2013 International Conference on Availability, Reliability and Security

On the Practicability of Cold Boot Attacks

Michael Gruhn and Tilo Müller
Friedrich-Alexander-Universität
Erlangen-Nürnberg, Germany
{*michael.gruhn,tilo.mueller*}@*cs.fau.de*

***Abstract*—Even though a target machine uses full disk encryption, *cold boot attacks* can retrieve unencrypted data from RAM. Cold boot attacks are based on the *remanence effect* of RAM which says that memory contents do not disappear immediately after power is cut, but that they fade gradually over time. This effect can be exploited by rebooting a running machine, or by transplanting its RAM chips into an analysis machine that reads out what is left in memory. In theory, this kind of attack is known since the 1990s. However, only in 2008, Halderman et al. have shown that cold boot attacks can be well deployed in practical scenarios. In the work in hand, we investigate the practicability of cold boot attacks. We verify the claims by Halderman et al. independently in a systematic fashion. For DDR1 and DDR2, we provide results from our experimental measurements that in large part agree with the original results. However, we also point out that we could not reproduce cold boot attacks against modern DDR3 chips. Our test set comprises 17 systems and system configurations, from which 5 are based on DDR3.**

***Keywords*-Cold Boot Attack, Remanence Effect, DDR3**

I. Introduction

Contrary to widespread belief, the contents of RAM are not instantly lost after power is cut but rather fade away gradually over time. Cold temperatures slow down the decay of bits in RAM further. This effect is called the *remanence effect* and was first described by Link and May in 1979 [1]. Since then, the remanence effect has been subject to security research several times [2], [3], [4], [5]. Theoretic attacks based on it were first proposed in the 1990s by Anderson and Kuhn [3], and were later described in detail by Gutmann [4] and Skorobogotov [5].

RAM can be classified into *dynamic* and *static* RAM (DRAM and SRAM). RAM modules in widespread PCs mostly use the DRAM technology because of its simplicity and low manufacturing cost. Gutmann explains in his work about data remanence in semiconductor devices [4] why DRAM exhibits the remanence effect. According to Gutmann, a DRAM chip consists of multiple DRAM cells where one cell stores the information of exactly one bit. Each cell consists of a capacitor. Each capacitor's voltage is compared to the load of a reference cell which stores a voltage half way between fully charged and fully empty. If the voltage of a cell is higher than the reference voltage the cell stores a one-bit; if it is lower it stores a zero-bit (or vise versa if the cell is an active low design). As the voltage of capacitors does not vanish instantly but rather decays exponentially, the bits are preserved for a short amount of time without power.

The memory controller refreshes each cell's voltage before it decays to the point where the bit information gets lost. Normally, the refresh rate is so high that each cell gets refreshed several times per second. However, the decay of capacitors is longer than the time between two refresh operations of the memory controller. This can be observed as the remanence effect, which lasts long enough that data in memory survives a system reboot. This observation prompted so-called *cold boot attacks*.

Cold boot attacks can make use of the property that the remanence effect is prolonged by cooling down RAM chips [5], [6]. Hence RAM modules are often cooled down or even frozen before those attacks. In the easiest form of a cold boot attack, the attacker reboots the system from USB thumb drive to start malicious system code that reads out what is left in memory. In a more advanced method, which becomes necessary when BIOS settings require a boot password or disallow to boot from USB, RAM modules can even be physically extracted in order to read them out in an analysis machine. In both cases, secret information, such as cryptographic keys for full disk encryption, can be retrieved from a computer's RAM that is running or suspended to RAM.

Despite the fact that the data remanence effect has been known for years, and that it has constantly been warned about, it was not until 2008 that Halderman et al. published the first practical attack based on the remanence effect in their work "Lest We Remember: Cold-Boot Attacks on Encryption Keys" [6]. Even though Halderman et al. have been cited by further research publications on the subject of cold boot attacks in subsequent years [7], [8], [9], [10], the practicability of this attack has never been verified independently, nor reconstructed by any of these publications, especially not for the modern DDR3 technology.

A. Contributions

In this work, we revisit the practicability of the memory extraction process described by Halderman et al.. We do not focus on the reconstruction problem of partially corrupted data, as done by many publications before, but challenge the attack process itself and investigate its reliability on old and new computer systems. In detail, our contributions are:

- We provide an independent study based on 12 computer systems with different hardware configurations that verifies the empirical practicability of cold boot attacks against DDR1 and DDR2. Despite deviations

978-0-7695-5008-4/13 $26.00 © 2013 IEEE
DOI 10.1109/ARES.2013.52

regarding temperatures and error rates, we could easily reproduce the results by Halderman et al., proving that cold boot attacks are indeed a serious threat. Besides the reboot variant, we reproduced the attack where cooled RAM chips are transplanted from one computer to another. We found that RAM transplantation is indeed possible and that in practice it provides reasonable low error rates so key reconstruction via known algorithms is possible.

