SeNonDiv: Securing Non-Volatile Memory using Hybrid Memory and Critical Data Diversion

Arijit Nath, Manik B. Bhosle and Hemangee K. Kapoor
Department of CSE, IIT Guwahati, Assam, India-781039
Email: {arijitnath, mb.bhosle, hemangee}@iitg.ac.in

***Abstract*—The emerging Non-volatile memories are projected as alternatives of traditional DRAM-based main memories. However, their non-volatility feature leads to serious security vulnerabilities. The sensitive data stored in these memories can be easily taken away due to prolonged data retention. A wide variety of encryption-based techniques protect these data at the cost of harmful side effects of encryption algorithms like high encryption/decryption latency and increased encryption-induced write activities. It launches a tug-of-war between security provisioning and system performance degradation as well as shortened lifetime of NVMs.**

In this paper, we propose a data-diversion based technique that protects the security-sensitive data of the applications by allocating the security critical pages in the volatile DRAM part of a DRAM-PCM hybrid main memory system on page faults. Experimental evaluation shows significant improvements in performance and lifetime compared to a partial encryption and a full encryption based technique.

***Index Terms*—Security-Sensitive, Non-volatile Memory, Phase Change Memory**

I. Introduction

The recent advancements in the semi-conductor technology bring forth an epoch-making revolution in the processing speed of the modern chip multiprocessors. This unprecedented growth in the computational power of these modern processors provides an excellent bedrock for the concurrent execution of the modern data-intensive applications. These applications demand large shares of memories to run in an efficient and secure manner. Also, the frequent interaction of these applications with security-sensitive data, that are prone to different security breaches make their execution even more challenging. Apart from this, the speed of the memory system that is not gearing up in sync with the computational power remains as a bottleneck for the execution of these workloads. The traditional DRAM-based main memories are not scalable enough to provide the needs like high density and low leakage energy consumption of the modern memory systems. The emergence of non-volatile memories (NVM) [1]–[3] like PCM, STT-RAM, ReRAM could suppress these issues prevalent in DRAM-based systems. With fascinating features like non-volatility, high density, and low leakage energy consumptions, these memories show immense potential of being replacement candidates for DRAM. However, downsides like poor cell endurance and high write latency thwart their adoption as a mainstream main memory standard. DRAM-PCM based hybrid memories that are composed of DRAM and PCM offer benefits of both DRAM and PCM. DRAM being faster in terms of read and write operations could deliver better performance when combined with PCM, whereas the high density of PCM provides more space for large data storage.

TABLE I
Comparison between Un-encrypted and Encrypted PCM ($*$: Lesser is better, $+$: More is better)

	CPI^*	$BitFlips^*$	$Energy^*$	$Lifetime^+$
Un-encrypted	1	1	1	1
Encrypted	1.32	3.14	1.16	0.36

Non-volatility, being the most intrinsic characteristic of NVMs, comes in handy while dealing with power/system failures, checkpointing improvement, and reducing application startup. However, this data persistence often opens the door to many data confidentiality attacks like stolen DIMM [2] and bus snooping attacks [4]. An adversary having access to the NVM DIMM can easily stream out sensitive data contents stored in these memories. A wide variety of techniques to deal with such attacks use encryption-based techniques to guard data contents stored in NVMs. However, encryption algorithms have their own latency overhead that impacts the system performance as encryption before data storage stands in the critical path of application execution. Not only that, the encryption algorithms show the Avalanche Effect [4] that leads to an enormous bit flips even in the change of a single bit in the input data. These write write-activities tend to shorten the lifetime of these NVMs in a terrible manner. Table I shows the effect of AES-based encryption (used in [4], [5]) to the stored data in the PCM main memory. It shows that encryption provides security to the stored data but compromises performance by 32%, increases bit flips and Energy consumption by 3.14 times and 16%, and reduces lifetime by 64% compared to the un-encrypted memory. Apart from this, most of these encryption-based security solutions encrypt all incoming data contents without considering their sensitivity. However, real-world applications deal with data contents with varying degrees of security needs. Some of them are highly security-sensitive (eg., Passwords, Credit cards credentials in banking system, etc.) and need higher protection against data theft vulnerabilities. Other data are less sensitive and could be kept in memory without security with no harmful consequences. This distinction between data contents as sensitive vs. non-sensitive helps in protecting the sensitive data.

