Journal of
Low Power Electronics and Applications

MDPI

Article

Selective Noise Based Power-Efficient and Effective Countermeasure against Thermal Covert Channel Attacks in Multi-Core Systems

Parisa Rahimi [1,*], Amit Kumar Singh [1] and Xiaohang Wang [2]

[1] School of Computer Science and Electronic Engineering, University of Essex, Wivenhoe Park, Colchester CO4 3SQ, UK; a.k.singh@essex.ac.uk
[2] School of Software Engineering, South China University of Technology, Guangzhou 511436, China; xiaohangwang@scut.edu.cn
* Correspondence: pr19863@essex.ac.uk

Abstract: With increasing interest in multi-core systems, such as any communication systems, infrastructures can become targets for information leakages via covert channel communication. Covert channel attacks lead to leaking secret information and data. To design countermeasures against these threats, we need to have good knowledge about classes of covert channel attacks along with their properties. Temperature–based covert communication channel, known as Thermal Covert Channel (TCC), can pose a threat to the security of critical information and data. In this paper, we present a novel scheme against such TCC attacks. The scheme adds selective noise to the thermal signal so that any possible TCC attack can be wiped out. The noise addition only happens at instances when there are chances of correct information exchange to increase the bit error rate (BER) and keep the power consumption low. Our experiments have illustrated that the BER of a TCC attack can increase to 94% while having similar power consumption as that of state-of-the-art.

Keywords: selective noise; multi-core systems; thermal covert channel; countermeasure; attack detection

Citation: Rahimi, P.; Singh, A.K.; Wang, X. Selective Noise Based Power-Efficient and Effective Counter-Measure against Thermal Covert Channel Attacks in Multi-Core Systems. *J. Low Power Electron. Appl.* **2022**, *12*, 25. https://doi.org/10.3390/jlpea12020025

Academic Editor: Andrea Acquaviva

Received: 6 January 2022
Accepted: 7 March 2022
Published: 3 May 2022

1. Introduction

Covert channels are communication channels used to transmit information. In chip-level security, covert channels are types of attacks that can create the capability of transferring data and information between processes that are not allowed to communicate by the system security policy. Two parties with the aim to exchange covert information can easily communicate and transfer information over a shared network without being detected. Therefore, it is hard to identify covert channels, which hence can contribute to very serious damage to security approaches if they are used for malicious purposes. In many communication media, the covert channel information is seen, such as timing, heat, or indistinct sounds. There is a wide range of side channels that may exist in a multi-core system, however, a covert channel, which uses heat as a way to transmit information and data, can be particularly dangerous.

The heat transfers are known as thermal covert channels (TCC) [1]. Thermal covert channels can be traced in multi-core systems. As in any communication system, a thermal covert channel includes a pair consisting of a transmitter and a receiver [2]. On the transmitter side, the temperature signals are generated from sensitive data such as user passwords by manipulating other activities like power consumption. The receiver's side, which is on the other end of the data transmission, reads its thermal sensor and recovers the original sensitive information or data transmitted [3]. Figure 1 illustrates an eight-core chip example, which assumes that there is a covert channel between core A and core B. It should be noted that core A is placed in a secured zone, where sensitive information does not have permission to be shared with the other cores outside this zone, and core B is

located in a non-secured zone [4]. The temperature signal can move around the chip by heat transfer among processor cores until it reaches the destination (receiver) [3,5]. In such cases, the bit "1" shows as a range of rising and is full of the temperature signal, while a bit "0" shows no changes. While thermal signals are committed to transferring temperature signals over a chip, it is highly likely to be degraded or disrupted by the environment temperature variations or the thermal noise [5]. An advanced countermeasure against the thermal covert channel needs to position affected cores and fight the TCC attacks.

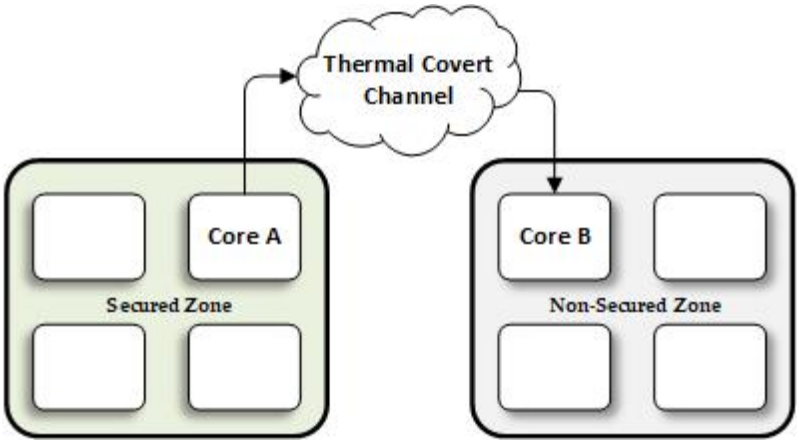

Figure 1. TCC communication in an eight multi-core system. The arrow from core A to core B indicates that the heat flows from core A to core B.

In recent years, some new designs of thermal covert channels have been proposed. For instance, Masti et al. [4] introduced a new design of thermal covert channels that has improved noise immunity and throughput. They showed the feasibility of side channels and especially thermal channels on multi-core systems for the first time, e.g., they created a 1-hop TCC channel (a channel of transmitter and receiver are 1-hop apart from each other) that could reach a throughput of 1.33 bps along with a BER of 11%. In this study, researchers did not consider power consumption and high BER.

In separate research [6] it was confirmed that a TCC channel has a throughput of more than 45 bps, and it also showed a TCC channel with a throughput of more than 5 bps with a BER less than 1%. Apparently, the ever-growing performance of the thermal covert channels pose serious risks and dangers to sensitive information or data security in today's many-core/multicore systems [7]. However, it is stated in [3,5] that TCC can be countered by decreasing the frequency of the system by Dynamic Voltage Frequency Scaling (DVFS) technology, producing jamming noise with the same transmission frequency as TCC [8] or as in [9] by limiting the accessibility to thermal information, etc.

The increase in operating temperature of an embedded system results in a bad experience and reliability problems. Reliability issues through system design are being extensively examined. It seems that in today's digital world, TCC attacks are posing an increasingly big security threat to multi-core/many-core systems, however, there have been little works done to improve the defence strategy or countermeasure against them.

In this paper, we focus on thermal covert channel attacks and propose a novel selective noise-based countermeasures. We have aimed to consider the weakness of previous works and propose a new countermeasure in a way that is undetectable by the attackers. For instance, Huang et al. [3,5] introduced a detection based on signal frequency scanning, positioning affected cores, and blocking to thwart possible TCC attacks, as shown in Figure 2. All experiments are performed using a simulator, Sniper-v7.2 [10,11], and more details of the simulator are provided in Section 5. However, it seems that the temperature signal is still at risk because the attackers can come up with a new method and find the thermal signal's pattern. Jiachen W et al. [12] suggested a countermeasure strategy based on scanning frequency spectrum rapidly to detect any possible attack. When a

thermal covert channel is detected and its frequency of transmission is found, a strong noise source is applied to the transmission frequency band for targeted jamming. These examples [3,5,12] have motivated us to propose a new countermeasure scheme based on recording temperature and adding extra noise (extra thread), in fact, selective noise, that changes the pattern of the thermal signal. The purpose of adding a selective noise is to generate more temperature when the bit is "1", which results in increasing the duration of bit "1". For instance, if the bit was "1" just for 2 ms, by the proposed method we increased it to 5 ms and modified the behaviour of the temperature signal.