- We provide empirical measurements showing the correlation between temperature and RAM remanence. The results of these measurements demonstrate that already cooling the surface temperature of a DDR1 or DDR2 module by just 10 °C can prolong the remanence effect notably. However, we also analyze up-to-date DDR3 RAM modules, which are not covered in the original work by Halderman et al.. This study is based on 5 different computer systems. While we demonstrate that simple warm reset attacks (not cutting power) are effective even against DDR3 systems, we were *not* able to detect any data remanence for DDR3 after cold boots. Even cooling the RAM chips did not reveal data remanence beyond cold boots. This leads us to the claim that cold boot attacks relying on RAM remanence beyond cold boots are not possible against modern DDR3 RAM chips.
- Last, we argue that all software-based countermeasures to the cold boot problem, which have been published since 2008, can be circumvented by transplanting the RAM modules from the victim's running computer to another computer controlled by the attacker. This makes cold boot attacks a rather generic attack on physical memory which can only be counteracted by physical security or new RAM chips that do not exhibit data remanence.

B. Related Work

As stated above, the first practical attack based on the remanence effect was described by Halderman et al. [6]. In their renowned paper from 2008, Halderman et al. have shown that cold boot attacks can be used to extract sensitive information, in particular cryptographic keys, from RAM. From extracted RAM cryptographic keys were reconstructed using recovery algorithms, and then used to break the full disk encryption of BitLocker, TrueCrypt, and FileVault.

In 2013, Müller and Spreitzenbarth performed cold boot attacks against smartphones for the first time [11]. To this end, they published a recovery tool called FROST which can be used to retrieve encryption keys from Android devices, thus proving that the ARM microarchitecture is also vulnerable to cold boot attacks.

While other publications, such as Wyns and Anderson [2] and Skorobogotov [5] also provide practical experiments on the RAM remanence, FROST and the work by Halderman et al. are to our knowledge the only publications featuring the practical memory extraction process of the cold boot attack itself. We contribute to this field by verifying the memory extraction process against laptops and desktop PCs, and by investigating the new DDR3 technology.

C. Outline

In Section II, we outline the test setup of our experiments, including the hardware and software configurations as well as the execution process of our experiments. In Section III, we present the results of our experiments, e.i. we present details about the temperature effects on RAM remanence. In Section IV, we outline how pure software-based countermeasures can be circumvented by different forms of the cold boot attack. And in Section V, we finally conclude our work.

II. Setup

In this section, we give an overview on our setup. We describe the hardware we used, the software we deployed, and our experimental setup.

A. Hardware

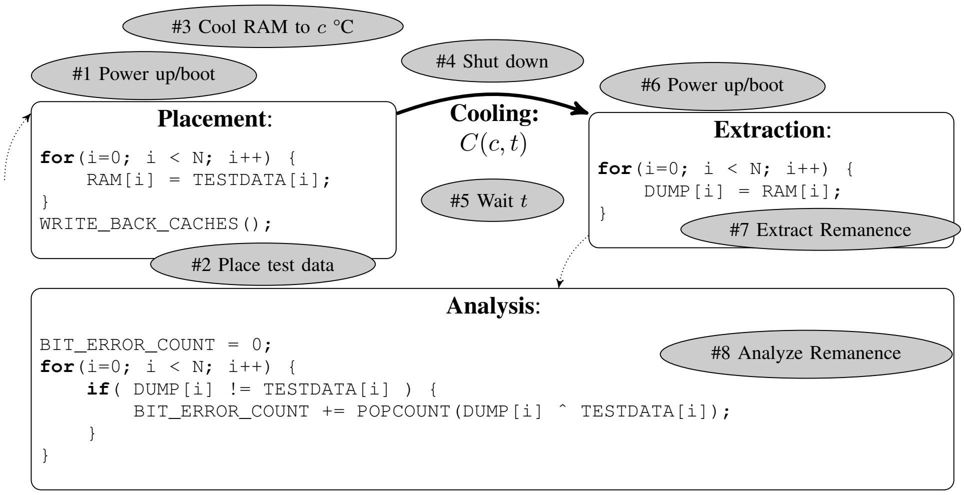
The hardware includes the computer systems we tested, as well as the equipment utilized in our experiments.

ID	System or Motherboard	DDR Type	RAM model	Size (MiB)
A	Asus Eee PC 1010H	2	HYMP512S64BP8-Y5	1024
B	Asus Eee PC 1010H	2	NT512T64UH8A1FN-3C	512
C	Asus Eee PC 1010H	2	04G00161765D *(GDDR2)*	1024
D	Asus Eee PC 1010H	2	KVR667D2S5/1G	1024
E	Asus Eee PC 1010H	2	CF-WMBA601G	1024
F	HP Compaq NX6325	2	HYMP512S64BP8-Y5	1024
G	HP Compaq NX6325	2	NT512T64UH8A1FN-3C	512
H	Intel Classmate PC NL2	2	HYMP512S64BP8-Y5	1024
I	Toughbook CF-19FGJ87AG	2	HYMP512S64CP8-Y5	1024
J	ASRock K8NF4G-SATA2	1	HYS64D64320GU-6-B	512
K	Fujitsu SCENIC P300 i845E	1	HYS64D64320GU-6-B	512
L	Fujitsu SCENIC D i845G	1	KVR400X64C3A/512	512
M	ASRock H77M-ITX	3	CMX8GX3M2A1600C9	8192
N	Fujitsu ESPRIMO P900 E90+	3	M378B5773CHO-CK0	2048
O	ASRock Z77 Pro4	3	HMT351U6BFR8C-H9	4096
P	Asus PBP67LE	3	M378B5773DHO-CH9	2048
Q	ASRock Z77 Pro3	3	KHX2133C8D3T1K2/4G	4096

Table I: List of tested computer systems and their corresponding RAM type and model.