In this paper, we propose a data-diversion based technique

called SeNonDiv, considering a hybrid main memory composed of DRAM and PCM. Unlike other techniques like [6]–[9] that utilize hybrid memories for performance improvement, SeNonDiv takes advantage of the volatile nature of the DRAM portion in the hybrid memory to provide security to the sensitive data. Depending on the availability of the sensitivity information of data pages to the Memory Management Unit (MMU) at page load time, SeNonDiv allocates the sensitive data in the DRAM, whereas the non-sensitive data can be allocated in the remaining areas of PCM and DRAM. However, placing non-sensitive data in PCM (compared to DRAM) is given more priority so that enough space in DRAM could be maintained for storing the sensitive data. Thus, the critical data stored in the volatile DRAM are lost once the system is powered off, while remanence of the non-critical data does no harm even if they are stolen. Unlike encryption-based techniques that induce bit flips and shorten the lifetime of NVMs, SeNonDiv offers security to the critical data by using the volatile nature of DRAM in hybrid memories and saves the PCM main memory from the harmful implications of encryptions. Note that SeNonDiv primarily targets the stolen DIMM attacks. However, security analysis related to other attacks like Bus-snooping and Cold boot attacks are discussed in Section IV.B.

The main contributions of the paper are as follows:

- We propose a data-diversion based technique called SeNonDiv to provide security to the critical data by placing them in the volatile DRAM portion of a DRAM-PCM hybrid main memory. The non-critical data are placed in the remaining memory.
- We compare the results of our proposed technique with two AES [10] based encryption variants. The proposed technique is evaluated in Gem5 [11] full system simulator integrated with NVMain [12].
- A security analysis of SeNonDiv against various security attacks is also presented that details the minor adjustments, when done in SeNonDiv, protects the sensitive data against those attacks.

The rest of the paper is organized as follows. Background and related works are presented in section 2. The proposed methodology is discussed in section 3. Section 4 elaborates the experimental evaluation, results and analysis and security analysis, followed by conclusion in section 5.

II. Background and Related Work

A. Background

PCM [1], [13], [14] is a Non-Volatile memory that uses phase-change chalcogenide material such as $Ge_2Sb_2Te_2$ to store binary bits. The states of these materials can be altered between amorphous and crystalline by heating the material to different temperatures. If the material is heated to more than its melting point (600°C) for a short period and quickly cooled down, it changes its state to amorphous with high resistance, representing a '0' state. On the other hand, if the material is heated to a temperature between crystalline (300°C) and melting point (600°C) for a longer duration, it changes to low resistance crystalline state, representing a '1' state. The PCM cells can withstand only a limited number of writes ($\sim 10^8$ writes) before wearing out completely. Also, the write operations are slower (~ 3 times) than reads.

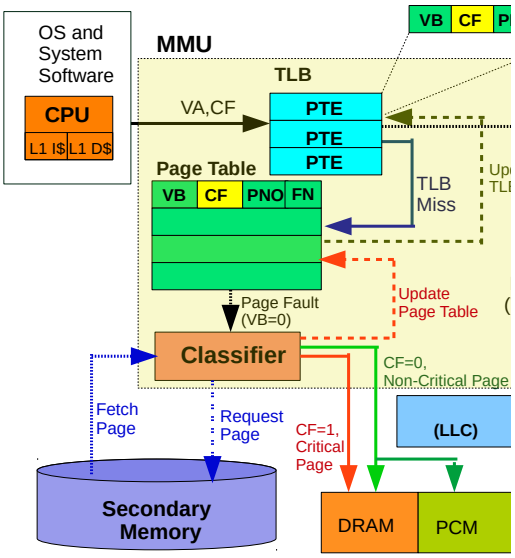
B. Related Work

Several software and hardware-based solutions are available to deal with the protection of sensitive data.

Intel SGX [15] in an extension to Intel's architecture that provides a set of security-related instruction codes that allow user and OS codes to define some private regions inside memory called enclaves. The contents within the enclaves are protected from processes running outside the enclave. Tamper-resistant software (TRS) [16] provides a set of techniques that make the understanding of the critical code difficult by changing the logic flow of critical regions in the applications. Flicker [17] developed by CMU creates a simple hosting environment alongside the primary OS and provides safety to the critical data by executing them in a secure environment.