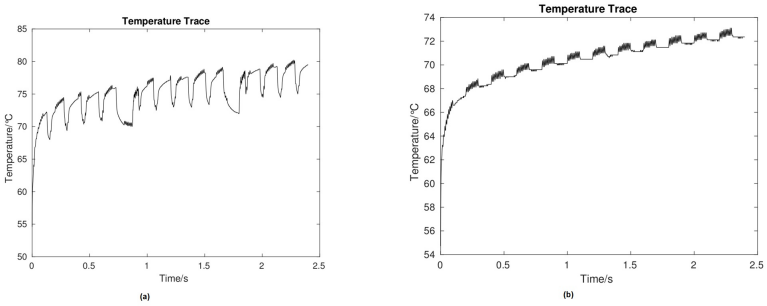

Figure 2. Thermal signal with and without countermeasure. (**a**,**b**) Temperature traces recorded by the receiver core without and with DVFS control, respectively.

The rest of this paper is structured as follows. In Section 2, we survey the previous works. Section 3 presents the system and threat model thermal of covert channels. Section 4, introduces the security scenario for our TCC. Section 5 discusses the results and our contribution. The final section (Section 6) concludes this paper.

2. Related Works

Covert channel attacks pose a security threat against any aspect or part of the computer and information systems to multi/many-core chips [13].

They have been studied for different purposes, such as operating systems [13], multi-core chips [1,14], and cloud systems [15]. Covert channels can be set up using a large number of communication media that span from inaudible sound [16], inter-arrival timing of packets [17,18], magnetic field [19,20], inter-light [21,22] to voltage [23].

At the chip level, covert channels can use dynamic frequency scaling [24], exploit cache-timing [25,26], etc., for data and information transmissions. For example, Darwish et al. [26] introduced a new online streaming methodology for the mitigation of covert timing channels. This method eliminates covert timing channels while having an impact on the Quality of Service (QoS). To test the performance of the proposed model, the classification-based strategy was applied. They confirmed that the classification accuracy of this model flow fell down to 50%. However, in this work, the main focus was on the timing of the covert channel and performances, while our paper relies on the temperature, power consumption, and BER of the thermal covert channel in the multi-core system.

The term covert channel is explored, when the source and the sink transfer the data or information actively against the term side-channel, which is utilized, where attackers recognized an unaware system to deduce sensitive data and information, e.g., a user password or a cryptographic key, and change them into the temperature signals [6]. Researching the security issues related to isolation and separation in computing systems is a well-defined area of study. In 1973, Lampson et al. [27] analysed this issue and mentioned the possibility of using covert channels, i.e., detecting system properties not designed for communication, with the aim of leaking restricted data. These attacks pose a significant security threat against any aspect or part of our system, from cloud systems to operating systems and multi/many-core chips, etc. [5].

Such as any countermeasure strategies, the TCC countermeasures include two main steps, detection and defence, which are summarized below.

2.1. Detection

In any countermeasure strategy, the first and most important step is detection. There are different methods to detect attacks introduced. For instance, the experiments by Kean et al. [28] illustrated that one system can include different tags with several codes, which can all be detected in a relatively short time, during the normal operation of the system. Experiments presented by Murdoch et al. [29] illustrated those changes in clock skews resulting from modest temperature changes, which can be remotely identified through network packet timestamps, even over tens of router hops [30]. Huang et al. [3,5] and Jiachen W et al. [12] showed that for the detection step, each core can measure the spectrum of its CPU workload traces that are recorded through a few fixed time intervals, and then use it as a frequency scanning technique to detect if there exist any thermal covert channel attacks.

2.2. Defence

The thermal covert is not similar to many types of covert channels. It does not rely on any shared sources such as catch or memory, which supports it in easily circumventing the system's defence. TCC, which uses a simple on-off keying line coding strategy to encode bit "0" and bit "1", is presented in [4]. Recis et al. [31] researched thermal-related attacks where sensitive information (e.g., a user password) is transmitted by reading the duration and speed of the fan. The angular speed of the fan changes based on the chip temperature to intercept it from overheating. A thermal covert channel over an x86-based scheme could transmit data with an average BER of 13.22% [4]. A different encoding platform based on return-to-zero was proposed in [14]. In this method, it was illustrated that a high transmission frequency could prevent thermal noise from participation with other active cores. Furthermore, Jiachen W et al. [12] suggested a countermeasure strategy based on scanning frequency spectrum rapidly to detect any possible attack. When a thermal covert channel is detected, and its frequency of transmission is found, a strong noise source is applied to the transmission frequency band for targeted jamming, so that a thermal covert channel could be blocked with a packet error rate (PER) of 85%. In this strategy, by adding strong scours noise, the systems may face overheating during the transmission process or result in increased power consumption. Another method that can be used to fight the TCC attacks is Dynamic Voltage Frequency Scaling (DVFS) [3,5]. In this countermeasure, after detecting the TCC attack and finding its position, it is attempted to block and fight the possible attacks. In the scheme, Huang et al. [3,5] confirmed that this approach could cause TCC attacks to suffer extremely high BER (>92%), by applying the DVFS strategy. In all of these strategies, the sensitive data is still in danger. For instance, in [4] BER is very low, in [12] devices may face overheating and high power consumptions, and in [3,5] thieves can have access to the thermal signal's pattern. By considering these issues, we proposed a novel countermeasure to address them.

3. System and Threat Model

In this section, we present the thermal covert channel architecture and explain the challenges linked with TCC attacks. Then we show our threat model considered for all the implemented attacks. Finally, we describe the methodology used in the TCC attack.

3.1. TCC Communication Architecture

Only limited works have focused on the thermal covert channel due to the complexity of the attacks (e.g., require expert knowledge and skills) and noisy traces [32]. A thermal covert channel relies on a pair consisting of a transmitter and a receiver, which is shown in Figure 3. To start the transmission process, first, the receiver and the transmitter require achieving agreement on the transmission frequency for the thermal covert channel. Then, the transmitter reads the sensitive information and data and packetizes them with a so-called Error Correcting Code (ECC) [2]. When the packet is created, it requires to be changed to the temperature signal for transmission. Basically, the transmitter sends the

thermal signal by controlling the power consumption of the transmitter core [33,34]. It means that the transmitter runs computation-intensive code with a particular purpose to generate a high temperature, inserting idle CPU cycles to cool down the chip and generate a low temperature [3,5]. For instance, a program is used to change the temperature based on the bit, e.g., if '0' then no change (inserting idle cycles to cool down the chip) and if '1' then the temperature is made high followed by a low to keep a low average temperature. On the receiver side, it gets the thermal signal from its local temperature sensor and transforms it into binary bitstreams. Then the receiver checks the ECC code for the integrity of the packet. If the received packet is recognized and contains the particular preamble field (e.g., 1010101) throughout the transmission process, the receiver extracts the data fields from the packet; otherwise, the receiver rejects the binary bitstreams [35].

