Epistemology of Rowhammer Attacks: Threats to Rowhammer Research Validity

Martin Heckel[1,2], Hannes Weissteiner[2], Florian Adamsky[1], and Daniel Gruss[2]

[1] Hof University of Applied Sciences, Hof, Germany
[2] Graz University of Technology, Graz, Austria

Abstract. The Rowhammer effect is a disturbance error in DRAM that attackers can trigger from software. The first publication on Rowhammer in 2014 evaluated 129 Dual In-line Memory Modules (DIMMs) on an FPGA and showed that 110 DIMMs are affected, indicating that Rowhammer is a widespread issue. However, until now, no case outside of academia is known in which Rowhammer was used for attacks, indicating a stark discrepancy between the attention Rowhammer receives and its real-world relevance.
This paper systematically analyzes 32 offensive Rowhammer papers, including 48 experiments. However, we avoid finger-pointing but identify six threats to the validity and relevance of Rowhammer research results and give multiple examples. The threats include small sample sizes, overestimated attacker capabilities, unrealistic attack scenarios, non-comparability of the results, age and wear of hardware, and sub-optimal attack performance metrics. Additionally, we provide recommendations with detailed justification to the scientific community to mitigate those threats: (1) pre-experimental testing of DIMM integrity, (2) increasing and broadening the DIMM sample size, (3) expanding reproduction studies of published work, (4) defining attacks in real-world conditions and distinguishing them from theoretical ones, (5) publishing DIMM manufacturing data, (6) documenting DIMM wear and, (7) leveraging multiple metrics for bit flip evaluations.

1 Introduction

The main memory, Dynamic Random-Access Memory (DRAM), remains crucial in all computer devices. The demand for higher storage capacity yields a high density of DRAM memory cells. However, the industry has reached a point where scaling becomes a problem. Scaling the capacitors and transistors beyond 40 nm is challenging [32] and can result in disturbance errors.

These disturbance errors were initially assumed to have little to no security implications [50]. Later, Kim et al. [28] showed that an attacker can trigger bit flips in DRAM rows by reading from nearby rows rapidly, which is known as *Rowhammer*. In recent years, researchers developed sophisticated exploits based on Rowhammer. These exploits achieve, for instance, privilege escalation on desktop computers [51, 13, 47, 17, 1, 54, 12, 9, 31, 18, 49, 57, 8, 36, 26, 19, 25],

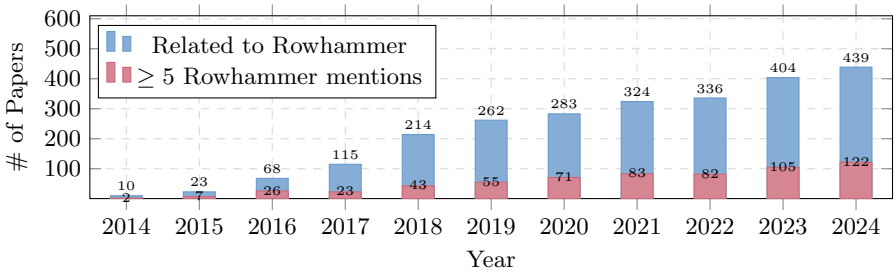

Fig. 1. Number of published research papers related to Rowhammer per year, and number of papers that mention rowhammer multiple times. We analyzed 2 509 papers identified by a Google Scholar search and counted the number of occurrences of the word "Rowhammer". If a paper has ≥ 5 occurrences of that word, we count it as a Rowhammer paper. This metric might include some paper that focus on another topic, but still provides an estimation of the number of publications related to Rowhammer.

mobile devices [58, 62, 9, 34, 30], and even on cloud systems [48, 4, 60, 55], all without a software vulnerability. Over the years, the number of scientific papers related to Rowhammer[3] increased, as shown in Figure 1.

With so many scientific publications, system administrators ask: *Should we integrate Rowhammer into our threat analysis?* However, to the best of our knowledge, Rowhammer has not been used in real-world attacks, such as malware or ransomware. It might be unrealistic to see malware or ransomware based on Rowhammer, but *we don't know if Rowhammer would be an attack vector usable for such attacks*. National or state actors could use Rowhammer as part of their attack chain. Overall, the lack of real-world attacks contradicts the number of Rowhammer publications from academia. There is a stark discrepancy between the attention Rowhammer's research has in the academic community and the relevance of Rowhammer in real-world attacks.

In this paper, we show multiple threats to Rowhammer research validity and discuss their influence on the overall validity of Rowhammer research. We analyze 32 publications that perform 48 experiments regarding these threats and show how relevant these threats are regarding these publications. We focus on offensive Rowhammer research since these publications typically perform experiments on how good attacks work on specific systems, resulting in the difference between academic and real-world estimation of exploitability. Finally, we show how researchers can prevent those threats in future research.

We point out cases where specific threats undermined the validity of previous work's experimental evaluation. *Identifying these threats would not have been possible without the tremendous effort put into these prior works.* We crucially build upon them for identification and do not want to point fingers at previous work. Instead, we want to provide recommendations to improve the validity of Rowhammer's research in general for future work.

[3] The results for the keyword "hammer" were almost identical.

We identify the following threats to Rowhammer research validity:

$\mathcal{T}$1 Small Sample Sizes. Most of the publications related to Rowhammer use a small sample size for their experimental evaluation, sometimes only a sample size of 1. Small sample sizes are insufficient to show that an attack works in general, and it raises the question of the prevalence of Rowhammer. An attack might only work under specific conditions or not work and yield results because the Dual In-line Memory Module (DIMM) is not functioning correctly, e. g., Target Row Refresh (TRR) is not working properly.

$\mathcal{T}$2 Overestimated Attacker Capabilities. The assumption of unrealistic capabilities of attackers leads to an overestimation of the impact of an attack. Many attacks require specific preconditions, e. g., elevated privileges to get the physical addresses mapped to virtual addresses or access to 1 GB hugepages, etc. Some preconditions render an attack ineffective for practical exploitation, e. g., requiring elevated privileges to perform a privilege escalation attack.

$\mathcal{T}$3 Unrealistic Attack Scenarios. In some publications, the authors use special hardware like FPGAs to have fine control over the DRAM commands or overclock DIMMs in the BIOS. These are unrealistic attack scenarios, and it is unclear if such attacks would work in a real-world scenario.

$\mathcal{T}$4 Results are not comparable with other Publications. The susceptibility of systems to Rowhammer depends strongly on the system itself and environmental parameters. For example, Orosa et al. [44] showed that the number of bit flips triggered depends on the temperature. Due to the lack of specifying and monitoring environmental parameters and the fact that each research group uses different systems for experimental evaluation, the results of multiple experimental evaluations are not comparable.

$\mathcal{T}$5 Age and Wear of Hardware not specified. There are indicators that Rowhammer bit flips may "burn in", similar to the malicious aging of circuits [27]. Thus, when a specific bit flip is triggered many times in DRAM, the number of activations to trigger the bit flip can decrease. TRR, a proprietary Rowhammer mitigation, might have to be adjusted over time to mitigate new patterns or improve performance. Therefore, the age of a DIMM is relevant information for estimating specific properties of a DIMM.

$\mathcal{T}$6 The Number of Bit flips is a bad Comparison Metric. Typically, the number of bit flips is used as a comparison metric. However, it strongly depends on the system used for experimental evaluation. Therefore, it is impossible to compare the effectiveness of different existing approaches without repeating them in the same setup. Additionally, this metric does not provide any information regarding exploitability.

Contributions. Our work makes the following contributions:

1. We perform a meta-analysis and evaluate potential threats to Rowhammer research validity using 32 publications that performed 48 experiments.
2. We identify six threats to the validity of Rowhammer research and provide a detailed justification.
3. We identify 8 recommendations to our community that help mitigate threats to validity in Rowhammer research.