The hardware-based solutions [2], [4], [18] that focus on providing security to sensitive data are mainly dependent on encryption techniques. The techniques [2], [4], [5] encrypt cache lines coming to NVM memories. Some techniques [4], [19], reduce encryption overhead by encrypting only the modified words within the cache lines. The technique DeWrite proposed in [5] cancels the writebacks of duplicate blocks residing in the main memory. Technique proposed in [20] reduces the Avalanche effect of encryption using the idea of compression and selective encryption. The highly compressed cache blocks generated after encryption are encrypted fully, while they rely on a non-deterministic fine-grain selective-encryption mechanism for poorly compressible data blocks. The NVM based caches also face similar data threats due to longer data retention. Techniques like semi NVM and data erasure at power off [21] are helpful in protecting sensitive data in STT-RAM-based caches. Cache bypassing and checkpointing [22] technique provide safety against data tempering based attacks. Also, Hardware Trojans discussed in [21], [23] possess threats by streaming out victim's secret information and launching fault injection-based as well denial of service (DoS) attacks. Authenticated Encryption (Packet authentication combined with encryption) based techniques like [18], [24], [25] prevent the extraction of secret information from the IP cores in NoC frameworks.

However, the hardware-based encryption techniques have to carry the burden of the adverse effects that come along with encryption. Encryption/Decryption are costly processes that involve latency overhead which impacts system performance. Also, encryption induced bitflips reduce the lifetime of NVMs. On the contrary, our hardware-based solution that completly avoids encryption, not only protects the critical data, but also improves performance and lifetime of PCM memory.



VA: Virtual Address, PA: Physical Address, VB: Valid
CF: Critical Flag, PNO : Page Number,
FN: Frame Number, PTE : Page Table Entry

Fig. 1. Architecture of the Proposed Scheme: SeNon

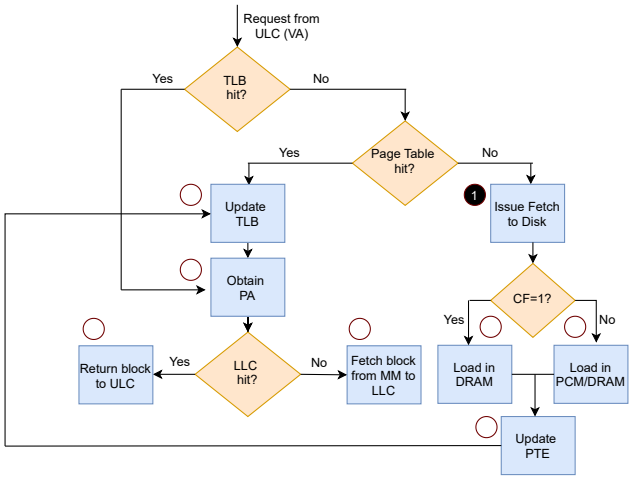


Fig. 2. Flowchart of our Proposed Technique: SeNonDiv

III. Proposed Methodology

A. Architecture

During the execution of an application, it inte data that needs a varying degree of security. Secu of some sensitive data (like important kernel dat pointer, confidential user information like passwo card details, etc.) are much higher than other les data. The hint of the criticality of a data block provided to the memory management unit (MMU help of a critical flag. Figure 1 shows the architecti proposed scheme SeNonDiv.

The CPU generates Virtual Addresses (VA) that are translated by the MMU to corresponding Physical Addresses (PA). These PAs are then fed to the memory controller for servicing required read/write requests to that PA. The MMU is responsible for the address translation, and it interacts with *Translation Look-Aside Buffer* and *Page Table* for translation. We incorporate a *Classifier* module in the MMU that loads the pages coming to main memories on page faults. It is shown in the diagram in Figure1 and is a part of MMU. We summarize the working of various components of MMU below:

•***Translation Look-Aside Buffer*** **(TLB) :** It is a fast cache located inside the processor chip that contains the page table entries (PTE) of the most recently accessed pages. We extend the PTE by including a one-bit Critical Flag (CF). This bit is set by the OS kernel by executing some privileged instructions (via system calls) on finding sensitive data contents on load time. The information of sensitivity of the pages could be passed to the OS kernel by the applications with the help of programming language primitives and compiler directives. For non-critical pages, this flag is reset to 0.