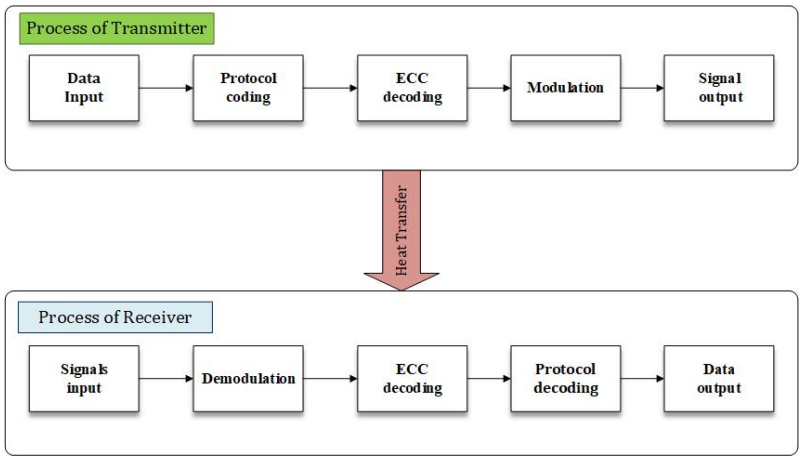

Figure 3. The components and signal flow of a communication system through a thermal covert channel, from the transmitter to the receiver.

3.2. Challenges of Thermal Covert Channel Attack

The first thermal leakage issue is that they behave integrative, for instance, temperature accumulates when an operation is executed (e.g., Rivest–Shami–Adleman (RSA) decryption). From a previous operation, the contribution of thermal leakage still remains in its next operation. Therefore, it is probable that at a different time a moment of the same operation (e.g., a multiplication) will have a different starting temperature. Therefore, it is highly likely that the same process (e.g., a multiplication) at different time moments shows a different starting temperature [32]. To address this issue, there are two possible approaches. In the first method, pauses can be implemented between the operations. This can be performed by regularly stopping the clock or by implementing pauses after each process, for instance, regularly forcing a delay in the processor. The second solution for this issue is to periodically cool the processor after each operation. This approach can be implemented by adding an external cooling system, such as a high-speed fan [36]. Another issue that thermal traces face is the difference of temperature offset with time. While the power is controlled by a voltage regulator, the temperature of the processor and the environment are not directly controlled. So, the offset changes multiple times during implementation. Today, most systems and devices use the dynamic clock and voltage scaling to maintain the temperature in a specific safe range. Hence, the thermal leakage also shows drifts in the offset during the implementation processes. So, TCC analysis needs a mechanism to filter out such drifts, as they most likely cannot work on traces with different offsets [37].

The last issue is one method of modelling the physical performance of the thermal leakage by analysing the system as an RC network [38]. This network acts similar to a low-pass filter with a cut-off frequency somewhere in the kilohertz range. This frequency reaction can pose an issue, as a wide range of systems and computers run in the Gigahertz range. With a low-pass filter, it seems very hard and difficult to measure any useful information and data when the systems are running at 800 MHz, even if it is the first

order. Luckily, it is not needed to record every clock cycle for side-channel analyses to be differentiated. If these processes take long enough, differences can be visible in the thermal traces [32].

3.3. Threat Model

In thermal attacks, a malicious operation leaks the data and secret information by modulating the devices' temperature [39]. Except for a few studies, as reported in [36], which estimated the decay rate of the DRAM cells to determine the temperature of a TCC, most of the thermal covert channels need to have access to on-chip digital thermal sensors (DTS) to get the temperature information of the chip [33]. For dynamic thermal management, temperature sensors are essential and are deployed in chips in a wide range [40]. The accuracy and number of temperature sensors can be ever improved in the future [41]. The attackers (being the TCC transceiver) are able to read or reach the local thermal sensor data by calling the software interfaces such as MSR or Intel's processor Model Specific Register, with a normal resolution of 1 °C [42]. The accuracy of some of the latest digital thermal sensors can be up to 0.1 °C [43]. The defender is able to read the temperatures of all cores and distribute the detection and blocking programmes to each core [13].

Our threat model for TCC have the following assumptions:

- The attackers have direct access to the target device or systems in order to record and archive thermal traces of the executed decryption. Many key recoveries attacks use a common set of methods or techniques for analysing thermal traces. Simple thermal analysis (STA) includes directly inspecting traces to deduce sensitive data or information when measurements can be recorded to certain data properties [32];
- The attackers can acquire the ciphertext. In the transmission process, the ciphertext data will be employed as default storage parameters for the specific transaction. So, during this process, it is most likely the attackers can access the ciphertext [44];
- The attackers have access to a similar system or device. For example, using off-the-shelf parts such as AVR, ARM, etc;
- The attackers have the capability to slow and cool down the target operating systems. This assumption can be reached by manipulating the external crystal, using some external fan, or by forcing the target system to compute a wide range of tasks in parallel [3,32];
- Since the signal on the transmitter side is already distorted by adding noise, it would not be possible to detect it correctly at the receiver side by increasing the frequency of thermal sensors, because the information received would be different from the original information. Further, for proper synchronization of data, it is better for both transmitter and receiver thermal sensors to operate at the same frequency. The transmitter and the receiver also need to reach an agreement on the frequency of transmission for the thermal covert channel [3].

4. Countermeasures against Thermal Covert Channel Attacks

Our proposed work has different steps to detect and fight against such thermal covert channel attacks. For the detection step, we used frequency scanning-based detection, which was presented by Huang et al. [3,5].

- Step 1 (Detection): The detection algorithm uses the module of bandpass filtering to filter out the thermal signals that are out of every band of interest. The detection algorithm is also designed in a way that it can manage TCC, whose receiver and transmitter runs through two different channels. While designing the countermeasure method, we consider that the receiver and the transmitter have access to the same thermal data files, and also that the transmitter and the receiver threads are both sited in the same physical core and share the same thermal sensors, known as Intra-core channel [3,5]. The frequency scanning process is used to detect any existing TCC channels.

In addition, each core specifies the signal's maximum amplitude from the frequency scanning process and leads it into a module known as the decision-making module. The decision-making module compares the signal's amplitude against a threshold ρ (ρ is the pre-set threshold for the signal amplitude in a detection cycle.) and decides whether a signal (e.g., 1 0 1 0) is from a covert channel or not. If the amplitude of the signal is higher than ρ, an attack is present; otherwise, there is no attack happening. Statistically, the threshold ρ is selected experimentally to be 0.02 dB [5].
Based on frequency scanning, if the defender, denoted as, global manager, identified the TCC channel exists, the proposed countermeasure is applied;

- Step 2 (Classify bits): For applying the proposed countermeasure, we need to classify bits into "1" and "0" according to the thermal signal changes. In our work, the bit "1" represents changes in the thermal signal, while the bit "0" illustrates no changes;
- Step 3 (Check bits): For this step, the system checks if the bit is "1" or not by considering the changes in the thermal signal. Then, if the bit is "1" the operation will start;
- Step 4 (Adding an extra noise): The last step is adding an extra noise to the temperature signal, which results in generating more heat to the system where the bit is "1", therefore, there is no bit change for a long duration, as Figure 4 shows.