Outline. Section 2 provides background. Section 3 overviews threats to Rowhammer research validity. Section 5 analyzes sample sizes of prior work, Section 6 analyzes attack scenarios, Section 7 analyzes empirical results and comparability, and Section 8 discusses the influence of aging and wear. Section 9 analyzes comparison metrics from prior works. Section 10 concludes.

2 Background and Related Work

This section provides background on DRAM, Rowhammer, and related work.

DRAM. In DRAM, data is stored in cells consisting of capacitors and transistors, organized in an array of rows and columns. A *wordline* connects all transistors in a row, i. e., all cells in a row are accessed at once. The charge from the cells is amplified and forwarded to the *row buffer* (either SRAM or a feedback loop of the *bitlines*). Reading a cell drains the capacitor's charge, i. e., the row buffer has to be written back to the DRAM array before another row can be loaded. The memory controller must periodically refresh capacitors that lose charge over time. DDR3 [20] and DDR4 [21] use a refresh interval of 64 ms for each cell, and DDR5 [22] uses 32 ms, i. e., refresh commands must be issued for each row within this interval. Refreshes are typically performed in batches [11].

DRAM banks are located on multiple DRAM chips, and are organized in *ranks*, with one or more ranks on a DIMM. DIMMs are connected to the CPU with buses called *channels*.

DRAM Addressing. The kernel maps virtual to physical addresses using page tables. Physical addresses are mapped to different devices and their spatial components by the memory controller. For DRAM, the memory controller determines, e. g., channel, DIMM, rank, bank, row, and column, using *DRAM addressing functions*. These functions can be linear, essentially an XOR combination of physical address bits, or non-linear. Addressing functions were published for some models [2, 16] but not recent ones.

Reverse-engineering linear DRAM addressing functions has been demonstrated using, e. g., timing [45, 59, 9, 14, 19] and performance counters [15]. However, non-linear DRAM addressing functions remain a challenge.

Rowhammer. When two rows in the same DRAM bank are accessed alternatingly, they are loaded into the row buffer and written back every time they are accessed, incurring numerous accesses the DRAM array. A high number of accesses to the DRAM array can lead to disturbance errors, typically in spatially nearby cells [28], called *Rowhammer*. If DRAM cells leak enough charge, their value is inverted at the sense amplifier. These bit flips have to happen before the next refresh, as cell charge is restored at refresh, i. e., fully charged or discharged. The accessed rows are called *aggressor rows*, and the rows likely to have bit flips afterward are called *victim rows*. Initially, different patterns like Single-Sided [28], Double-Sided [13], or One-Location [12] were used. Newer approaches [9, 18, 19] fuzz these patterns to bypass TRR.

In 2014, Kim et al. [28] published the first scientific analysis of Rowhammer. They showed that 110 of the 129 DIMMs in their FPGA-based setup are affected

by Rowhammer. They also demonstrated bit flips on one Intel Sandy Bridge, Ivy Bridge, Haswell, and one AMD Piledriver system using one 2 GB DDR3 DIMM.

In 2015, Seaborn and Dullien [52] presented two Rowhammer exploits: A NaCl sandbox escape and a local privilege escalation based on flipping bits in page-table entries (PTEs). One year later, Razavi et al. [48] showed that Rowhammer can be exploited in a cross-VM scenario.

In 2014, vendors started to deploy mitigations against Rowhammer [6]. One of the first approaches was to double the refresh rate, as suggested by Kim et al. [28]. However, they already reported that lowering the refresh interval from 64 ms to 8.2 ms may degrade performance by 11 % to 35 %.

Another approach is to use Error Correction Code (ECC) DRAM to correct bit flips. However, Cojocar et al. [7] showed that even ECC does not prevent Rowhammer when high numbers of bit flips occur. Later, vendors introduced Target Row Refresh (TRR), a mechanism that tracks DRAM accesses and refreshes potential victim rows between regular refreshes. TRR implementations are proprietary and adjusted for new DIMMs when new attacks are published. Still, multiple publications bypassed TRR [12, 9, 18, 19]. However, there are multiple other approaches for mitigations: Some are based on counting activations [43, 3], and some on the location of rows in DRAM [5], some on cryptographic checksums [24].

Related Work. Mutlu and Kim [39] were the first to provide a retrospective of Rowhammer attacks and defenses. They surveyed the existing research papers at that time and discussed them in detail. Additionally, they focused on their previously proposed hardware mitigation PARA [28]. They also discussed Rowhammer attacks on other memory technologies, such as NAND flash.

Loughlin et al. [35] created a taxonomy for existing mitigations and proposed a memory controller extension against future attacks. They also described the limitations of countermeasures and argued that there is a disconnect between existing hardware and proposed software mitigations from the community. On the meta-level, they suggested DRAM vendors should publish precise information about their defenses to help build more effective mitigations.

Naseredini [40] surveyed of Rowhammer attacks and defenses, categorizing research into attack techniques and mitigation strategies. He analyzed them year by year and created an overview of different approaches over the years.

Recently, Zhang et al. [63] systematized Rowhammer attacks and defenses on commodity systems. They establish a unified framework to analyze Rowhammer attacks, grouping them by origins, methodologies, and objectives. They also classify various defense mechanisms including ECC and TRR.

These works provide an excellent overview of Rowhammer but do not systemically analyze problems in the research methodology that threaten validity.

3 Threats to Rowhammer Research Validity

In this section, we describe six threats to Rowhammer research validity. We identify potential problems and propose mitigations to establish a rigorous sci-

entific process for future Rowhammer research, based on a representative set of Rowhammer publications. The results of these high-quality, peer-reviewed publications led to the insights and recommendations presented in this paper.

3.1 $\mathcal{T}1$ Sample Sizes is Too Small

With small sample sizes, deriving general claims in empirical settings is impossible. Testing a Rowhammer attack on a single DIMM shows an attack is theoretically possible. However, a DIMM is a complex piece of electronics. Multiple potential causes exist for bit flips [38]:

- **Bad memory cells** can introduce random bit-flips.
- **Temperature** outside the operating range can impact reliability.
- **Cosmic rays** can hit DIMMs, yielding completely random bit flips.
- **Voltage fluctuations** by the power supply can introduce faults.
- **Manufacturing variations** can make a DIMM more vulnerable.
- **Electrical properties** of the motherboard (e. g., path length differences, impedance issues, or faulty contacts) can affect reliability.

Some attacks may only work due to undocumented preconditions or faulty hardware. These attacks are not reproducible, reducing trust in their validity. To reduce the influence of these factors, a higher sample size is required, ideally using different test systems. Additionally, higher sample sizes allow for the estimation of the prevalence of Rowhammer, i. e., the fraction of affected DIMMs.

A reasonable estimation of the prevalence of Rowhammer is essential: If the estimate is too low, Rowhammer research may become underrepresented despite of it's high impact. If the estimate is too high, too much effort might be put into solving a problem that only has little real-world implications.

3.2 $\mathcal{T}2$ Dependence on Elevated Attacker Privileges

In 2015, Seaborn and Dullien [52] demonstrated two exploits based on Rowhammer: A NaCl sandbox escape and a local privilege escalation based on PTEs. Consequently, obtaining virtual-to-physical address mappings was made privileged [29]. In newer attacks, other concepts like uncached memory [30], Transparent Hugepages (THPs) [48, 26], or 1 GB Hugepages [18, 19] were used. Many exploitation techniques from prior work rely on very particular prerequisites and *have been mitigated as a reaction to the publication of these techniques* by changing default configurations or requiring elevated privileges for vulnerable interfaces. Therefore, most systems with default configurations do not meet these prerequisites anymore. Elevated attacker privileges make the attack more difficult to reproduce and may decrease trust in the empirical results. As a result, Rowhammer research may become a niche area where findings are only relevant to other Rowhammer studies and lack broader implications.