•***Page Table*** **(PT) :** It is stored in the Hybrid main memory. It stores the PTEs of all the pages of the running process.

•***Classifier*** **:** On a page fault (when the valid bit (VB) =0), the page is fetched from the main memory. The *classifier* module in the MMU checks the Critical Flag (CF) in the PTE of the referenced page. Then, it diverts the incoming page to the appropriate partition in the hybrid memory based on the critical flag. The corresponding PTE of the page is updated by setting the Valid Bit (VB) as 1 and adding the new frame number.

B. Operation

The architecture of SeNonDiv extends the Page Table Entry (PTE) by adding a 1-bit critical flag (CF). This bit can be set by executing some privileged instructions of the OS kernel invoked via system calls on finding sensitive data contents. The MMU translates address from virtual to physical with the help of TLB and Page Table (PT). The *Classifier* module of the MMU allocates the pages in hybrid main memory with the help of the critical flag inside the PTE of the page. The CPU generates virtual address that goes to MMU for address translation. We describe the workflow of our proposed technique: SeNonDiv as follows. The flow chart of the operations of SeNonDiv is presented in Figure 2.

The requests from the Upper Level Caches (ULC) come in the form of virtual addresses (VA) that get translated to their corresponding physical addresses (PA) with the help of TLB before reaching the LLC. The process of address translation is described below:

1. TLB Hit : Getting a TLB hit indicates the presence of the PTE of the page number contained in the VA. Therefore, the VA is translated to its corresponding PA by using the PTE present in the TLB (shown as ❻ in Figure 2). This PA is used to search the block in the LLC.

2. TLB Miss : TLB miss indicates that PTE is absent in the TLB. Therefore, the MMU must consult the Page Table for address translation. It checks the valid bit (VB) of the corresponding PTE in the Page Table. If VB=1, the block is

present in the main memory (Page-Table hit). Update the TLB using the entry from the page-table (shown as ❺). Use this entry to generate the PA for accessing the LLC (shown as ❻).

On the other hand, in the Page Table, VB=0 indicates that the block is not present in the main memory (page-table-miss/Page Fault). The page containing the block must be fetched from the secondary storage (Shown as ❶). A request for the missed page is generated to the secondary memory. The *Classifier* loads the incoming pages from secondary storage to the hybrid main memory. It allocates the pages with the help of the critical flag (CF) of the corresponding PTE of the missed page. If the CF=1, it treats the incoming page as *Critical* and allocates it in the DRAM region of the hybrid main memory (shown as ❷). On the other hand, if the CF=0, the *Classifier* treats the incoming page as *Non-Critical* and allocates it in the remaining space of PCM and DRAM (shown as ❸). The allocation of the *Non-Critical* pages is preferably done in the PCM region of the hybrid memory so that enough space could be kept available in the DRAM region for allocation of the Critical pages. It then updates the PTE of the page by setting valid bit (VB) as 1 and inserts the frame number of the frame allocated to the page in the PTE (shown as ❹). The TLB is updated by bringing that PTE to TLB (shown as ❺). Finally, the corresponding PA is generated before LLC search begins (shown as ❻).

3. LLC access: With the help of the PA, the LLC is searched for the requested block. On LLC hit, the block is returned from LLC to the requesting ULC (shown in ❼). On the other hand, an LLC miss triggers a main memory fetch for the block (shown as ❽). Finally, the block is returned to LLC, which returns the block to the requestor ULC.

IV. Experimental Evaluation

The amount of critical data in the applications across different organizations varies depending on its risk level and the data storage methods. However, the percentage of critical data is lower compared to the whole data contents. Therefore, without loss generality, we take 25% of the accesses to contain critical data while performing our experiments. We implemented our proposed technique: SeNonDiv on a full system simulator Gem5 [11] integrated with NVMain [12], a cycle-accurate main memory simulator designed for NVMs. The system parameters used in the experiments are shown in Table II. We evaluated our results using SPEC 2006 benchmark suite [26]. We have taken the AES encryption latency to be 96ns per line based on the specifications [27].

We compare our results with two encryption-based techniques. Both techniques use counter mode encryption using block cipher AES like in [4], [5]. Apart from this, we have added another baseline that takes full PCM memory without encryption (NoEncr) to show its comparison of performance with SeNonDiv(In figures 3, 4 and 5). [1]

[1]However, the parameters like bitflips, energy and lifetime per PCM banks for SeNonDiv and NoEncr will be almost same and therefore are not shown in the respective figures (Figures 6, 7 and 8).