Computer Systems: We focused on mobile devices such as laptops because due to their greater exposure to physical access by an attacker they represent likely targets of the cold boot attack. Table I gives a list of the systems we tested. All RAM chips were non-ECC SDRAM modules. Identical RAM model numbers mean the same RAM module was used in the corresponding systems. From now on, we refer to each system configuration by its respective identifier denoted as A to Q.

Thermometer: Temperature measurements were performed with a Sinometer DT8380 contactless infrared laser thermometer. Its measurement range is from -30 °C to 380 °C. It has a distance to spot size of 12:1, meaning measuring from a distance of 12cm the temperature within a spot of a 1cm diameter is measured. [12]



```
for(i=0; i < N; i++) {
    RAM[i] = TESTDATA[i];
}
WRITE_BACK_CACHES();
```

```
for(i=0; i < N; i++) {
    DUMP[i] = RAM[i];
}
```

```
BIT_ERROR_COUNT = 0;
for(i=0; i < N; i++) {
    if( DUMP[i] != TESTDATA[i] ) {
        BIT_ERROR_COUNT += POPCOUNT(DUMP[i] ^ TESTDATA[i]);
    }
}
```

Figure 1: Abstract setup of our experiments. Either a system is rebooted, or optionally its RAM modules are removed and transplanted into another system during step #5.

Cooling Agent: Cooling was provided by multiple cans of “KÄLTE 75 SUPER” spray from CRC Kontakt Chemie, which is a professional cooling agent especially suited for use with electronic components. It provides cooling power of 267J/ml and according to its specification the lowest attainable cooling temperature is -55 °C [13].

1) Test Data Placement: The test data consisted of a 2 MiB chunk of random data generated by a PRNG [14] and a 687 KiB 687 x 1024 pixel portable graymap (PGM) formated picture of the Mona Lisa. The random data was used as machine comparable data for error analysis, while the pictures were used for visual inspection. The test data was written starting from the physical address 8 MiB.

B. Software

We wrote two special-purpose bootloaders for our work, one for data placement and one for data extraction. Writing our own bootloaders saved us to boot into full OSs that potentially falsify our results. The data placement simulates content, such as full disk encryption keys that reside in a target’s RAM.

Our software to place the test data was a minimal bootloader containing the PRNG for the machine comparable data. It also contained the picture of the Mona Lisa in its static data. After the data was copied to the desired memory location the WBINVD instruction [15, WBINVD - Write Back and Invalidate Cache] was issued to force the data to be written to RAM instead of potentially remaining in CPU caches. The extraction bootloader extracts the remanence of the placed data, thus completing our cold boot attack. The extracted data is eventually analyzed.

The software we used for extraction is based on the USB SCRAPER tool by Bill Paul [16], which was also used for the experiments in the original cold boot attack publication [6].

The USB SCRAPER is a simple boot image that can be written to a bootable device, such as a USB stick. When booted, the USB SCRAPER copies the entire addressable memory contents of the system to the boot device. While this provides all functionality needed to perform cold boot attacks even against systems with several GiB of RAM, it was not very comfortable to use for our repeated measurements because dumping the entire memory takes a long time. Thus, the tool was modified to dump only the previous placed test data without wasting time on extracting the rest of the memory that is not relevant to our measurements.

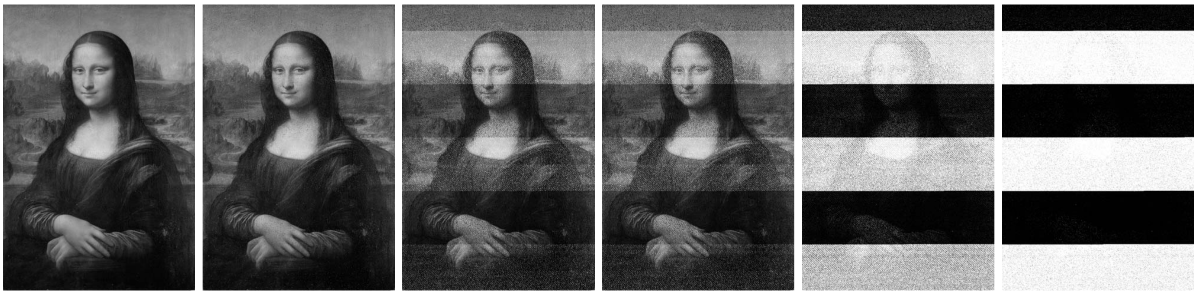
C. Experiment

Our experiments follow the procedure outlined as follows.

1) Structure: The canonical procedure of our cold boot attack consists of the 8 steps illustrated in Figure 1. Step #1 is to boot the system into the placer tool. The placer then, in step #2, copies the test data into the system’s RAM. Step #3 is to cool the RAM module down to the desired temperature c for the current measurement. Then, in step #4, the system is shutdown and in step #5, t seconds are waited. Depending on the kind of experiment, the RAM module may also be removed and transplanted into a different system during step #5. Afterwards, in step #6, the system containing the RAM is powered on again. In step #7, the booted extraction program dumps the remaining test data from RAM to disk. In the final step #8, the gathered remanence of the test data is compared with the clean undecayed test data, and the number of changed bits (i.e., bit errors) are counted.