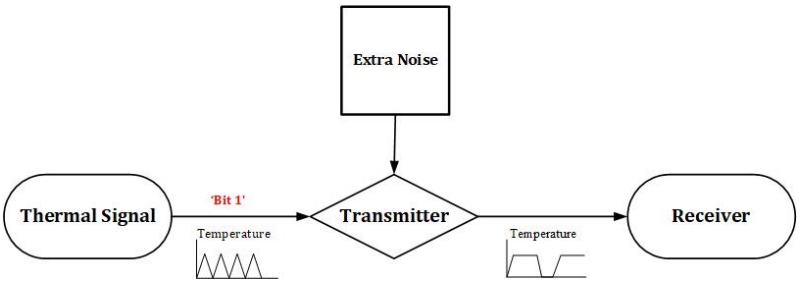

Figure 4. Proposed countermeasure process.

In the proposed method, once a core is determined to be associated with a thermal covert channel, a countermeasure shall apply to fight any future thermal covert channel transmission. Here we assume that a thermal covert channel thread is running on a dedicated logical core all over a full communication session.

We add extra noise to the thermal signal to generate more temperature when the thermal signal is at high temperature (t_h), which results in increased duration of t_h and changing the pattern of the thermal signal and making it secure.

Note that the global manager cannot immediately apply countermeasure to control the suspicious core. First, the global manager scales down the temperature of the suspicious core for a few seconds (e.g., 0.5 s) to decrease the risk, and then a countermeasure is implemented to block any future TCC transmission [3,5]. As for the simulation results, from Figure 5, one can see that the average temperature of a thermal signal without countermeasure is around 72 °C (Figure 5a) and when the countermeasure is applied the average temperature increases to 76 °C (Figure 5b).

4.1. Extra Noise Generation by Random Numbers

There are different methods introduced to generate random numbers. Some of them are very simple and some of them are very complicated. However, our purpose to use this operation is to create some extra temperature in a very simple method that does not require special types of equipment or lab to implement, and all researchers and designers can have very easy access to it [45,46]. Therefore, for implementing our idea, we used C/C++ programming and `int rolldie(int rand_ number)` function to create some random numbers. Algorithm 1 illustrates the process of generating random numbers. In the first step, the system gets two different random numbers (known as X1 and X2), then sums them to achieve the final results.

Algorithm 1 Psuedorandom Number Generator

Input: Two random numbers X1 and X2

Output: Psuedorandom sequence period of X1 + X2

```
for n = 100, do
    if n > 0 then
        X1 = rollDie();
        X2 = rollDie();
        Count n = X1 + X2
        return
    end if
end for
```

We started our experiment with 20 random numbers, however, it did not have any effect on our system, as the duration of this process was too short and did not generate significant temperature. Hence, we increased these numbers until we achieved the best results. The best result was reached when 100 random numbers were generated, and we got high BER and low power consumption. We considered this operation (generating random numbers) as one extra noise and then applied it to the transmission process. The results are shown in Table 1.

Table 1. One extra noise with different random numbers.

Countermeasures	Generate Random Numbers	Bit Error Rate	Avg. Power Consumption (w)
One Extra Noise	20	22%	22.50966
	50	22%	22.80794
	100	94%	23.74631
	150	94%	23.96022

During our work, we considered two important metrics: bit error rate and power consumption. In our proposed countermeasure, it can be difficult for the attackers to identify or guess the temperature signal pattern, so the sensitive information cannot easily leak. Consider that a system has a thermal covert channel with a transmission frequency of 100 Hz. Figure 5 shows the temperature signal without and with the proposed countermeasure. As Figure 5b illustrates, when the proposed countermeasure is applied, it shows a significantly different thermal profile.

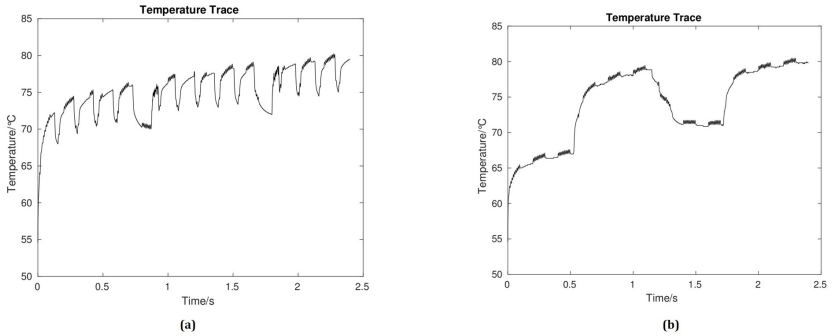

Figure 5. Temperature timing diagrams of the receiver core without (**a**) and with proposed countermeasure on the transmitter core (**b**).

In our proposed countermeasure, the BER of the TCC reaches 94%, and the power consumption did not increase significantly. A high BER indicates that our proposed countermeasure can be utilized to fight TCC attacks. On the other hand, system overheating leads to increased power consumption throughout the transmission process. Consequently, this issue is addressed in this paper.

4.2. Selective Noise

To fight TCC attacks and address the system's overheating issue, we proposed a new countermeasure, which essentially consists of selective noise. In other words, this operation is designed based on the recorded current temperature of bit "1". If the current temperature shows any changes or fluctuation, then the system decides to add extra noise or not, which is known as selective noise. We will discuss it in more detail in the next section. By adding selective noise, we could first change the pattern of the thermal signal, and then avoid increasing the unnecessary temperature, which results in high power consumption.

4.3. Selective Noise Countermeasure

The methodology is illustrated in Figure 6 and has the following main steps:

- Step 1 (Record temperature): This step consists of monitoring and recording the temperature. The extra noise is added when the bit is "1". The current temperature is then recorded by the temperature sensor and used for the next step;
- Step 2 (Temperature monitoring of processors and adding selective noise): In the last step, the next bit's temperature is compared with the recorded temperature (TR) from the previous step. If the temperature of the current bit is less than TR or starting to decrease, the selective noise is added to keep the temperature at the same level, otherwise it is moved to the next bit. This progress continues until the end of the transmission process.

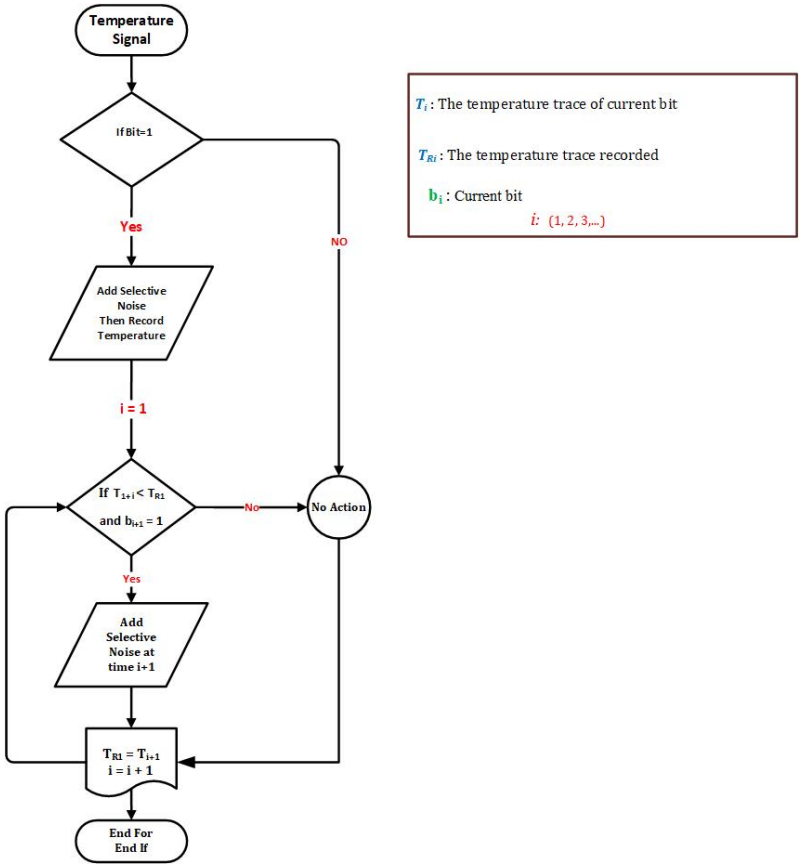

Figure 6. The process of adding selective noise.