3.3 $\mathcal{T}3$ Uncertain Practical Applicability

Another threat to the validity of Rowhammer research is the uncertain practical applicability of results on off-the-shelf hardware. Some experimental evaluations

of Rowhammer attacks are performed on specialized hardware, e. g., FPGAs, with the advantage of fine-grained control over DRAM commands.

Additionally, some Rowhammer attacks work on commodity hardware, yet require extreme parameters for DRAM operation, e. g., extreme overclocking. Thus, these attacks require physical access and control over firmware settings.

Rowhammer simulators like Hammertime [53] and Hammulator [56] enable faster development of Rowhammer attacks and defenses by providing faster and more deterministic bit flips. However, while this enables better comparability of different Rowhammer attacks, it has the disadvantage of not being a real system. Emulators provide good metrics for comparisons, but replacing experimental evaluation with simulators might increase the difference between academic results and the real-world exploitability of Rowhammer. Such research is essential for understanding the Rowhammer effect. However, such foundational research cannot be directly applied to real-world attacks. Follow-up work is needed.

3.4 $\mathcal{T}4$ Comparability across Publications

The position and number of bit flips during a Rowhammer attack depend on environmental parameters such as temperature [44]. Additionally, they rely on the systems and DIMMs that are evaluated. Thus, directly comparing different approaches is impossible, as most publications use different setups.

In some publications, the experimental setups are not described sufficiently. For example, CPU models, DIMM model numbers, Kernel versions, etc. are often missing. The memory controller is directly integrated into the CPU. Thus, different CPU models may have different memory access behaviors. Other kernel versions may influence the attack. For example, the change in the permission of `/proc/pid/pagemap` [29] made the attacks more difficult, as users cannot obtain physical memory addresses. Thus, due to undocumented hard- and software, experiments are often not reproducible anymore.

The physical environment is often not documented, e. g., the temperature of the DIMMs depends on whether the test system is in an office environment or a climate-controlled server room. Therefore, the environmental effects that affect experimental results are unknown, making it hard to compare them.

Another problem is that different DIMMs, even if they are the same model, are affected differently by Rowhammer [33]. While one DIMM might yield a high number of bit flips, another DIMM of the same model might not be susceptible to Rowhammer at all. This diversity makes results hard to reproduce and hinders comparing novel and existing attacks.

3.5 $\mathcal{T}5$ Unspecified Age and Wear of Hardware

Typically, the DIMMs used in experimental setups are not documented. Scientific papers aim for general applicability rather than singling out specific manufacturers, but documentation of the used hardware is essential for reproducibility. Additionally, aging generally affects circuits and their reliability, and Karimi et

al. [27] showed that this can be sped up maliciously. Thus, the DIMMs' manufacturing date and wear are crucial to contextualizing the experimental evaluation.

In DRAM, bit flips induced by Rowhammer can "*burn in*" [27], i. e., they can become more likely when triggered many times. Consequently, the susceptibility of DIMMs used for Rowhammer research has increased over the years. Typically, the usage in prior Rowhammer experiments is not documented for the DIMMs used in the experimental evaluation. Therefore, it is hard to compare the effectiveness of attacks between experiments on different DIMMs.

The algorithms used for Rowhammer attack detection in TRR are not specified, most likely differ between manufacturers and even DIMM models. Therefore, effectiveness of TRR depends on the specific model or even manufacturing date of a DIMM. Vendors may adjust TRR to mitigate published attacks in newer DIMM generations. However, when the DIMM model is unknown, it is impossible to estimate which specific attacks are mitigated by TRR.

Age and usage of a DIMM may affect the Rowhammer susceptibility and, thus, they may influence the results of empirical evaluations. However, both parameters are usually not documented, increasing the difficulty of reproducing results. In addition, it decreases the comparability of publications. Both effects might reduce the trust in experimental results.

3.6 $\mathcal{T}6$ Suboptimal Metrics for Comparison

In the current research, the susceptibility of a system to Rowhammer is often expressed in the number of bit flips found in a given time or memory area. However, these metrics are not standardized. For example, Kang et al. [26] used bit flips per hour. Other work [41] used minimal activations until the first bit flips occurred, which is also known as *hammer count*. In contrast, Jattke et al. [18] used multiple measurements, including a total bit flips found in a given time and total number of bit flips over a sweep[4] of 256 MiB. Ridder et al. [49] used the percentage of times they observed bit flips at a vulnerable location. Thus, the metrics presented in different publications can not be used to compare the performance of attacks across publications. Therefore, to compare a novel attack to existing work, researchers must reproduce the prior attack on their hardware with their measurements. Due to the limited reproducibility of Rowhammer attacks, this is an unrealistic demand. Thus, the number of unique exploitable bit flips can be a better metric to estimate the performance of novel attacks.

Different exploitation strategies depend on *exploitable* bit flips, e. g., bit flips that occur at specific offsets and in particular directions. Typically, only one exploitable bit flip is required for a successful exploit chain. Thus, the attack runtime until the first exploitable bit flip may express a good estimation of the real-world applicability of a specific attack. Additionally, new insights on exploitation techniques may lead to novel exploit chains, allowing better estimations of the importance of Rowhammer outside the academic world. While the

[4] When Blacksmith [18] found an effective pattern, it *sweeps* over the same contiguous memory region and reports the number of bit flips.

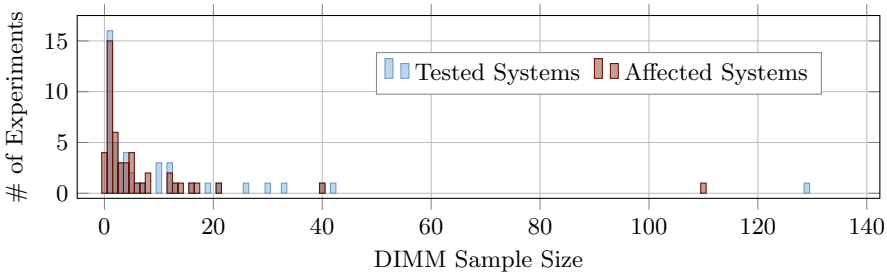

Fig. 2. Number of experiments for specific sample sizes and number of affected items.

raw number of bit flips, either scaled by time or a number of accesses, can still be used to compare different attack strategies on a consistent test setup, it does not provide a universally comparable metric across different machines.

4 Methodology

We started with a Google Scholar search for the word "Rowhammer" and found 2 509 publications. Google Scholar also includes publications that mention the word only once. Thus, we checked if a paper has ≥ 5 occurrences of that word; we ended up with 463 publications (including presentation, bachelor theses, etc.). Then, we manually filtered for peer-reviewed papers that perform Rowhammer attacks and ended up with 55 papers. Then, we filtered for papers in highly ranked conferences (CORE ranking A or A*) and ended up with 22 papers. After that, we had multiple meetings with researchers from different groups that have published Rowhammer attack papers in the past to discuss the papers and ask if relevant papers were missing. In the end, we selected 32 publications with 48 experimental evaluations. Some of the selected studies include experimental evaluations for different approaches to different types of systems. The list of papers we used in our analysis is in Appendix 10.

5 Analysis of Sample Sizes

We survey the sample sizes of 32 publications with 48 experimental evaluations. The results are shown in Figure 2.

The average sample size (e. g., tested DIMMs, mobile phones, single-board computers, etc. depending on the experiment) is 10.60, and the median sample size is 3.5, while most experiments used a sample size of 1. Of 48 experiments we analyzed, 16 used a sample size of 1, which limits the ability to draw general conclusions. We cannot exclude the possibility that these experiments depend on broken or faulty DIMMs and do not work on other systems. For these experiments, there is no information regarding the prevalence.