Figure 1 contains pseudo code for the test data *placement*, the *extraction*, and the *analysis* on how many bits have decayed. The pseudo code uses the following variables and terminology:

- `TESTDATA`: test data on boot medium
- `DUMP`: RAM dump stored on boot medium
- `RAM`: physical RAM region being analyzed



(a) 0 sec / 100% **(b)** 2 sec / 99.2% **(c)** 3 sec / 93.4% **(d)** 4 sec / 93.1% **(e)** 5 sec / 61.4% **(f)** 6 sec / 51.9%

Figure 2: A picture of the Mona Lisa being recovered from system C's RAM at normal operational temperature of 20 to 25 °C after different amounts of time. Each picture's caption includes the percentage of *correct* bits that were recovered.

- `BIT_ERROR_COUNT`: number of bit errors
- `POPCOUNT`: hamming weight of a byte
- `WRITE_BACK_CACHES`: forces cached data to be written to RAM
- `N`: Number of bytes in test data
- $C(c,t)$: cooling to c °C and t seconds without power

2) Execution: We executed our experiments according to the following procedures.

Cooling: In step #3, the cooling agent is sprayed evenly over the entire surface of the RAM module so that all RAM chips are covered equally. If possible, both sides of the RAM module are sprayed. In case of the laptop systems A to I this was not possible due to the placement of RAM modules. After the initial cooling, no more cooling is applied so the RAM modules slowly warm up again during step #5. While this lack of cooling during step #5 may cause distortion in the decay curves of our measurements, applying constant cooling to maintain a constant temperature is practically not feasible. In real cold boot attacks, it is possible to reboot a machine to the point where the memory controller is refreshing the RAM modules again within 5 to 10 seconds. It is also possible to transplant RAM modules in this time. Hence measurements beyond that time are purely illustrators of the RAM remanence but irrelevant to real practical cold boot attacks anyway.

Temperature Measurements: The temperature is measured right after step #3 and before step #4. Because we measure with a contactless infrared laser thermometer, all measurements reflect surface temperatures. We measure from a distance of around 8 cm yielding a measuring spot 7 mm in diameter. This spot is small enough to encompass exactly one RAM chip. For the chip to measure, we choose the hottest chip of the RAM module. This chip is on all tested systems, without exception, in the lower row of the RAM module near the socket. The upper chips away from the socket are consistently several °C cooler.

RAM chips are cooled down while the system is running, meaning while the RAM chips still produce heat. The plastic surface of the RAM chips gives higher temperature readings than a label on the RAM module or its metal circuit parts. Therefore, our measured temperatures do not reflect the real core temperature of the RAM module but rather the upper bound of the surface temperature.

Our measurement procedure is probably the reason why we have only temperature down to 0 °C, even though the cooling agent can cool a disconnected RAM chip down to -30 °C (measured using the same procedure). However, to obtain more relevant results, the RAM is cooled in normal operational state.

In case the temperature measurement indicates that cooling is insufficient for the current experiment, cooling is reapplied and the temperature is measured again.

Timing Measurements: All timings mentioned are measured between step #4 and step #6. As an indicator of the events for step #4 and step #5, pressing the system's power button is used. While this does not reflect the true point where a RAM module is cut and reconnected to power, it is an established and reproducible point in time.

Analysis: The extracted test data is analyzed as follows: We first calculate the number of bit errors as outlined in the pseudo code of Figure 1. We then divide the error count by the number of total bits, in our case 2Mi. This gives use the percentage of correct bits. Note that, since we use random test data, it can happen with a statistical probability of 50% that a bit of our random test data is exactly the ground state of the bit in RAM and hence ”correct” by chance. That is, the minimum percentage of correct bits is 50%.

All measurements, especially the temperatures, are best efforts and not precisely accurate, as it is not possible to allocate the exact same amount of cooling agent every time. If for any of our experiments there was a deviation from the above mentioned procedure, it is indicated as such.

III. Results

We now present the results of our experiments. We first probe our test systems for RAM remanence. We then analyze the correlation between temperatures and RAM remanence. And last, we present results of our RAM transplantation experiments.