Here, we adopt different countermeasure strategies that take into account both security requirements and power consumption. Specifically, adding selective noise effectively can change the pattern of the temperature signal. To compare with our previous method, the BER of the receiver does not show significant changes (94%) but the power consumption will not be extremely high. As Figure 7 shows, by adding selective noise, the thermal

signal shows different behaviour. The temperature fluctuation is decreased, and we could keep it at the same level, which results in avoiding system overheating and low power consumption. When the transmission process is complete, at the receiver side, the receiver reads the temperature sensor and records the thermal signal, and then decodes the signal to recover the original data. Next, if the received packet is specified to be uncompromised and includes a special preamble field during the transmission, the receiver extracts the data fields from the packet, otherwise, the receiver drops the binary bitstreams and waits for retransmission [5].

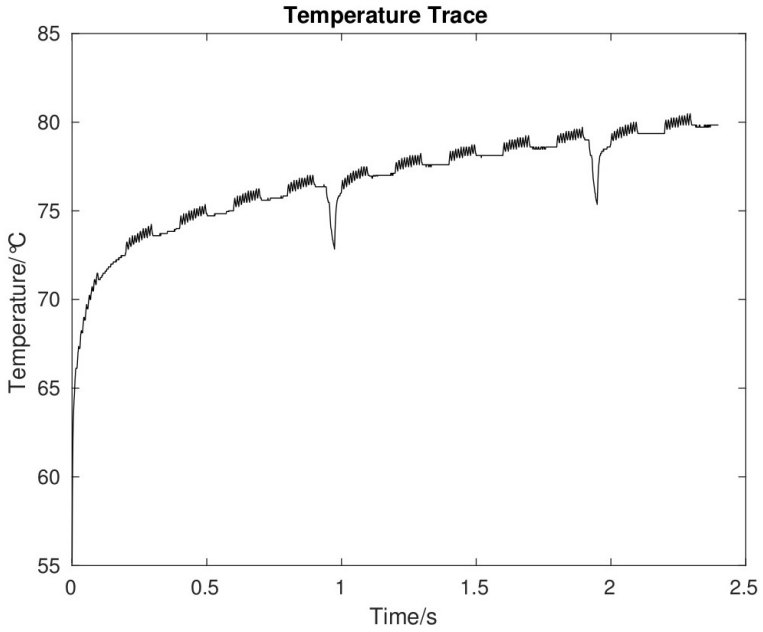

Figure 7. The process of adding selective noise.

5. Experimental Evaluation

Two sets of experiments have been performed, first, adding extra noise to every bit 1 and adding selective noise. Mostly, these experiments are implemented by using a multi-core simulator, Sniper-v7.2 [10,11], as a fast and reliable simulator for multi-core [47]. McPAT-v1.0 is integrated as the power model. McPAT is an integrated power, area, and timing modelling framework for multithreaded, and many-core/multi-core architectures [48]. This analysis has been done in order to thwart the covert thermal channel attack of our targeted device. To covertly transmit sensitive information from a secured zone to a non-secured one, the thermal covert channel programs require generating and measuring temperature signals by using the HotSpot simulator. We chose the Hotspot-v6.0 thermal model to dynamically produce temperatures for all the cores. We adopt a few benchmarks from PARSEC and SPLASH-2 and their temperature signals will be treated as the thermal noise to a thermal covert channel. The experimental setup uses Sniper and Hotspot simulation tools and TCC programs. Their configuration details are now tabulated in Table 2. The floor plan of the processor cores follows the one reported in [3,5].

As Table 2 shows, the transmission frequency of the TCC dynamically changes with a range of 10 Hz, 50 Hz, 150 Hz, and 200 Hz, etc., and the working CPU frequency of the transmitter/receiver core is 2000 MHz. In addition, it is identified whether there is a TCC attack or not. If an attack is found, the position of the transmitters/receivers associated with the thermal covert channels are required to be found. This positioning accuracy is referred to as P_{acc}.

$$P_{acc} = \begin{cases} 1 & P_{detected} = P_{transmitter//receiver} \\ 0 & \text{otherwise} \end{cases} \quad (1)$$

where:

$P_{detected}$: is the position (core id) of the detected thermal covert channel transmitter/receiver cores.
$P_{transmitter}$: is the actual position of the transmitter/receiver cores.

Table 2. Simulation configuration.

Sniper Configuration	
Instruction set architecture	x86-64
Operate System	Ubuntu 16.04.5 LTS
Number of cores	4 × 4, 8 × 8
Number of SMT threads per core	2
Frequency of CPUs (MHz)	2000
Benckmarks of PARSEC	Blackscholes, Canneal, Fluidanimate, Streamcluster, Swaptions, X-264, Dedup, Freqmine
Benckmarks of SPLASH-2	Raytrace, Barnes
Configuration for an integrated Hotspot	
Chip thickness	0.15 mm
Silicon thermal conductivity	100 W/(m·K)
Silicon specific heat capacity	1.75 × 106 J/(m^3·K)
Heat sink side	0.06 m
Heat sink thickness	6.9 mm
Heat sink thermal conductivity	400 W/(m·K)
Specific heat capacity of heat sink	3.55 × 106 J/(m^3·K)
Configuration for TCC programs	
Transmission frequency	10 Hz, 20 Hz, 50 Hz, 80 Hz, 100 Hz, 150 Hz, etc.
Preamble of a packet	1010101
Packet size in bits	64
ECC method	Hamming code

Table 3 shows the average accuracy of positioning the transceiver in different multi-core systems, which is around 97%. By using this detection strategy, we can almost always identify a TCC attack.

Table 3. Detection accuracy of P_{acc}.

System Sizes	P_{acc}
4 × 4	0.975
8 × 8	0.97

Table 4 illustrates that the proposed countermeasure strategy can efficiently fight TCC attacks, up to the point that with this proposed countermeasure, the average BER of the TCC attacks across 8 × 8 systems can be higher than 94%. An interesting fact is that the average power consumption when the selective noise is in effect does not increase significantly. The formula that is used for computing the average power consumption is as follows [47]:

$$P_avg = \frac{\sum_{c=1}^{N_c}(\sum_{u=1}^{N_c} 2_{cu})}{N_c}[W] \quad (2)$$

where:
N_c: The number of cores.