There are 19 experiments with a sample size between 2 and 10, and 6 with a sample size between 11 and 20. Three experiments have a sample size between

21 and 30, 2 between 31 and 40, and 1 between 41 and 50. After that, there is a gap until 129 DIMMs are analyzed by Kim et al. [28]. On average, 7.33 items are affected with a median of 2.0. Most experiments report 1 affected item.

The number of affected items (DIMM, Mobile Device, etc. depending on the experimental setup) was 0 in 4 experiments, so a specific attack or approach did not work. We group experiments based on the type of system verified, so it is possible that an attack worked on one system type but not the other: The experimental evaluation of HalfDouble [30] shows that it works on ARM-based devices but not on any x86-based devices. Most experiments identified 1 affected item. The small affected sample size does not allow for general conclusions, as the results may depend on broken or not correctly working DIMMs.

There are 20 experiments with a number of affected items between 2 and 10. 6 experiments show that 11 to 20 items are affected, 1 experiment reports 21, and 1 experiment reports 40 affected items. The experiment from Kim et al. [28] reports 110 affected DIMMs, the highest number of affected items. As discussed in Section 3.1, there are multiple reasons that bit flips can occur that are not caused by Rowhammer. While some effects, like cosmic rays, are uncontrollable and very rare, other issues, like bad memory cells, can affect multiple measurements. Therefore, our first recommendation, **$\mathcal{R}$1**, is to test DIMMs for any faults that may affect the accuracy of the experimental results.

> **$\mathcal{R}$1**: DIMMs used in empirical research must be tested for other problems, e. g., using Memtest86 (except for integrated Rowhammer tests), to ensure that no other (non-Rowhammer) problems are present.

Our second recommendation **$\mathcal{R}$2** is to increase the sample size to $\geq$ 30 DIMMs. This number is more of a rule-of-thumb from the central limit theorem than a strict cut-off for every experiment. Still, it is frequently referenced as a minimum viable sample size to achieve at least some diversity and statistical reliability. Additionally, we recommend including multiple manufacturers, different capacities, and various speeds to demonstrate a broader coverage.

> **$\mathcal{R}$2**: Increase the sample size to $\geq$ 30 DIMMs total, spread across 3 major vendors, each with at least 2 different capacities.

Our third recommendation **$\mathcal{R}$3** is to encourage the scientific community to do more reproduction studies, like Gerlach et al. [10].

> **$\mathcal{R}$3**: Do more reproduction studies of published work to gain more insights regarding the prevalence. More venues should accept reproduction studies.

6 Dependence on Elevated Attacker Privileges

This section reviews 32 publications and analyzes the experimental setup of 48 experiments. Figure 3 illustrates the results. The majority (68.57 %) of experimental setups use x86 systems. We hypothesize that this is the case because many tools already exist for x86, so they can be reused and adjusted. In 6 setups (12.5 %), mobile devices, such as smartphones and Chromebooks, were

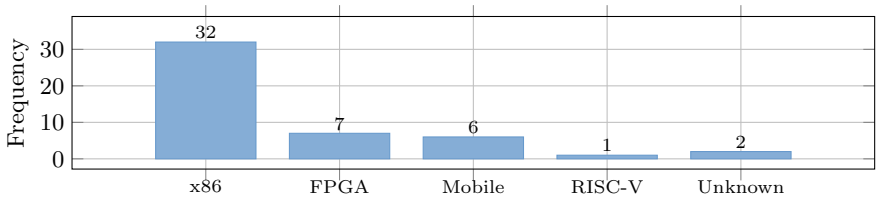

Fig. 3. Frequencies of the different experimental setups for Rowhammer experiments.

analyzed. Seven experimental setups (14.6 %) use an FPGA to send commands directly to the tested DIMMs. A RISC-V-based lab system was used in 1 experimental setup. For two setups, test systems were not described in detail, making it hard to reproduce them and impossible to estimate their practical impact.

FPGA experiments do not reflect realistic attack scenarios. FPGA-based setups send commands directly to the DIMMs. While allowing for greater control over the behavior of the DIMMs, this approach does not reflect a realistic attack scenario, where an attacker can only indirectly instruct the memory controller to access specific regions. Also, these setups may use specific parameters, e. g., timings, that are unavailable or uncommon on commodity systems. Also, it may not even be possible to configure a commodity system to use the parameters, e. g., timings, used by the FPGA-based setups. While the results of these experiments show essential insights into the DIMMs' low-level behavior, they do not represent a realistic attack scenario. Therefore, the relevance of these experiments for real-world attacks is limited.

Attacks require access to pagemap file. In Linux, `/proc/⟨pid⟩/pagemap` maps virtual to physical addresses. Access to this file is limited to privileged users since 2015 [29]. We found that 5 experimental setups require access to the pagemap file. Therefore, this requires either a severely outdated kernel or privileged access. In the first case, many other exploits, e. g., DirtyCOW [46], can be used to escalate privileges. In the second case, the attacker already has elevated privileges, so no further escalation is necessary.

Attacks require elevated privileges. We found that 6 experimental evaluations require 1 GiB Hugepages. Once these Hugepages are requested from the kernel and mounted somewhere, they can be mapped without elevated privileges. However, requesting and mounting Hugepages require elevated privileges. Therefore, these attacks are only realistic when the attacker has elevated privileges, or the system has requested and mounted 1 GiB Hugepages which are not used by another process (otherwise, the process of the attacker would not be able to map it). In the first case, no privilege escalation is necessary since the attacker already has root privileges. The latter case is exploitable but requires a specific, non-default system configuration.

Attacks require special OS settings. Razavi et al. [48] showed that exploiting Kernel Same-page Merging (KSM) combined with Rowhammer to trigger bit flips on another KVM guest on the same host is possible. However, this requires KSM to be enabled, which is not the case by default for most Linux distributions, except for special ones like Proxmox VE. The attack also requires

the attacker's process to be started in the attacker's VM before the process of the victim is started—similarly, Bosman et al. [4] exploited memory deduplication on Windows with a Rowhammer attack. Memory deduplication is a feature from Hyper-V and is not enabled by default on Windows Server. In our survey, we found that only 11 experimental evaluations assume a realistic attack scenario exploitable on a commodity system with default configuration. Since the prevalence of affected systems is not known as described in Section 5, no statistically significant estimations on the number of systems affected by Rowhammer can be made. Most publications introduce attacks assuming unrealistically high capabilities of the attacker or uncommon system configurations. Other publications require a custom memory controller based on an FPGA. Therefore, attacks should be classified based on the required preconditions. For example, attacks that require specific, non-standard configurations, should provide a reasonable explanation of why this configuration is realistic. Attacks that assume an unrealistically capable adversary should be clearly labeled as such. We recommend in **$\mathcal{R}$4** distinguishing between attacks that are possible in theory and attacks exploitable in a realistic experimental setup.

$\mathcal{R}$4: Attacks should only be classified as such when assessed under realistic attack scenarios, and there should be a more apparent distinction between actual attacks and potential (theoretical) attacks.

7 Comparability across Publications

Orosa et al. [44] showed that the number and position of bit flips depend on environmental parameters, e. g., temperature. Thus, comparing results from the same experimental setup is difficult when environmental parameters are unknown. Out of 48 experimental setups inspected, only 2 [44, 36] verified the impact of the temperature. In 2 other experiments, the authors reported a constant temperature [37, 61]. Two experiments measured the impact of the refresh interval t_{REFI}, but did not specify the temperature. 44 (91.6 %) did not specify the temperature and 46 experiments (95.83 %) did not specify t_{REFI}. The refresh interval t_{REFI} is defined to be 64 ms on DDR3 and DDR4 and 32 ms on DDR5. However, some mitigations set t_{REFI} to 32 ms on DDR3. In total, 42 experimental evaluation setups did not document any environmental parameters. Typically, no experimental evaluation of prior work is performed when a new attack is published. Due to the variability between experimental setups, new attacks' performance can not be compared to prior work.