TABLE II
System Parameters

Components	Parameters
Processor	2Ghz, Single-Core, Alpha
L1 Cache	Private, 32KB SRAM, Split I/D Caches, 2-way assoc, 64 B block,1 cycle latency
L2 Cache(LLC)	Shared, SRAM, 64 B block, 8-way assoc, 10- cycle latency, 4MB Cache
Main Memory	DRAM : 2GB, 2 Channels, PCM : 2GB, 2 Channels
Memory latency [5]	DRAM : Read 50ns Write 50ns PCM : Write 60ns Write 150ns
Benchmarks	lbm, libquantum, calculix, gromacs, sjeng, namd, leslie3d

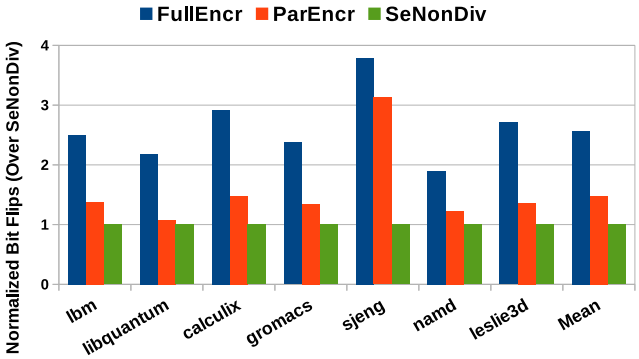


Fig. 3. Normalized Bit Flips per PCM bank over SeNonDiv (Lesser is better)

•**No Encryption (NoEncr) :** Here the main memory is composed of 4GB PCM. Blocks are placed without encryption.
•**Partial Encryption (ParEncr) :** The incoming blocks are monitored for criticality by checking their *critical flag* that is set during the LLC miss. If an incoming block is found to be *critical*, then it is encrypted. Otherwise, the *non-critical* blocks are not encrypted. The main memory is composed of 4GB PCM (4 Channels, 1GB/Channel).
•**Full Encryption (FullEncr) :** Every incoming cache block is encrypted irrespective of their criticality. The main memory is composed of 4GB PCM (4 Channels, 1GB/Channel).
•**Proposed Technique (SeNonDiv) :** It checks the criticality of the pages coming to the main memory from secondary memory. The *critical* pages are loaded in the volatile DRAM part of the main memory, while the *non-critical* pages are stored in the remaining DRAM and the PCM of the hybrid main memory.

A. Results and Analysis

The proposed technique: SeNonDiv is compared against the two techniques ParEncr and FullEncr in terms of bitflips reduction, Energy consumption, Write Speedup, lifetime, performance and, Average Memory Access Time (AMAT), of PCM memory. SeNonDiv protects sensitive data without the use of encryption. On the other hand, the encryption-based techniques provide security guarantee to the critical data at the cost of encryption induced increased write activities.

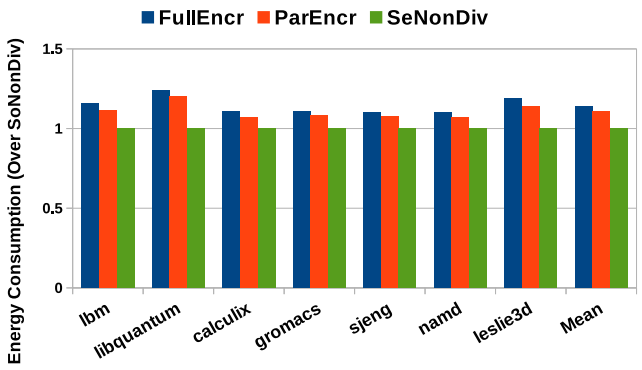


Fig. 4. Normalized Energy consumption per PCM bank over SeNonDiv (Lesser is better)