N_u: The number of functional units within a core.
P_{cu}: The average power consumption for functional unit u from core c.

Table 4. Experimental results.

Countermeasures	Bit Error Rate	Avg. Power Consumption (w)
TCC with periodically scanning the frequency and add a strong noise source [12].	85%	22.27
TCC with DVFS (Figure 2b) [3,5].	92%	22.31799
TCC with One Extra Noise (Figure 4b)	94%	23.74631
TCC with Selective Noise (Figure 6)	94%	22.81277

Figure 8 shows that the average BER's of our proposed countermeasure is always higher than that of the DVFS countermeasure for different system sizes.

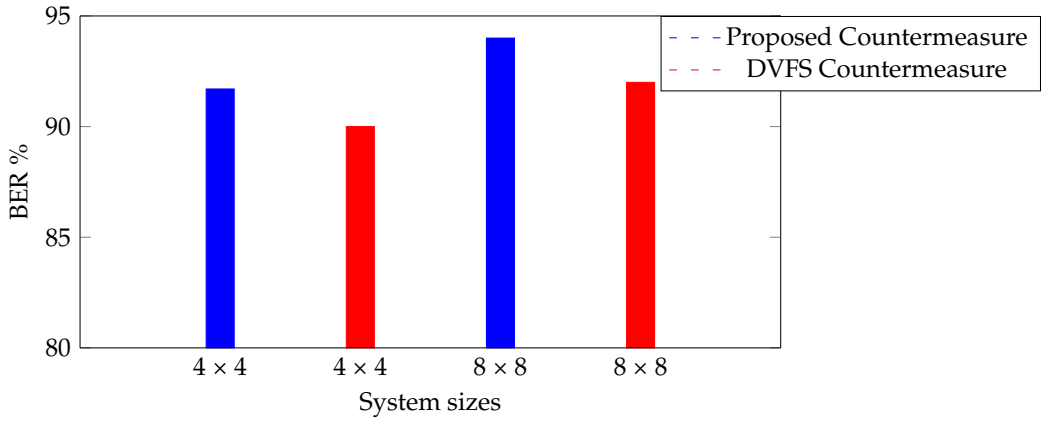

Figure 8. BER with the proposed method and DVFS in different systems.

We also determined that the average power consumption increased, along with the number of extra noises. The average and maximum temperatures also follow this trend. As shown in Table 5, by increasing the number of noises we are able to reach a high BER, but it may cause a rise in the temperature average power consumption of the systems, which is not good for the systems.

Table 5. Experimental results with different numbers of extra noise.

Countermeasures	Bit Error Rate	Avg. Power Consumption (w)
Two Extra Noise	95%	25.77204
Three Extra Noise	98%	28.19446
More than three Extra Noise (Five Extra Noise)	98%	34.20743

Table 6 indicates different aspects of considerations in TCC attacks. In thermal covert channel attacks, most of the existing methodologies did not consider high BER, low power consumption, and performance along with countermeasure. However, in the proposed work, we examine the thermal covert channel attacks in multi-core systems with the goals of optimizing two metrics: bit error rate (BER) and power consumption. We have also performed evaluations for various system sizes. This paper is intended to help the researchers and system designers in gaining deep insight into designing power efficient and secure multi-core systems in the future.

Table 6. Consideration of Various Aspects in Thermal Covert Channel Attacks by Existing and Our approaches.

References	Countermeasure	High BER	Performance	Low Power Consumption
[3,5]	✓	✓	✓	×
[4]	✓	×	×	×
[12]	✓	×	×	✓
[49]	✓	×	×	×
[50]	✓	×	×	×
[14]	✓	×	×	×
[1]	✓	×	×	×
[31]	✓	×	×	×
Proposed work	✓	✓	×	✓

Experimental results have confirmed that TCC attacks are possible in case of the leakage of sensitive information and data through several seconds or milliseconds. So, it seems that an application must repeatedly check the data and information that thwart such attacks, which may be costly or result in overheating of the devices.

The proposed work has a wide range of benefits, which are summarised below:

- The first and most important advantage of this work is the high bit error rate, which means that the vulnerability of the system will decrease significantly. Therefore, any possible TCC attacks can hardly happen in this methodology;
- Low power consumption: one of the most important metrics of any embedded system is power consumption [51]. This is mainly because the increase in power consumption is a significant factor for elevated temperatures. Therefore, low power consumption can affect the systems' temperatures;
- Last, but not least, is easy implementation. Some of the proposed countermeasure methods may need special types of equipment or labs, which may be costly or hard to implement, such as [32,37,52]. However, the proposed countermeasure can easily be employed for any device or system.

Despite all the benefits, the proposed approach has drawbacks., If the pattern of the main thermal signal is, e.g., 1 1 0 0 1 1 1 instead of 1 0 1 0 1 0 1, this strategy cannot work very well, as the proposed approach is based on both recording and monitoring the temperatures and then applying countermeasures, as mentioned in Section 4.3. Thus, when the temperature of the current bit (e.g., 1 1 1) is higher than the previous bit (e.g., 1 1), the system cannot decide to apply the selective-noise countermeasure. Hence, the multi-core system will be at risk.

Furthermore, the power spectrum of the temperature signals of a core when a normal application (e.g., Blackscholes from PARSEC) runs falls into three bands: band A, which ranges DC to 50 Hz, band B from 50 Hz to 400 Hz, and band C, which is higher than 400 Hz (cut-off frequency), and detection is not performed in band A and band C. The TCC signal in Band A can be easily damaged or disrupted by the thermal signal generated by a typical application [5] and band C is considered unavailable for data transmission. Therefore, in the aspect of countermeasure thermal covert channel attack, only possible TCC channels within band B have been detected [3]. Hence, if the transmission frequency is less than 50 Hz, the system cannot detect the possible attacks. All of these issues are taken into account for future work.

6. Conclusions

To the best of our knowledge, this is the first work to have used the pattern of the thermal signal to reduce the risks and dangers, which threaten sensitive information and data of the many-core/multi-core systems. Our experimental results have confirmed that

the novel proposed countermeasure could be practically against TCC attacks with a high BER of 94%. With its low overhead, power consumption, and complexity, the proposed countermeasure is a suitable scheme that can be used by many-core/multi-core systems to fight against such attacks. This study shows that thermal attacks are inevitable, however, appropriate countermeasures should be designed to fight them by considering other aspects of the system, such as power consumption as an important metric in the multi-core systems. In the future, we plan to perform evaluations with all the possible configurations in real hardware, which will need thorough engineering of the system under consideration.

Author Contributions: Conceptualization, P.R.; methodology, P.R.; software, P.R.; validation, P.R.; formal analysis, P.R.; writing—original draft preparation, P.R.; writing—review and editing, P.R., A.K.S. and X.W.; supervision, A.K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data supporting reported results can be found in https://github.com/Parisarhm/Selective-Noise-Based-Power-Efficient-and-Effective-Countermeasure-against-Thermal-Covert-Channel-.git accessed on 5 January 2022.

Conflicts of Interest: The author declare no conflict of interest.