Environmental parameters known to have effects on the susceptibility of systems to Rowhammer should be controlled, monitored, and documented in future work, In the case of temperature, we recommend keeping the room at a fixed, measured temperature or measuring the temperatures with the integrated sensors of the lab systems. t_{REFI} should be measured and documented for each system. Additionally, we should encourage reproducing prior experiments on different test setups to gain some "ground truth", which allows for better confidence when comparing different approaches.

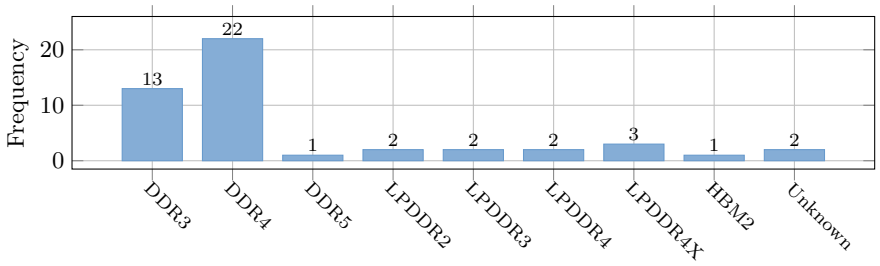

Fig. 4. Frequencies of different DRAM types in the analyzed experimental evaluations.

8 Unspecified Age and Wear of Hardware

DIMMs used for security research could be highly susceptible to Rowhammer because bit flips could "burn in" [27, 28]. However, most publications do not include the manufacturing date or wear of the DIMMs. This can lead to a self-increasing effect as DIMMs are used in more experiments over time [27, 28]. Since DDR4, most DIMMs support TRR, a Rowhammer mitigation based on detecting Rowhammer patterns. However, TRR is an umbrella term for many different (proprietary) vendor implementations. We assume that more recent TRR versions protect against more recent Rowhammer attacks. However, only 7 of the experimental evaluations we analyzed specify the manufacturing dates. It is impossible to estimate whether the hypothesis that older DIMMs are more strongly affected by older attacks is true. We recommend in **$\mathcal{R}$5** that authors should publish the manufacturing date of DIMMs.

> **$\mathcal{R}$5**: Authors should publish the manufacturing data of the DIMMs used in experimental evaluation.

In contrast to the manufacturing data, most papers specify the DRAM generation used in the experimental evaluation. Figure 4 gives an overview of different DRAM generations and the number of experimental evaluations that used them. We show that 13 experimental setups utilize DDR3 DIMMs and 22 utilize DDR4 DIMMs. In contrast, only 1 experimental evaluation was done on DDR5, even though it was released in 2020. The number of experiments performed on LPDDR is much lower: There are 2 experiments on LPDDR2, LPDDR3, and LPDDR4 each. For LPDDR4X, there are 3 experiments. Only 1 experiment analyzed Rowhammer on HBM2. Two publications did not mention which DRAM generation they used for experimental evaluation.

The generation of DRAM can be used to derive information regarding the age of the tested DIMM. Taking the year of the publication and the DRAM standard into account, and assuming that the DIMM was not manufactured after the standard for the next generation was available, we calculate the potential age of DIMMs used for experimental evaluation. The results are shown in Figure 5

When using this approach, the potential minimum age of a DDR3 DIMM, while used in an experiment, is 0 years for a publication from 2014 [28] when assuming the DIMM was manufactured in 2014 since the DDR4 standard was

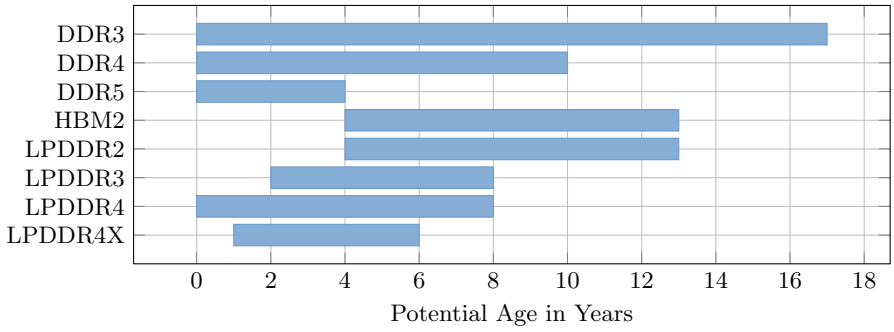

Fig. 5. Estimated potential age of DIMMs at the time of experimental evaluation.

released in 2014. The maximum age of DDR3 DIMMs used in any studies is 17 years in the case of a publication from 2024 [26] when assuming the DIMM was manufactured in 2007, immediately after the standard was launched. Different DRAM generations and standards have varying age ranges, affecting the reliability and features of DIMMs. Therefore, **$\mathcal{R}$6** states that authors should document the actual manufacturing dates of the used DIMMs. Additionally, they should provide an estimation of the wear of the DIMMs, e. g., how long and often they were used for Rowhammer evaluation. This would allow estimations on the reliability on the DIMMs, based on their age and wear.

$\mathcal{R}$6: Authors should submit information about the DIMMs' wear in experimental evaluation.

9 Suboptimal Metrics for Comparison

Kim et al. [28] were the first to use the number of bit flips as an absolute metric. This metric depends on the execution strategy, e. g., if the same memory area is scanned multiple times. Also, some works count only unique bit flips, while others count all. In that context, *uniqueness* may be based on the same memory cell, access patterns, and data stored in cells before performing the Rowhammer attack. Additionally, the number of bit flips strongly depends on the experiment's runtime or the size of the scanned memory area. This approach is used in multiple papers in our survey [28, 58, 62, 34, 36, 26, 9].

Some papers use the relative number of bit flips as an alternative. This approach aims to make the absolute number of bit flips comparable. By normalizing the number of bits flips against a reference value, e. g., the size of the scanned memory area or the scan time, this metric estimates how affected a single setup, or DIMM, actually is. However, these relative metrics are still influenced by the same factors as the absolute number of bit flips. This approach is used in multiple publications [28, 10, 42, 12, 58, 30, 60, 25, 37, 49, 55, 19]. In contrast to counting all bit flips, other publications count only exploitable bit flips. Exploitable bit flips is a better metric for estimating the impact of potential exploitation.

However, this strongly depends on the definition of *exploitable*: Attacks based on bit flips in the Page Frame Number (PFN) part of a PTE [7, 18, 30, 19]. It was also shown that cryptographic algorithms can be attacked by flipping bits in the keys [48, 7, 18, 19]. Bit flips can target opcodes in binaries and libraries [12, 7, 18, 19]. There are also attacks based on bit flips in URLs [48]. Thus, in **$\mathcal{R}$7**, authors should include multiple metrics for bit flips.

$\mathcal{R}$7: Authors should use multiple metrics for bit flips to allow for better comparisons to other works.

10 Conclusion

We systematically analyzed 32 publications with 48 experimental evaluations and identified six major threats to Rowhammer's research validity. We have shown that in 33 % of the experiments, the sample size is only 1; therefore, many other factors could be the reason for bit flips. From the overall 32 x86 consumer hardware, only 22 described an approach to get physical addressing information. Half of these 22 experiments on x86 required unrealistically high capabilities of an attacker (e. g., root privileges), making them an isolated problem in academia. We found that the experimental results are often incomparable because environmental parameters are not controlled or documented, and inconsistent units for bit flips have been used. Additionally, 25 analyzed publications do not document the age and wear of used hardware. We developed the following 7 recommendations with detailed justification to improve future Rowhammer research: pre-experimental testing of the DIMMs, increasing sample size, value reproduction studies, defining attacks in real-world conditions and distinguishing them from theoretical ones, publishing more information about the used DIMMs, including wear, and using multiple units for bit flips evaluation.

