *Proc. of Austrochip*, 2008.
[14] F. Poucheret, L. Barthe, P. Benoit, L. Torres, P. Maurine, and M. Robert. Spatial EM jamming: A countermeasure against EM Analysis? In *18th IEEE/IFIP VLSI System on Chip Conf. (VLSI-SoC)*, pages 105–110, 2010.
[15] J. J. Quisquater and D. Samyde. Electromagnetic analysis (EMA): measures and counter-measured for smart cards. In *Proc. of E-smart*, pages 200–210, 2001.
[16] H. Tanaka. Information leakage via electromagnetic emanations and evaluation of Tempest countermeasures. In *Lecture notes in computer science, Springer*, pages 167–179, 2007.
[17] Y. Hayashi, N. Homma, T. Mizuki, H. Shimada, T. Aoki, H. Sone, L. Sauvage, and J. L. Danger, "Efficient evaluation of EM radiation associated with information leakage from cryptographic devices," *IEEE Transactions on Electromagnetic Compatibility*, vol. 55, pp.555–563, June 2013.
[18] H. Sekiguchi and S. Seto, "Study on maximum receivable distance for radiated emission of information technology equipment causing information leakage", *IEEE Transactions on Electromagnetic Compatibility*, vol. 55, pp.547–554, June 2013.
[19] Y. Hayashi, N. Homma, T. Mizuki, T. Aoki, H. Sone, L. Sauvage, J. L. Danger, "Analysis of electromagnetic information leakage from cryptographic devices with different physical structures", *IEEE Transactions on Electromagnetic Compatibility*, vol. 55, pp. 571–580, June 2013.
[20] H. W. Ott *Electromagnetic Compatibility Engineering*, Wiley, 2009.
[21] C. R. Paul, *Introduction to Electromagnetic Compatibility*, Wiley, 2006.
[22] J. Demme, R. Martin, A. Waksman, S. Sethumadhavan, "Side-channel vulnerability factor: A metric for measuring information leakage," *39th Annual International Symposium on Computer Architecture (ISCA) 2012*, pp. 106–117, June 2012.
[23] S. Chen, R. Wang, X. F. Wang, and K. Zhang, "Side-channel leaks in web applications: a reality today, a challenge tomorrow," *IEEE Symposium on Security and Privacy (SP)2010*, pp. 191–206, May 2010.
[24] R. Hund, C. Willems, T. Holz, "Practical timing side channel attacks against kernel space ASLR", *IEEE Symposium on Security and Privacy (SP), 2013*, pp. 191–205, May 2013.
[25] D. Gullasch, E. Bangerter, S. Krenn, "Cache games bringing access-based cache attacks on AES to practice", *IEEE Symposium on Security and Privacy (SP), 2011*, pp. 490–505, May 2011.

Alenka Zajić (S'99-M'09-SM'13) Alenka Zajić received the B.Sc. and M.Sc. degrees form the School of Electrical Engineering, University of Belgrade, in 2001 and 2003, respectively. She received her Ph.D. degree in Electrical and Computer Engineering from the Georgia Institute of Technology in 2008. Currently, she is an Assistant Professor in School of Electrical and Computer Engineering at Georgia Institute of Technology. Prior to that, she was visiting faculty in School of Computer Science at Georgia Institute of Technology, a post-doctoral fellow in the Naval Research Laboratory, and design engineer at Skyworks Solutions Inc. Her research interests span areas of electromagnetics, wireless communications, signal processing, and computer engineering.

Dr. Zajić received the Neal Shepherd Memorial Best Propagation Paper Award, the Best Paper Award at ICT 2008, the Best Student Paper Award at WCNC 2007, and was also the recipient of the Dan Noble Fellowship in 2004, awarded by Motorola Inc. and IEEE Vehicular Technology Society for quality impact in the area of vehicular technology. Currently, she is an editor for IEEE Transactions on Wireless Communications.

Milos Prvulovic (S'97-M'03-SM'09) received the B. Sc. degree in electrical engineering from the University of Belgrade in 1998, and the MS and PhD degrees in computer science from the University of Illinois at Urbana-Champaign in 2001 and 2003, respectively. He is an Associate Professor in the School of Computer Science at the Georgia Institute of Technology, where he joined in 2003. His research interests are in computer architecture, especially hardware support for software monitoring, debugging, and security.

He is a past recipient of the NSF CAREER award, a senior member of the ACM, the IEEE, and the IEEE Computer Society.