A. The Remanence Effect

Not all systems we tested exhibit RAM remanence. On some machines, RAM is reset even on warm resets, and even though any POST procedures are disabled and all fast and/or quick boot features are enabled in the system's BIOS, as advised by Halderman et al. [6, 3.4 BIOS footprints and

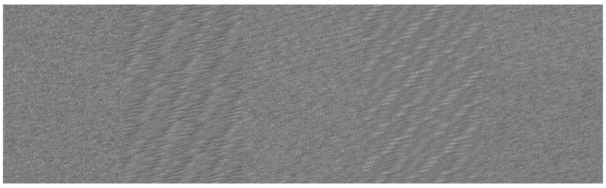
	RAM remanence observable after a		
	warm reset	cold reboot	
		without cooling	with cooling
A	Yes		
B	Yes	No	Yes
C	Yes		
D	Yes	No	Yes
E	Yes		
F	Yes		
G	Yes		
H	No (reset)		
I	No (reset)		
J	Yes		
K	No (reset)		
L	No (reset)		
M	Yes	No (noise with "signature" every 256 KiB)	
N	Yes	No (noise)	
O	Yes	No (noise)	
P	Yes	No (noise)	
Q	Yes	No (noise)	

Table II: List of observable RAM remanence in our test systems with different types of cold boot attacks.

memory wiping]. This fact was also observed by Chan et al. [7, Table 1]. Whether this memory reset is done as part of fullfilling the TCG Platform Reset Attack Mitigation Specification [17] or, as suspected by Halderman et al., as a quirk of ECC-capable systems to always bring the RAM to a known state whether or not ECC RAM is actually installed or not, remains an open question. Table II provides an overview over the state of observable RAM remanence in the various systems we tested with different types of cold boot attacks. Note the difference of the DDR3 systems M to Q, compared to the DDR1/DDR2 systems A to L.

Interestingly, all tested DDR3 systems maintain their entire RAM contents through warm resets, meaning that they are vulnerable to simple local warm reset attacks. However, after cold reboots, for all of them only noise patterns can be observed. These noise patterns are different each time and unrelated to the placed test data. See Figure 3 for noise patterns acquired on different systems. There exists one exception however: On system M, the first four bytes of every 256 KiB block always equal 0x5a on memory reset. This seemingly deliberately placed "signature" suggests explicit scrubbing of the memory. But since the memory is preserved on a warm reset such deliberate scrubbing is unlikely. We rather like to argue that these noise patterns are an inherent behavior of power cycling high density low voltage running DDR3 RAM. This claim is supported by the fact that all tested DDR3 systems exhibit the exact same behavior. But in either way we could not, even by excessively cooling them down to -10 °C, extract any of our placed test data. Contrary to that, for most DDR1 and DDR2 systems, we were able to do so.

Figure 2 is a representation of the RAM remanence in system C visualized as a sequence of Mona Lisa pictures. Figure 4 plots the time related bit decays of the systems A to G and J, exhibiting RAM remanence even after a cold reboot at their normal operational temperatures. The measurements at time $t = 0$ represent a warm reset. The first measurement above $t = 0$ represents the fastest time a system could be power cycled, i.e., powered down and up again. Note that even though A uses the same RAM module as F, and B the same as G, their curves differ. This can be explained as system F and G supply the fan and thus presumably also the memory controller with power for about 0.5 seconds after the power button is pressed. Thus, the actual time without power is less than presented in the graph. This shows that a simple cold reboot attack, which is impossible on system A and B, is possible on system F and G, even though those systems use the same RAM modules.



(a) M **(b)** N **(c)** O **(d)** P **(e)** Q

Figure 3: Noise patterns in DDR3 systems after a cold reboot.

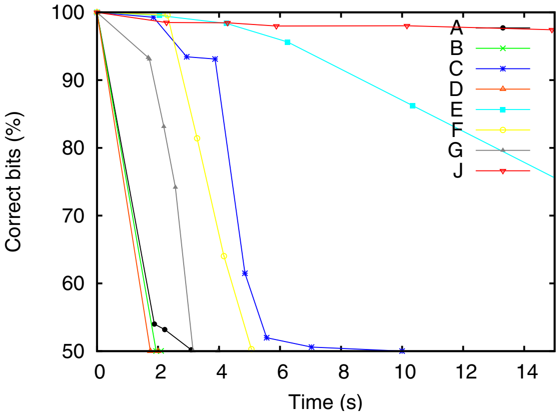


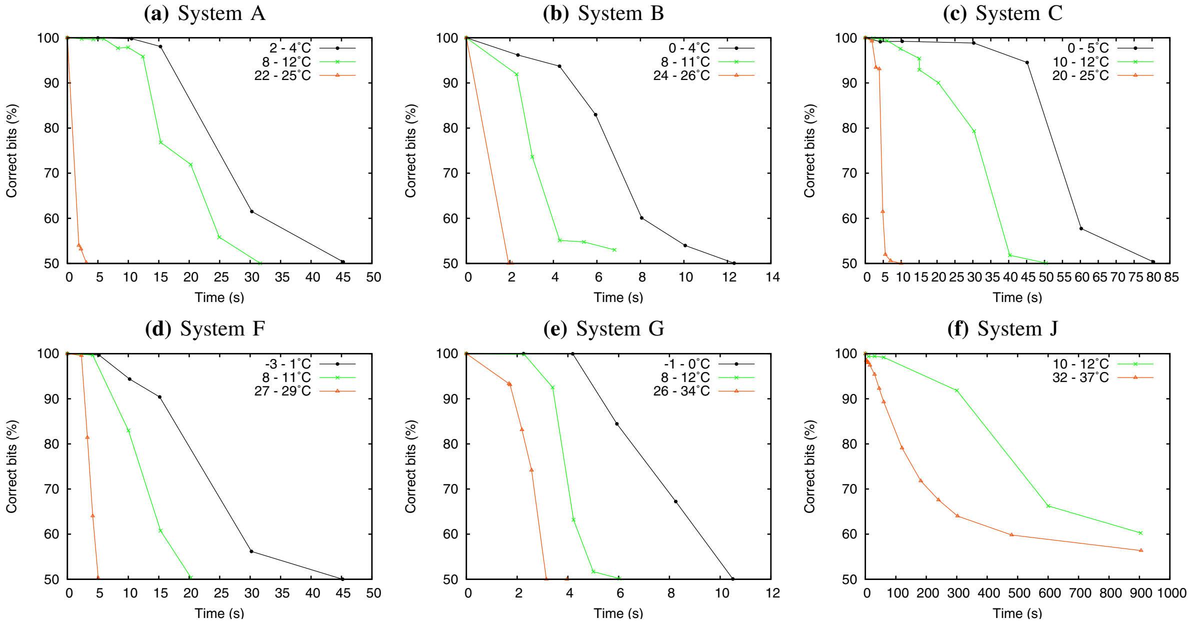
Figure 4: RAM remanence of systems A to G and system J.