References

1. Miedl, P.; Bartolini, D.B.; Thiele, L. On the Capacity of Thermal Covert Channels in Multicores. In Proceedings of the 11th European Conference on Computer Systems (EuroSys), London, UK, 18–21 April 2016; pp. 1–16.
2. Wang, Z.; Lee, R.B. Covert and Side Channels Due to Processor Architecture. In Proceedings of the 22nd Annual Computer Security Applications Conference (ACSAC'06), Miami Beach, FL, USA, 1–15 December 2006; Volume 8354, pp. 473–482.
3. Huang, H.; Wang, X.; Jiang, Y.; Singh, A.K.; Yang, M.; Huang, L. Detection of and Countermeasure against Thermal Covert Channel in Many-core Systems. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2021**, *41*, 252–265. [CrossRef]
4. Masti, R.J.; Rai, D.; Ranganathan, A.; Müller, C.; Thiele, L.; Capkun, S. Thermal covert channels on multi-core platforms. In Proceedings of the 24th USENIX Security Symposium (USENIX Security 15), Washington, DC, USA, 12–14 August 2015; pp. 865–880.
5. Huang, H.; Wang, X.; Jiang, Y.; Singh, A.K.; Yang, M.; Huang, L. On countermeasures against the thermal covert channel attacks targeting many-core systems. In Proceedings of the 57th ACM/IEEE Design Automation Conference (DAC), San Francisco, CA, USA, 20–24 July 2020; pp. 1–6.
6. Gu, P.; Stow, D.; Barnes, R.; Kursun, E.; Xie, Y. Thermal-aware 3D design for side-channel information leakage. In Proceedings of the IEEE 34th International Conference on Computer Design (ICCD), Scottsdale, AZ, USA, 2–5 October 2016; pp. 520–527.
7. Huang, X.; Wang, X.; Jiang, Y.; Singh, A.K.; Yang, M. Dynamic allocation/reallocation of dark cores in many-core systems for improved system performance. *IEEE Access* **2020**, *8*, 165693–165707. [CrossRef]
8. Okhravi, H.; Bak, S.; King, S.T. Design, implementation and evaluation of covert channel attacks. In Proceedings of the IEEE International Conference on Technologies for Homeland Security (HST), Waltham, MA, USA, 8–10 November 2010; pp. 481–487.
9. Alagappan, M.; Rajendran, J.; Doroslovački, M.; Venkataramani, G. DFS covert channels on multi-core platforms. In Proceedings of the IFIP/IEEE International Conference on Very Large Scale Integration (VLSI-SoC), Abu Dhabi, United Arab Emirates, 23–25 October 2017; pp. 1–6.
10. Florea, A.; Buduleci, C.; Chiş, R.; Gellert, A.; Vinţan, L. Enhancing the sniper simulator with thermal measurement. In Proceedings of the 18th International Conference on System Theory, Control and Computing (ICSTCC), Sinaia, Romania, 17–19 October 2014; pp. 31–36.
11. Carlson, T.E.; Heirman, W.; Eyerman, S.; Hur, I.; Eeckhout, L. An evaluation of high-level mechanistic core models. *ACM Trans. Archit. Code Optim. (TACO)* **2014**, *11*, 1–25. [CrossRef]
12. Wang, J.; Wang, X.; Jiang, Y.; Singh, A.K.; Huang, L.; Yang, M. Combating Enhanced Thermal Covert Channel in Multi-/Many-Core Systems With Channel-Aware Jamming. *IEEE Trans. Comput.-Aided Des. Integr. Circuits Syst.* **2020**, *39*, 3276–3287. [CrossRef]
13. Wu, Z.; Xu, Z.; Wang, H. Whispers in the hyper-space: high-bandwidth and reliable covert channel attacks inside the cloud. *IEEE/ACM Trans. Netw.* **2014**, *23*, 603–615. [CrossRef]
14. Tuptuk, N.; Hailes, S. Covert channel attacks in pervasive computing. In Proceedings of the IEEE International Conference on Pervasive Computing and Communications (PerCom), St. Louis, MO, USA, 23–27 March 2015; pp. 236–242.
15. Costa, G.; Pinelli, F.; Soderi, S.; Tolomei, G. Covert Channel Attack to Federated Learning Systems. *arXiv* **2021**, arXiv:2104.10561.