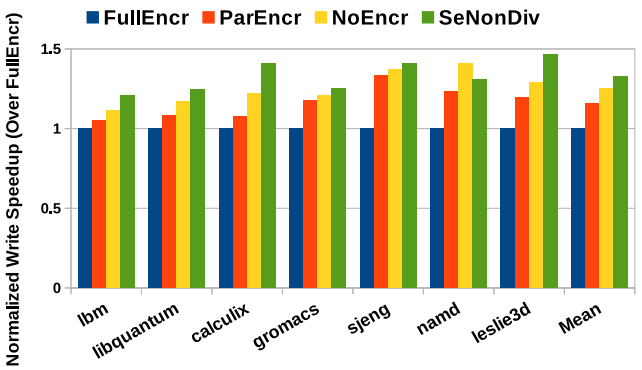


Fig. 5. Normalized Write Speedup over FullEncr (More is better)

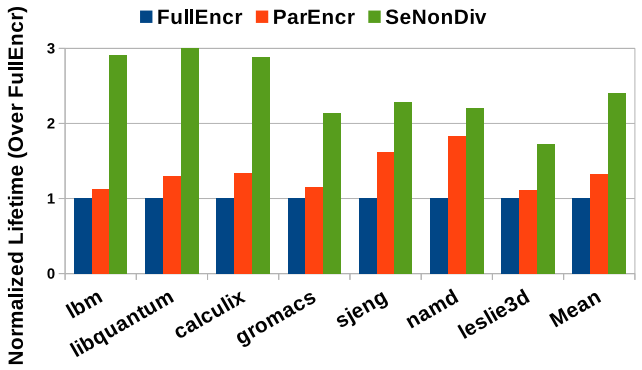


Fig. 6. Normalized Lifetime per PCM bank over FullEncr (More is better)

1. Effect on Bit Flips and Energy Consumption:

SeNonDiv allocates critical data based on a critical flag. The non-critical data gets allocated in the remaining DRAM region or PCM memory without encryption. Therefore, it reduces bit flips in the PCM memory substantially compared to the other two encryption-based techniques: ParEncr and FullEncr. ParEncr only encrypts the critical cache blocks, whereas FullEncr encrypts every cache block irrespective of their criticality. Therefore, bitflips for FullEncr are far more compared to the ParEncr technique. On average, SeNonDiv reduces bitflips by 47% and 156% over ParEncr and FullEncr, respectively. On the other hand, ParEncr reduces bitflips by 42% over FullEncr technique (Shown in Figure 3)

For PCM-based main memories, write energy plays a dominant role in the overall energy consumption. Therefore, techniques reducing bitflips also reduce energy consumption. SeNonDiv shows a considerable reduction in energy consumption in the PCM arrays. Figure 4 shows normalized energy consumption of SeNonDiv and the two encryption-based techniques : ParEncr and FullEncr. On average, the reductions in energy consumption shown by SeNonDiv over ParEncr and FullEncr are 10% and 14%, respectively. On the other hand, ParEncr shows only 3% improvement in energy consumption over FullEncr technique.

2. Effect on Write Speedup:

The latency to perform write operations is more than the read operations for NVMs. Therefore, write operations play a major role in improving system performance. Encryption based techniques incur heavy encryptions latency. This encryption latency gets added with the write latency of PCM since encryption comes in the critical path in the execution. On the other hand, SeNonDiv avoids encryption entirely and, the writes need lesser latency to complete. Therefore, SeNonDiV achieves a fairly nice speed up in the write operations. On average, SeNonDiv shows speedup of 6%, 14% and 32% over ParEncr and FullEncr, respectively (Shown in Figure 5).

3. Effect on Lifetime:

PCM-based memories have limited write endurance, i.e., they can withstand only a limited number of writes before wearing out completely. The lifetime of PCM-based memory is the timespan until the first breakdown of a byte in the PCM-memory. We use the number as well as the distribution of bitflips in the PCM cells to determine the lifetime of the PCM component of the hybrid main memory.

Enormous bit flips induced by encryption leads to increased write-activities in the PCM memory cells. These costly writes accelerate the wearing out of the PCM cells and lead to severe degradation in lifetime. However, our technique SeNonDiv provides security to critical data pages without encryption. Therefore, SeNonDiv does not have to pay the burden of encryption induced write activities in the PCM cells and improves lifetime over both the encryption-based techniques: ParEncr and FullEncr. Lifetime improvement shown by SeNonDiv over ParEncr and FullEncr are 33% and 140% respectively (shown in Figure 6).