We can conclude that cold boot attacks are feasible on most machines with DDR1 and DDR2, although the longevity of the RAM remanence varies considerably, as can be seen in Figure 4.

B. Temperature and RAM Remanence

We now analyzed the correlation of RAM remanence with the RAM temperature in more detail. To this end, we performed several cold boot attacks according to the experiment structure outlined in Section II-C, each with a different temperature c and a different time t. Again, the measurements at time $t = 0$ represent a warm reset and the shortest measurement above $t = 0$ represents the fastest time that systems can be power cycled.

Figure 5 shows plots about times and bit decays at different temperatures for systems A to E and H. These plots clearly show the correlation of lower temperatures and longer RAM remanence. Especially the legacy DDR1



(a) System A (b) System B (c) System C (d) System F (e) System G (f) System J

Figure 5: RAM remanence of systems A, B, F, G and J over time and at different temperatures. Note the different time scales. The highest temperature of each system's measurement is its normal operation temperature.

Temperature (°C)	Errors (bits)	Correct bits (%)
5.9	77645	99.537200
5.4	136120	99.188662
3.5	157994	99.058282
5.4	216734	98.708165
5.2	268176	98.401546

Table III: Temperature, time and error measurements for several cold boot attacks with RAM transplantation from system A to system B. The temperature was measured after the RAM was transplanted.

system J shows a remarkable long RAM remanence. The remanence of systems A and B, which at normal operational temperature barely exists, can be prolonged such that the systems can be power cycled and retain over 99% and 96% of their bit information. As mentioned before, the differences between system A and F, and the difference between B and G, which use the same RAM, can be attributed to the fact that systems F and G are supplying the memory controller with power for about 0.5 seconds beyond the power button being pushed, plus the fact that the RAM in systems F and G has a higher operational temperature than the RAM in systems A and B.

With these measurements, the temperature influence on RAM remanence can be clearly confirmed for our DDR1 and DDR2 test systems. Even a modest surface temperature drop of only 10 °C prolongs remanence.

C. RAM Transplantation Attacks

As is evident from the previous section, RAM remanence is long enough, or can be prolonged long enough via cooling, that there is enough time for a RAM module to be transplanted from one system to another without loosing the majority of its content. In practice, we were, able to transplant the RAM from system A to system B. For this, we cooled the RAM module in system A, quickly removed the RAM module from the running system, and then inserted it into system B. System B was subsequently booted and the RAM remanence extracted. We could in various attempts consistently recover over 98% of all bits correctly. Our attempts became increasingly better and our last attempts even surpassed 99%. Table III gives a list with the figures for our attempts.

Even a seemingly failed attempt (the power button of system B was missed) resulted in 95% of all bits being transferred correctly. Considering the findings of Halderman et al. where they reconstructed an AES key within seconds if 7% of the bits are decayed [6, 5 Key Reconstruction], or Heninger and Shacham showing their ability to efficiently reconstruct an RSA private key with small public exponent from only 27% of the bits [8], still renders our ”failed” attempt a success. Cold boot key recovery for other ciphers such as Serpent and Twofish, as published by Albrecht and Cid [10], can also tolerate higher bit errors than what we exhibited. Hence RAM transplantation is a feasible attack scenario‘.

In another experiment, we successfully transplanted the RAM from system H, which according to Table II resets its entire memory upon boot, into system A. This way we could efficiently circumvent whatever mechanism causes the memory into system H to be reset. This can also be used to circumvent BIOS locks as we stress in the next section.

However, RAM transplantation is not always possible. It requires two systems with compatible memory controllers, e.g., trying to transplant from system K to J, A to F, F to A, I to A, I to F, and H to F, all failed.

IV. Bypassing Software Countermeasures

We now take a look at various software-based countermeasures to the cold boot attack that came up since 2008, and discuss how those can be circumvented. Some of the presented countermeasures are already present, others have only been theoretical proposals.

A. RAM Reset on Boot

Clearing RAM on boot, as mandated in the TCG Platform Reset Attack Mitigation Specification [17], can stop a straightforward reboot attack. However, it cannot stop an advanced transplantation attack as presented in the previous section. This technique is nonetheless a recommended mitigation technique that should be deployed as it indeed raises the difficulty of cold boot attacks.