16. Deshotels, L. Inaudible sound as a covert channel in mobile devices. In Proceedings of the 8th USENIX Workshop on Offensive Technologies (WOOT'14), San Diego, CA, USA, 19 August 2014.
17. Yao, F.; Venkataramani, G.; Doroslovački, M. Covert timing channels exploiting non-uniform memory access based architectures. In Proceedings of the Great Lakes Symposium on VLSI 2017, Banff, AB, Canada, 10–12 May 2017; pp. 155–160.
18. Zhang, X.; Tan, Y.A.; Liang, C.; Li, Y.; Li, J. A covert channel over volte via adjusting silence periods. *IEEE Access* **2018**,*6*, 9292–9302. [CrossRef]
19. Matyunin, N.; Szefer, J.; Biedermann, S.; Katzenbeisser, S. Covert channels using mobile device's magnetic field sensors. In Proceedings of the 2016 21st Asia and South Pacific Design Automation Conference (ASP-DAC), Macau, China, 25–28 January 2016; pp. 525–532.
20. Guri, M.; Monitz, M.; Elovici, Y. SBee: Air-gap covert-channel via electromagnetic emission from USB. In Proceedings of the 2016 14th Annual Conference on Privacy, Security and Trust (PST), Auckland, New Zealand, 12–14 December 2016; pp. 264–268.
21. Guri, M.; Hasson, O.; Kedma, G.; Elovici, Y. VisiSploit: An optical covert-channel to leak data through an air-gap. *arXiv* **2016**, arXiv:1607.03946.
22. Liu, W.; Zhou, X.; Huo, J.; Yan, K. Modeling of visible light channel based on matrix reconstruction. *Int. Soc. Opt. Photonics* **2016**, *9902*, 990205.
23. Gnad, D.R.; Nguyen, C.D.K.; Gillani, S.H.; Tahoori, M.B. Voltage-based Covert Channels in Multi-Tenant FPGAs. *IIACR Cryptol. ePrint Arch.* **2019**, *2019*, 1394.
24. Bossuet, L. Dvfs as a security failure of trustzone-enabled heterogeneous soc. In Proceedings of the 2018 25th IEEE International Conference on Electronics, Circuits and Systems (ICECS), Bordeaux, France, 9–12 December 2018; pp. 489–492.
25. Ge, Q.; Yarom, Y.; Cock, D.; Heiser, G. A survey of microarchitectural timing attacks and countermeasures on contemporary hardware. *J. Cryptogr. Eng.* **2018**, *8*, 1–27. [CrossRef]
26. Darwish, O.; Al-Fuqaha, A.; Brahim, G.B.; Jenhani, I.; Anan, M. Towards a streaming approach to the mitigation of covert timing channels. In Proceedings of the 2018 14th International Wireless Communications & Mobile Computing Conference (IWCMC), Limassol, Cyprus, 25–29 June 2018; pp. 255–260.
27. Lampson, B.W. A note on the confinement problem. *Commun. ACM* **1973**, *16*, 613–615. [CrossRef]
28. Iakymchuk, T.; Nikodem, M.; Kępa, K. Temperature-based covert channel in FPGA systems. In Proceedings of the 6th International Workshop on Reconfigurable Communication-Centric Systems-on-Chip (ReCoSoC), Montpellier, France, 20–22 June 2011; pp. 1–7.
29. Murdoch, S.J. Hot or not: Revealing hidden services by their clock skew. In Proceedings of the 13th ACM Conference on Computer and Communications Security, Alexandria, VA, USA, 30 October–3 November 2006; pp. 27–36.
30. Hutter, M.; Schmidt, J.M. The temperature side channel and heating fault attacks. In Proceedings of the International Conference on Smart Card Research and Advanced Applications, Berlin, Germany, 27–29 November 2013; Springer: Berlin/Heidelberg, Germany, 2013; pp. 219–235.
31. Reis, C.; Barth, A.; Pizano, C. Browser Security: Lessons from Google Chrome: Google Chrome developers focused on three key problems to shield the browser from attacks. *Queue* **2009**, *7*, 3–8. [CrossRef]
32. Lee, J.S.; Skadron, K.; Chung, S.W. Predictive temperature-aware DVFS. *IEEE Trans. Comput.* **2009**, *59*, 127–133. [CrossRef]
33. Hackenberg, D.; Schöne, R.; Ilsche, T.; Molka, D.; Schuchart, J.; Geyer, R. An energy efficiency feature survey of the intel haswell processor. In Proceedings of the IEEE International Parallel and Distributed Processing Symposium Workshop, Hyderabad, India, 25–29 May 2015; pp. 896–904.
34. Singh, A.K.; Dey, S.; McDonald-Maier, K.; Basireddy, K.R.; Merrett, G.V.; Al-Hashimi, B.M. Dynamic energy and thermal management of multi-core mobile platforms: A survey. *IEEE Des. Test.* **2020**, *37*, 25–33. [CrossRef]
35. Vateva-Gurova, T.; Suri, N. On the Detection of Side-Channel Attacks. In Proceedings of the IEEE 23rd Pacific Rim International Symposium on Dependable Computing (PRDC), Taipei, Taiwan, 4–7 December 2018; pp. 185–186.
36. Happe, M.; Agne, A.; Plessl, C. Measuring and predicting temperature distributions on FPGAs at run-time. In Proceedings of the International Conference on Reconfigurable Computing and FPGAs, Cancun, Mexico, 30 November–2 December 2011; pp. 55–60.
37. Gao, F.; Zhu, L.; Gai, K.; Zhang, C.; Liu, S. Achieving a covert channel over an open blockchain network. *IEEE Netw.* **2020**, *34*, 6–13. [CrossRef]
38. Wen, S.; Wang, X.; Singh, A.K.; Jiang, Y.; Yang, M. Performance optimization of many-core systems by exploiting task migration and dark core allocation. *IEEE Trans. Comput.* **2020** *71*, 92–106. [CrossRef]
39. Chen, S.; Xiong, W.; Xu, Y.; Li, B.; Szefer, J. Thermal covert channels leveraging package-on-package DRAM. In Proceedings of the 18th IEEE International Conference On Trust, Security And Privacy in Computing and Communications/13th IEEE International Conference On Big Data Science And Engineering, Rotorua, New Zealand, 5–8 August 2019; pp. 319–326.
40. Saligane, M.; Khayatzadeh, M.; Zhang, Y.; Jeong, S.; Blaauw, D.; Sylvester, D. All-digital SoC thermal sensor using on-chip high order temperature curvature correction. In Proceedings of the 2015 IEEE Custom Integrated Circuits Conference (CICC), San Jose, CA, USA, 28–30 September 2015; pp. 1–4.
41. Chundi, P.K.; Zhou, Y.; Kim, M.; Kursun, E.; Seok, M. Hotspot monitoring and temperature estimation with miniature on-chip temperature sensors. In Proceedings of the 2017 IEEE/ACM International Symposium on Low Power Electronics and Design (ISLPED), Taipei, Taiwan, 24–26 July 2017; pp. 1–6.

42. Murdock, K.; Oswald, D.; Garcia, F.D.; Van Bulck, J.; Gruss, D.; Piessens, F. Plundervolt: Software-based fault injection attacks against Intel SGX. In Proceedings of the 2020 IEEE Symposium on Security and Privacy (SP), San Francisco, CA, USA, 18–21 May 2020; pp. 1466–1482.
43. Pan, S.; Makinwa, K.A. A 0.25 mm 2-Resistor-Based Temperature Sensor With an Inaccuracy of 0.12° C (3σ) From -55 °C to 125 °C. *IEEE J. Solid-State Circuits* **2018**, *53*, 3347–3355. [CrossRef]
44. Kean, T.; McLaren, D.; Marsh, C. Verifying the authenticity of chip designs with the DesignTag system. In Proceedings of the IEEE International Workshop on Hardware-Oriented Security and Trust, Anaheim, CA, USA, 9 June 2008; pp. 59–64.
45. Raffaelli, F.; Ferranti, G.; Mahler, D.H.; Sibson, P.; Kennard, J.E.; Santamato, A.; Sinclair, G.; Bonneau, D.; Thompson, M.G.; Matthews, J.C. A homodyne detector integrated onto a photonic chip for measuring quantum states and generating random numbers. *Quantum Sci. Technol.* **2018**, *3*, 025003. [CrossRef]
46. Vivoli, V.C.; Sekatski, P.; Bancal, J.D.; Lim, C.C.W.; Martin, A.; Thew, R.T.; Zbinden, H.; Gisin, N.; Sangouard, N. Comparing different approaches for generating random numbers device-independently using a photon pair source. *New J. Phys.* **2015**, *17*, 023023. [CrossRef]
47. Chiş, R.; Florea, A.; Buduleci, C.; Vinţan, L. Multi-objective optimization for an enhanced multi-core SNIPER simulator. *Proc. Rom. Acad.-Ser. A* **2018**, *19*, 85–93.
48. Aljuffri, A.; Zwalua, M.; Reinbrecht, C.R.W.; Hamdioui, S.; Taouil, M. Applying Thermal Side-Channel Attacks on Asymmetric Cryptography. *IEEE Trans. Very Large Scale Integr. (VLSI) Syst.* **2009**, *29*, 1930–1942. [CrossRef]
49. Zander, S.; Branch, P.; Armitage, G. Capacity of temperature-based covert channels. *IEEE Commun. Lett.* **2010**, *15*, 82–84. [CrossRef]
50. Long, Z.; Wang, X.; Jiang, Y.; Cui, G.; Zhang, L.; Mak, T. Improving the efficiency of thermal covert channels in multi-/many-core systems. In Proceedings of the 2018 Design, Automation & Test in Europe Conference & Exhibition (DATE), Dresden, Germany, 19–23 March 2018; pp. 1459–1464.
51. Rahimi, P.; Singh, A.K.; Wang, X.; Prakash, A. Trends and Challenges in Ensuring Security for Low-Power and High-Performance Embedded SoCs. In Proceedings of the 2021 IEEE 14th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC), Singapore, 20–23 December 2021; pp. 226–233.
52. Claeys, T.; Rousseau, F.; Simunovic, B.; Tourancheau, B. Thermal covert channel in Bluetooth low energy networks. In Proceedings of the 12th Conference on Security and Privacy in Wireless and Mobile Networks, Miami, FL, USA, 15–17 May 2019; pp. 267–276.