Acknowledgments. This work was funded by the Deutsche Forschungsgemeinschaft under grant number 503876675 and the Austrian Science Fund under grant number 10.55776/I6054, as well as the European Union under grant number ROF-SG20-3066-3-2-2.

This preprint has not undergone any post-submission improvements or corrections. The version of Record of this contribution will be published in the proceedings *Computer Security – ESORICS 2025*

References

[1] Misiker Tadesse Aga, Zelalem Birhanu Aweke, and Todd Austin. "When good protections go bad: Exploiting anti-DoS measures to accelerate Rowhammer attacks". In: *HOST*. 2017.

[2] AMD. *BIOS and Kernel Developer's Guide (BKDG) for AMD Family 16h Models 00h-0Fh Processors*. 2015. URL.

[3] Zelalem Birhanu Aweke, Salessawi Ferede Yitbarek, Rui Qiao, Reetuparna Das, Matthew Hicks, Yossi Oren, and Todd Austin. "ANVIL: Software-based protection against next-generation Rowhammer attacks". In: *ACM SIGPLAN Notices* 51 (2016), pp. 743–755.

[4] Erik Bosman, Kaveh Razavi, Herbert Bos, and Cristiano Giuffrida. "Dedup Est Machina: Memory Deduplication as an Advanced Exploitation Vector". In: *S&P*. 2016.

[5] Ferdinand Brasser, Lucas Davi, David Gens, Christopher Liebchen, and Ahmad-Reza Sadeghi. "CAn't Touch This: Software-only Mitigation against Rowhammer Attacks targeting Kernel Memory". In: *USENIX Security*. 2017.

[6] Chromium Issue Tracker. *Security: NaCl sandbox escape via DRAM "rowhammer" memory corruption*. 2014. URL.

[7] Lucian Cojocar, Kaveh Razavi, Cristiano Giuffrida, and Herbert Bos. "Exploiting Correcting Codes: On the Effectiveness of ECC Memory Against Rowhammer Attacks". In: *S&P*. 2019.

[8] Michael Fahr Jr, Hunter Kippen, Andrew Kwong, Thinh Dang, Jacob Lichtinger, Dana Dachman-Soled, Daniel Genkin, Alexander Nelson, Ray Perlner, Arkady Yerukhimovich, et al. "When Frodo Flips: End-to-End Key Recovery on FrodoKEM via Rowhammer". In: *CCS*. 2022.

[9] Pietro Frigo, Emanuele Vannacci, Hasan Hassan, Victor van der Veen, Onur Mutlu, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi. "TRRespass: Exploiting the Many Sides of Target Row Refresh". In: *S&P*. 2020.

[10] Lukas Gerlach, Fabian Thomas, Robert Pietsch, and Michael Schwarz. "A Rowhammer Reproduction Study Using the Blacksmith Fuzzer". In: *European Symposium on Research in Computer Security*. 2023.

[11] Google. *Measuring the DRAM refresh rate by timing memory accesses*. 2015. URL.

[12] Daniel Gruss, Moritz Lipp, Michael Schwarz, Daniel Genkin, Jonas Juffinger, Sioli O'Connell, Wolfgang Schoechl, and Yuval Yarom. "Another Flip in the Wall of Rowhammer Defenses". In: *S&P*. 2018.

[13] Daniel Gruss, Clémentine Maurice, and Stefan Mangard. "Rowhammer.js: A Remote Software-Induced Fault Attack in JavaScript". In: *DIMVA*. 2016.

[14] Martin Heckel and Florian Adamsky. "Reverse-Engineering Bank Addressing Functions on AMD CPUs". In: *Workshop on DRAM Security (DRAMSec)*. 2023.

[15] Christian Helm, Soramichi Akiyama, and Kenjiro Taura. "Reliable Reverse Engineering of Intel DRAM Addressing Using Performance Counters". In: *Modeling, Analysis, and Simulation of Computer and Telecommunication Systems (MASCOTS)*. IEEE. 2020.

[16] Intel. *Intel Xeon Processor E5 v4 Product Family: Datasheet Volume 2: Registers*. 2016.

[17] Yeongjin Jang, Jaehyuk Lee, Sangho Lee, and Taesoo Kim. "SGX-Bomb: Locking Down the Processor via Rowhammer Attack". In: *SysTEX*. 2017.
[18] Patrick Jattke, Victor van der Veen, Pietro Frigo, Stijn Gunter, and Kaveh Razavi. "BLACKSMITH: Rowhammering in the Frequency Domain". In: *S&P*. Nov. 2021.
[19] Patrick Jattke, Max Wipfli, Flavien Solt, Michele Marazzi, Matej Bölcskei, and Kaveh Razavi. "ZenHammer: Rowhammer Attacks on AMD Zen-based Platforms". In: *USENIX Security*. 2024.
[20] JEDEC Solid State Technology. *DDR3 SDRAM Standard*. 2012. URL.
[21] JEDEC Solid State Technology. *DDR4 SDRAM Standard*. 2021. URL.
[22] JEDEC Solid State Technology. *DDR5 SDRAM Standard*. 2024. URL.
[23] Sangwoo Ji, Youngjoo Ko, Saeyoung Oh, and Jong Kim. "Pinpoint Rowhammer: Suppressing Unwanted Bit Flips on Rowhammer Attacks". In: *AsiaCCS*. 2019.
[24] Jonas Juffinger, Lukas Lamster, Andreas Kogler, Maria Eichlseder, Moritz Lipp, and Daniel Gruss. "CSI: Rowhammer - Cryptographic Security and Integrity against Rowhammer". In: *S&P*. 2023.
[25] Jonas Juffinger, Sudheendra Raghav Neela, Martin Heckel, Lukas Schwarz, Florian Adamsky, and Daniel Gruss. "Presshammer: Rowhammer and Rowpress without Physical Address Information". In: *DIMVA*. 2024.
[26] Ingab Kang, Walter Wang, Jason Kim, Stephan van Schaik, Youssef Tobah, Daniel Genkin, Andrew Kwong, and Yuval Yarom. "SledgeHammer: Amplifying Rowhammer via Bank-level Parallelism". In: *USENIX Security*. 2024.
[27] Naghmeh Karimi, Arun Karthik Kanuparthi, Xueyang Wang, Ozgur Sinanoglu, and Ramesh Karri. "MAGIC: Malicious Aging in Circuits/-Cores". In: *ACM TACO*. 2015.
[28] Yoongu Kim, Ross Daly, Jeremie Kim, Chris Fallin, Ji Hye Lee, Donghyuk Lee, Chris Wilkerson, Konrad Lai, and Onur Mutlu. "Flipping Bits in Memory Without Accessing Them: An Experimental Study of DRAM Disturbance Errors". In: *ISCA*. 2014.
[29] Kirill A. Shutemov. *Pagemap: Do Not Leak Physical Addresses to Non-Privileged Userspace*. 2015. URL.
[30] Andreas Kogler, Jonas Juffinger, Salman Qazi, Yoongu Kim, Moritz Lipp, Nicolas Boichat, Eric Shiu, Mattias Nissler, and Daniel Gruss. "Half-Double: Hammering From the Next Row Over". In: *USENIX Security*. 2022.
[31] Andrew Kwong, Daniel Genkin, Daniel Gruss, and Yuval Yarom. "RAMBleed: Reading Bits in Memory Without Accessing Them". In: *S&P*. 2020.
[32] Benjamin C Lee, Engin Ipek, Onur Mutlu, and Doug Burger. "Architecting Phase Change Memory as a Scalable DRAM Alternative". In: *International Symposium on Computer Architecture (ISCA)*. 2009.
[33] Dawei Li, Di Liu, Yangkun Ren, Ziyi Wang, Yu Sun, Zhenyu Guan, Qianhong Wu, and Jianwei Liu. "FPHammer: A Device Identification Framework based on DRAM Fingerprinting". In: *TrustCom*. 2023.