B. Locking the Boot Process

Locking the boot process, e.g., with a BIOS master password, does also not prevent the advanced transplantation attack, because an attacker can still transplant the RAM into a system with full control over the boot process. But like the RAM reset on boot, also this practice is encouraged as it at least complicates cold boot attacks.

C. Temperature Detection

A proposed countermeasure [18] is the usage of temperature sensors which in case a sudden temperature drop is detected initialize a memory wiping process. However, we found that this method again only makes the cold boot attack harder but does not prevent it. To simulate an attack that circumvents temperature sensors, we used system A as the victim and system B as the attacker. We powered system B up without RAM. This obviously causes the boot to fail and leaves the system in an unresponsive failure state but the system's memory controller is still fully functional and refreshes any RAM module inserted. The RAM module is very quickly, within under a second, removed from running system A and inserted into system B. As this transfer is done without cooling no temperature sensors are triggered. Once the RAM is in system B, it is refreshed by the memory controller immediately. To finish the attack we need to perform a normal cold boot attack on system B, this brings it back to normal operational state and boots it into our extraction program. Since the RAM is already out of control of the temperature triggering system this does not pose a problem.

Temperature (°C)	Errors (bits)	Correct bits (%)
6.8	127406	99.240601
8.0	1217038	92.745888
10.7	1749820	89.570260
9.9	4408509	73.723239

Table IV: Temperatures and bit errors for several transplantations from system A to system B without cooling. In the attempt with 73% correct bits the RAM socket of system B was accidentally missed causing more loss. The temperature given is from cold booting system B.

Because the RAM transfer has to be performed without cooling it causes considerable RAM decay, even though it can be performed in under one second. Nevertheless, we were able to extract 90% of the bits correctly, even reaching 99% in one attempt. However the attack is very fragile and the slightest mishaps while performing the non-cooled transfer results in severe data loss. In one such instance only 73% of the bits could be captured correctly. Table IV gives an overview over the stats of various attempts. As detailed in Section III-C, available research on key reconstruct says that, despite these elevated error rates, encryption keys can be reconstructed.

Even though we demonstrate how temperature sensors can be circumvented, they pose a further obstacle because they make a phase of uncooled decay mandatory.

D. 0x7c00 Defense

The 0x7c00 (or boot block) defense method is another proposed countermeasure [18]. On the x86 architecture, 0x7c00 is the memory address to which an IBM-PC compatible BIOS loads the boot device's master boot record (MBR), i.e., the first 512 bytes of a bootable device [19]. The theory behind the 0x7c00 defense is to place sensitive data, such as encryption keys, into this 512 bytes at 0x7c00, so that any reboot will overwrite them. However the RAM module can be transplanted into a system with two memory slots with the slot holding the lower address space in which 0x7c00 resides being filled with a dummy RAM module, and the upper slot with the victim's RAM module. So again, this countermeasure complicates the attack, but it does not prevent it entirely. Besides that it only offers protection for 512 bytes.

E. Outlook

As demonstrated, all proposed software solutions to the cold boot problem can be circumvented with an adaption of the attack, because once RAM modules are removed from a system there is not way for the system to react upon the event of removal. Hence, although some solutions provide a certain level of mitigation, pure software solutions cannot be entirely effective. The only way to avoid cold boot attacks seems to be physical security, or to build upon RAM chips that are less affected by remanence.

When preventing an attacker to access a running and/or suspended system with sensitive data in RAM, or by preventing RAM transplantation, cold boot attacks become impossible. However, this requires some kind of physical protection. For example, the first can be achieved by always turning the system off and never just locking it or leaving it in suspend, and the latter can be achieved by soldering RAM directly onto the system's motherboard, as done in smartphones and tablets today. But most notably, according to our tests, a promising countermeasure may be to switch from DDR1 and DDR2 modules to DDR3 in future as we were not able to accomplish anything more than a warm reset attack against DDR3 rendering all circumvention techniques presented in this section inapplicable.

V. Conclusions

Our work provides an independent study on the practicability of cold boot attacks. We systematically recreated the practical RAM extraction procedure as presented by Halderman et al.. Our empirical measurements on DDR1 and DDR2 showed the correlation between temperatures and RAM remanence, demonstrating that even minor cooling of the surface temperature of a RAM module by just 10 °C prolongs the remanence effect notably. By providing profound documentation on our experiments, a detail that is missing in the publication by Halderman et al., other researches can better match and compare their findings to ours.

Elevating the attack to currently used DDR3 RAM however failed, and we were not able to accomplish any attack more advanced than the basic warm reset attack on DDR3 RAM (which can be prevented with a simple BIOS boot lock). Admittedly, it is too early to promote DDR3 as the ultimate countermeasure against cold boot attacks, and further experiments are required in future. For example, it stays unclear whether the DDR3 construction type alone renders the remanence effect unobservable, presumably due to very short remanence times caused by the lower voltage used, the higher integration density and the resulting lower charges in the RAM cells, or if the DDR3 memory controller plays a role as well. If the latter is the case, specialized DDR3 controllers on an attacker's machine could re-enable cold boot attacks again.

References

[1] W. Link and H. May, *Eigenshaften von MOS-Ein-Transistorspeicherzellen bei tieften Temperaturen*, 1979.

[2] P. Wyns and R. L. Anderson, "Low-temperature operation of silicon dynamic random-access memories," *Electron Devices, IEEE Transactions on*, vol. 36, no. 8, pp. 1423–1428, Aug. 1989.

[3] R. Anderson and M. Kuhn, "Tamper Resistance: A Cautionary Note," in *Proceedings of the 2nd conference on Proceedings of the Second USENIX Workshop on Electronic Commerce - Volume 2*, ser. WOEC'96. Oakland, CA: USENIX Association, 1996, pp. 1–1.

[4] P. Gutmann, "Data remanence in semiconductor devices," in *Proceedings of the 10th conference on USENIX Security Symposium - Volume 10*, ser. SSYM'01. Berkeley, CA, USA: USENIX Association, 2001, pp. 4–4. [Online]. Available: http://dl.acm.org/citation.cfm?id=1251327.1251331

[5] S. Skorobogatov, "Low temperature data remanence in static RAM," University of Cambridge, Computer Laboratory, Tech. Rep. UCAM-CL-TR-536, Jun. 2002. [Online]. Available: http://www.cl.cam.ac.uk/techreports/UCAM-CL-TR-536.pdf

[6] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Cal, A. J. Feldman, and E. W. Felten, "Least we remember: Cold boot attacks on encryption keys," in *In USENIX Security Symposium*, 2008.

[7] E. M. Chan, J. C. Carlyle, F. M. David, R. Farivar, and R. H. Campbell, "Bootjacker: Compromising computers using forced restarts," in *In Proceedings of the 15th ACM Conference on Computer and Communications Security*, 2008, pp. 555–564.

[8] N. Heninger and H. Shacham, "Reconstructing RSA Private Keys from Random Key Bits," in *Advances in Cryptology - CRYPTO 2009*, ser. Lecture Notes in Computer Science, S. Halevi, Ed. Springer Berlin Heidelberg, 2009, vol. 5677, pp. 1–17. [Online]. Available: http://dx.doi.org/10.1007/978-3-642-03356-8_1

[9] E. Chan, S. Venkataraman, F. David, A. Chaugule, and R. Campbell, "Forenscope: a framework for live forensics," in *Proceedings of the 26th Annual Computer Security Applications Conference*, ser. ACSAC '10. New York, NY, USA: ACM, 2010, pp. 307–316. [Online]. Available: http://doi.acm.org/10.1145/1920261.1920307

[10] M. Albrecht and C. Cid, "Cold boot key recovery by solving polynomial systems with noise," in *Proceedings of the 9th international conference on Applied cryptography and network security*, ser. ACNS'11. Berlin, Heidelberg: Springer-Verlag, 2011, pp. 57–72. [Online]. Available: http://dl.acm.org/citation.cfm?id=2025968.2025974

[11] T. Müller and M. Spreitzenbarth, "FROST: Forensic Recovery Of Scrambled Telephones — IT-Sicherheitsinfrastrukturen (Informatik 1)," Website: https://www1.informatik.uni-erlangen.de/frost (Last accessed: 21 February 2013), 2012.

[12] Sinometer Instruments, "DT8380, DT8550 - INFRARED THERMOMETERS," Datasheet: http://www.sinometer.com/pdf/DT8380,\%208550.pdf (Last accessed: 21 February 2013), 2003.

[13] CRC Industries Deutschland GmbH, "TECHNISCHES MERKBLATT - KÄLTE 75 SUPER, Ref.: 20848," Datasheet: http://www.crcind.com/wwwcrc/tds/TKC4\%20FREEZE75S.PDF (Last accessed: 21 February 2013), 2003.

[14] P. L'Ecuyer, "Tables of Maximally-Equidistributed Combined Lfsr Generators," 1998. [Online]. Available: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.43.3639

[15] Intel Corporation, "Intel 64 and IA-32 Architectures Software Developer's Manual Volume 2 (2A, 2B & 2C): Instruction Set Reference, A-Z," Manual: http://download.intel.com/products/processor/manual/325383.pdf (Last accessed: 1 February 2013), 2012.

[16] J. A. Halderman, S. D. Schoen, N. Heninger, W. Clarkson, W. Paul, J. A. Cal, A. J. Feldman, and E. W. Felten, "Memory Research Project Source Code, Center for Information Technology Policy at Princeton," Website: https://citp.princeton.edu/research/memory/code/ (Last accessed: 21 February 2013), 2008.

[17] Trusted Computing Group, Incorporated, "TCG Platform Reset Attack Mitigation Specification, Specification Version 1.00, Revision 1.00," Specification: https://www.trustedcomputinggroup.org/resources/pc_client_work_group_platform_reset_attack_mitigation_specification_version_10/ (Last accessed: 27 February 2013), 2008.

[18] P. McGregor, T. Hollebeek, A. Volynkin, and M. White, "Braving the Cold: New Methods for Preventing Cold Boot Attacks on Encryption Keys," in *Black Hat Security Conference*. BitArmor Systems, Inc., aug 2008.

[19] A. S.Tanenbaum, *OPERATING SYSTEMS: Design and Implementation*. Prentice-Hall, 1987.