[34] Moritz Lipp, Misiker Tadesse Aga, Michael Schwarz, Daniel Gruss, Clémentine Maurice, Lukas Raab, and Lukas Lamster. "Nethammer: Inducing Rowhammer Faults through Network Requests". In: *SILM Workshop*. 2020.

[35] Kevin Loughlin, Stefan Saroiu, Alec Wolman, and Baris Kasikci. "Stop! Hammer Time: Rethinking Our Approach to Rowhammer Mitigations". In: *Workshop on Hot Topics in Operating Systems*. 2021.

[36] Haocong Luo, Ataberk Olgun, Abdullah Giray Yağlıkçı, Yahya Can Tuğrul, Steve Rhyner, Meryem Banu Cavlak, Joël Lindegger, Mohammad Sadrosadati, and Onur Mutlu. "RowPress: Amplifying Read Disturbance in Modern DRAM Chips". In: *ISCA*. 2023.

[37] Michele Marazzi and Kaveh Razavi. "RISC-H: Rowhammer Attacks on RISC-V". In: *Workshop on DRAM Security (DRAMSec)*. 2024.

[38] Memtest86. *Troubleshooting Memory Errors*. 2021. URL.

[39] Onur Mutlu and Jeremie S. Kim. "RowHammer: A Retrospective". In: *IEEE TCAD* (2020).

[40] Amir Naseredini. *Exploring the Horizon: A Comprehensive Survey of Rowhammer*. 2023. arXiv: 2310.06950 [cs.CR]. URL.

[41] Ataberk Olgun, F. Nisa Bostanci, Ismail Emir Yuksel, Oguzhan Canpolat, Haocong Luo, Geraldo F. Oliveira, A. Giray Yaglikci, Minesh Patel, and Onur Mutlu. "Variable Read Disturbance: An Experimental Analysis of Temporal Variation in DRAM Read Disturbance". In: *HPCA*. 2025.

[42] Ataberk Olgun, Majd Osseiran, A Giray Yağlıkçı, Yahya Can Tuğrul, Haocong Luo, Steve Rhyner, Behzad Salami, Juan Gomez Luna, and Onur Mutlu. "An Experimental Analysis of RowHammer in HBM2 DRAM Chips". In: *DSN*. 2023.

[43] Ataberk Olgun, Yahya Can Tugrul, Nisa Bostanci, Ismail Emir Yuksel, Haocong Luo, Steve Rhyner, Abdullah Giray Yaglikci, Geraldo F Oliveira, and Onur Mutlu. "ABACuS: All-Bank Activation Counters for Scalable and Low Overhead RowHammer Mitigation". In: *USENIX Security*. 2024.

[44] Lois Orosa, Ulrich Rührmair, A Giray Yaglikci, Haocong Luo, Ataberk Olgun, Patrick Jattke, Minesh Patel, Jeremie Kim, Kaveh Razavi, and Onur Mutlu. "Spyhammer: Using rowhammer to remotely spy on temperature". In: *arXiv:2210.04084* (2022).

[45] Peter Pessl, Daniel Gruss, Clémentine Maurice, Michael Schwarz, and Stefan Mangard. "DRAMA: Exploiting DRAM Addressing for Cross-CPU Attacks". In: *USENIX Security*. 2016.

[46] Phil Oester. *CVE-2016-5195*. 2016. URL.

[47] Rui Qiao and Mark Seaborn. "A New Approach for Rowhammer Attacks". In: *HOST*. 2016.

[48] Kaveh Razavi, Ben Gras, Erik Bosman, Bart Preneel, Cristiano Giuffrida, and Herbert Bos. "Flip Feng Shui: Hammering a Needle in the Software Stack". In: *USENIX Security*. 2016.

[49] Finn de Ridder, Pietro Frigo, Emanuele Vannacci, Herbert Bos, Cristiano Giuffrida, and Kaveh Razavi. "SMASH: Synchronized Many-sided Rowhammer Attacks From JavaScript". In: *USENIX Security*. 2021.

[50] Jerome H. Saltzer and M. Frans Kaashoek. *Principles of Computer System Design: An Introduction*. 2009.

[51] Mark Seaborn. *Exploiting the DRAM rowhammer bug to gain kernel privileges*. Mar. 2015. URL.

[52] Mark Seaborn and Thomas Dullien. "Exploiting the DRAM Rowhammer bug to gain kernel privileges". In: *Black Hat USA*. 2015.

[53] Andrei Tatar. *Hammertime: a software suite for testing, profiling and simulating the Rowhammer DRAM defect*. 2018. URL.

[54] Andrei Tatar, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi. "Defeating software mitigations against rowhammer: a surgical precision hammer". In: *RAID*. 2018.

[55] Andrei Tatar, Radhesh Krishnan, Elias Athanasopoulos, Cristiano Giuffrida, Herbert Bos, and Kaveh Razavi. "Throwhammer: Rowhammer Attacks over the Network and Defenses". In: *USENIX ATC*. 2018.

[56] Fabian Thomas, Lukas Gerlach, and Michael Schwarz. "Hammulator: Simulate Now - Exploit Later". In: *DRAMSec*. 2023.

[57] Youssef Tobah, Andrew Kwong, Ingab Kang, Daniel Genkin, and Kang G Shin. "SpecHammer: Combining Spectre and Rowhammer for New Speculative Attacks". In: *S&P*. 2022.

[58] Victor van der Veen, Yanick Fratantonio, Martina Lindorfer, Daniel Gruss, Clémentine Maurice, Giovanni Vigna, Herbert Bos, Kaveh Razavi, and Cristiano Giuffrida. "Drammer: Deterministic Rowhammer Attacks on Mobile Platforms". In: *CCS*. 2016.

[59] Minghua Wang, Zhi Zhang, Yueqiang Cheng, and Surya Nepal. "Dramdig: A Knowledge-assisted Tool to UncoverDRAM Address Mapping". In: *Design Automation Conference (DAC)*. 2020.

[60] Yuan Xiao, Xiaokuan Zhang, Yinqian Zhang, and Radu Teodorescu. "One Bit Flips, One Cloud Flops: Cross-VM Row Hammer Attacks and Privilege Escalation". In: *USENIX Security*. 2016.

[61] A Giray Yağlıkçı, Haocong Luo, Geraldo F De Oliviera, Ataberk Olgun, Minesh Patel, Jisung Park, Hasan Hassan, Jeremie S Kim, Lois Orosa, and Onur Mutlu. "Understanding RowHammer Under Reduced Wordline Voltage: An Experimental Study Using Real DRAM Devices". In: *DSN*. 2022.

[62] Zhenkai Zhang, Zihao Zhan, Daniel Balasubramanian, Xenofon Koutsoukos, and Gabor Karsai. "Triggering Rowhammer Hardware Faults on ARM: A Revisit". In: *ASHES Workshop*. 2018.

[63] Zhi Zhang, Decheng Chen, Jiahao Qi, Yueqiang Cheng, Shijie Jiang, Yiyang Lin, Yansong Gao, Surya Nepal, Yi Zou, Jiliang Zhang, and Yang Xiang. "SoK: Rowhammer on Commodity Operating Systems". In: *Asia CCS*. 2024. DOI: 10.1145/3634737.3656998.

Appendix

Table 1 overviews the analyzed Rowhammer studies.

Author	Pattern	Memory Type	Environment	Test Setup	Focus	Sample size	Flips observed on	Year
Kim et al. [28]	One-Location	DDR3	Unspecified	FPGA	Bit Flips	129 DIMMs	110 DIMMs	2014
Qiao and Seaborn [47]	?	?	Unspecified	Unspecified	Exploitation	?	?	2016
Bosman et al. [4]	Double-Sided	DDR3	Unspecified	1 Lab System	Exploitation	1 DIMM	1 DIMM	2016
Veen et al. [58]	Double-Sided	LPDDR2	Unspecified	1 Smartphone	Exploitation, Bit Flips	1 Smartphone	1 Smartphone	2016
Veen et al. [58]	Double-Sided	LPDDR3	Unspecified	26 Smartphones	Exploitation, Bit Flips	26 Smartphones	17 Smartphones	2016
Veen et al. [58]	Double-Sided	LPDDR4	Unspecified	1 Smartphone	Exploitation, Bit Flips	1 Smartphone	0 Smartphones	2016
Razavi et al. [48]	Double-Sided	DDR3	Unspecified	1 Lab System	Exploitation	1 DIMM	1 DIMM	2016
Xiao et al. [60]	Single-Sided, Double-Sided	DDR3	Unspecified	5 Lab Systems	Exploitation, Bit Flips	5 DIMMs	4 DIMMs (only done on 4)	2016
Xiao et al. [60]	Single-Sided, Double-Sided	DDR4	Unspecified	1 Lab System	Exploitation, Bit Flips	1 DIMM	0 DIMMs (not done on DDR4)	2016
Gruss et al. [13]	Double-Sided	DDR3	Unspecified	2 Lab Systems	Bit Flips	6 DIMMs	5 DIMMs	2016
Gruss et al. [13]	Double-Sided	DDR4	tREFI	2 Lab Systems	Bit Flips	4 DIMMs	2 DIMMs	2016
Jang et al. [17]	Double-Sided	DDR4	Unspecified	1 Lab System	Exploitation, Bit Flips	1 DIMM	1 DIMM	2017
Aga et al. [1]	Single-Sided, Double-Sided	DDR4	Unspecified	1 Lab System	Bit Flips	4 DIMMs	3 DIMMs	2017
Gruss et al. [12]	One-Location	DDR3	Unspecified	2 Lab Systems	Exploitation, Bit Flips	4 DIMMs	4 DIMMs	2018
Gruss et al. [12]	One-Location	DDR4	Unspecified	1 Lab System	Exploitation, Bit Flips	2 DIMMs	2 DIMMs	2018
Tatar et al. [54]	Single-, Double-Sided, Amplified	DDR3	Unspecified	2 Lab Systems	Exploitation	33 Memory Setups?	14 Memory Setups?	2018
Lipp et al. [34]	Double-Sided	DDR4	Unspecified	3 Lab Systems	Exploitation, Bit Flips	1 DIMM	1 DIMM	2018
Lipp et al. [34]	One-Location	LPDDR2	Unspecified	1 Smartphone	Exploitation, Bit Flips	1 Smartphone	1 Smartphone	2018
Tatar et al. [55]	Double-Sided	DDR3	Unspecified	2 Lab Systems	Bit Flips	4 DIMMs	4 DIMMs	2018
Zhang et al. [62]	Double-Sided	LPDDR3	Unspecified	1 Single Board Computer	Bit Flips	1 Single Board	1 Single Board	2018
Cojocar et al. [7]	Double-Sided	?	Unspecified	?	Exploitation	?	?	2019
Ji et al. [23]	Double-Sided	DDR3	Unspecified	1 Lab System	Bit Flips	16 DIMMs	12 DIMMs	2019
Kwong et al. [31]	Single-Sided, Double-Sided	DDR3	Unspecified	1 Lab System	Bit Flips	2 DIMMs	2 DIMMs	2020
Frigo et al. [9]	Many-Sided	DDR4	Unspecified	1 Lab System	Bit Flips	42 DIMMs	13 DIMMs	2020
Frigo et al. [9]	Many-Sided	LPDDR4X	Unspecified	13 Mobile Devices	Bit Flips	13 Mobile Devices	5 Mobile Devices	2020
Ridder et al. [49]	Many-Sided	DDR4	Unspecified	3 Lab Systems	Bit Flips	5 DIMMs	3 - 5 DIMMs (not clarified)	2021
Jattke et al. [18]	Fuzzed (Blacksmith)	DDR4	Unspecified	10 Lab Systems	Bit Flips	40 DIMMs	40 DIMMs	2022
Jattke et al. [18]	Fuzzed (Blacksmith)	LPDDR4X	Unspecified	JEDEC developer board	Bit flips	19 Chips	16 Chips	2022
Kogler et al. [30]	Half-Double	DDR4	Unspecified	FPGA	Bit Flips	3 DIMMs	2 DIMMs	2022
Kogler et al. [30]	Half-Double	LPDDR4X	Unspecified	7 Mobile Devices	Bit Flips	7 Mobile Devices	5 Mobile Devices	2022
Kogler et al. [30]	Half-Double	DDR4	Unspecified	1 Notebook	Bit Flips	1 Notebook	0 Notebooks	2022
Kogler et al. [30]	Half-Double	LPDDR4	Unspecified	2 MiniPCs	Bit Flips	2 MiniPCs	0 MiniPCs	2022
Tobah et al. [57]	Double-Sided	DDR3	Unspecified	1 Lab System	Exploitation, Bit Flips	3 DIMMs	3 DIMMs	2022
Tobah et al. [57]	Many-Sided	DDR4	Unspecified	3 Lab Systems	Exploitation, Bit Flips	3 DIMMs	3 DIMMs	2022
Orosa et al. [44]	Single-Sided	DDR4	Temperature	FPGA	Bit Flips	12 DIMMs	12 DIMMs	2022
Yağlıkçı et al. [61]	Double-Sided	DDR4	50C	FPGA	Bit Flips	30 DIMMs (272 Chips)	64 Chips	2022
Fahr Jr et al. [8]	Double-Sided	DDR3	Unspecified	1 Lab System	Exploitation	2 DIMMs	¿= 1 DIMM	2022
Gerlach et al. [10]	Fuzzed (Blacksmith)	DDR4	Unspecified	4 Lab Systems	Bit Flips	10 DIMMs	8 DIMMs	2023
Olgun et al. [42]	Double-Sided	HBM2	Unspecified	FPGA	Bit flips	1 Chip	1 Chip	2023
Luo et al. [36]	Single-Sided	DDR4	Temperature	FPGA	Bit Flips	21 DIMMs	21 DIMMs	2023
Luo et al. [36]	Single-Sided	DDR4	Unspecified	1 Lab System	Bit Flips	1 DIMM	1 DIMM	2023
Juffinger et al. [25]	Fuzzed (Blacksmith)	DDR4	Unspecified	Lab Systems	Bit Flips	12 DIMMs	6 DIMMs	2024
Juffinger et al. [25]	Single-Sided	DDR4	Unspecified	Lab Systems	Bit Flips	12 DIMMs	2 DIMMs	2024
Marazzi and Razavi [37]	Double-Sided	DDR4	23C	1 Lab System (RISC-V)	Bit Flips	1 DIMM	1 DIMM	2024
Kang et al. [26]	Many-Sided	DDR3	Unspecified	1 Lab System	Bit Flips	1 DIMM	? DIMMs	2024
Kang et al. [26]	Many-Sided	DDR4	Unspecified	1 Lab System	Bit Flips	2 DIMMs	2 DIMMs (not clarified)	2024
Jattke et al. [19]	Fuzzed (Zenhammer)	DDR4	Unspecified	3 Lab Systems	Bit Flips	10 DIMMs	8 DIMMs	2024
Jattke et al. [19]	Fuzzed (Zenhammer)	DDR5	Unspecified	1 Lab System	Bit Flips	10 DIMMs	1 DIMM	2024

Table 1. Overview of the analyzed Rowhammer studies.