4. Implications on Performance and AMAT:

Performance is measured in terms of Cycles Per Instruction (CPI). Encryption incurs substantial encryption and decryption latency during write and read operations in the PCM memories. These latencies directly influence the cycles needed to complete one instruction i.e, CPI, as they come in the critical path of instruction execution leading to higher CPI. Fortunately enough, our technique SeNonDiv allocates pages to DRAM and PCM partitions based on their criticality without relying on encryption. Also, the DRAM partition of hybrid memories is faster in terms of reads and writes (particularly

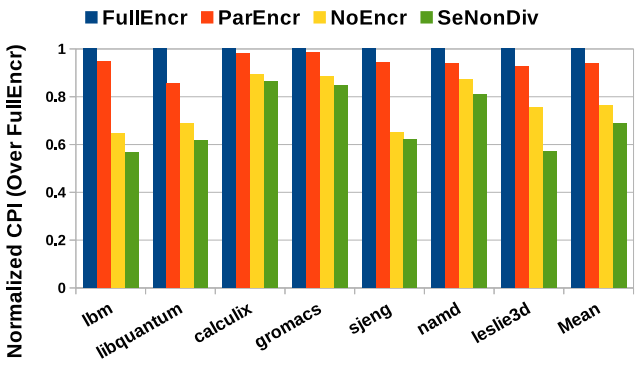


Fig. 7. Normalized CPI over FullEncr (Lesser is better)

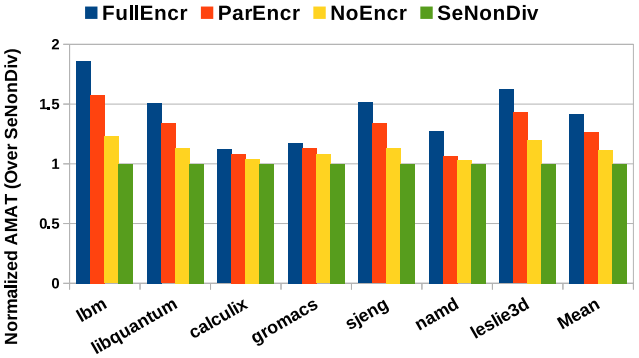


Fig. 8. Normalized AMAT over SeNonDiv (Lesser is better)

writes). Therefore, critical data storage and retrieval become faster in hybrid memories compared to PCM only memories. Figure 7 shows the normalized CPI (Over FullEncr) of SeNonDiv, NoEncr,ParEncr and FullEncr respectively. On average, the improvements in CPI shown by SeNonDiv over NoEncr, ParEncr, and FullEncr are 11%, 27%, and 32%, respectively.

Average Memory Access Time (AMAT) gives the average estimation of the time taken to deliver the data items from the memory hierarchy. Reduction in AMAT helps in improving the system performance. Data delivery from main memory to LLC is slower in the case of Encryption based techniques as the decryption involves high latency. Since SeNonDiv does not encrypt the stored data, therefore it can be delivered quickly. Also, critical data, along with other non-critical data stored in the DRAM partition, can be delivered quickly compared to the data stored in PCM partition. As a result, SeNonDiv outperforms NoEncr, ParEncr and FullEncr with reduction of 12%, 27% and 42% in AMAT, respectively (Figure 8).

B. Security Analysis

Our technique SeNonDiv offers security guarantee against stolen DIMM attacks. However, with slight changes, SeNonDiv can be made immune against bus snooping attacks and cold boot attacks.

1. Stolen DIMM attack : Storing sensitive data contents in NVMs is risky due to their longer data retention. SeNonDiv classifies the data based on sensitivity and places the sensitive data on DRAM memory in a DRAM-PCM hybrid memory. Sensitive data stored in volatile DRAM are erased once the system is powered off. Therefore, an attacker can not stream out the sensitive data contents from a stolen DRAM DIMM. On the other hand, acquiring non-sensitive data contents stored in NVM does not possess any security threat.

2. Bus Snooping Attack : By observing the sensitive data transferred to DRAM DIMM over an off-chip data bus, an attacker can launch a bus-snooping attack [4], [28]. Therefore, we propose an enhancement of SeNonDiv called SeNonDiv$_{adv}$, which protects sensitive data by encrypting them before transferring to DRAM. However, SeNonDiv$_{adv}$ compromises performance compared to SeNonDiv as it uses encryption for sensitive data stored in DRAM portion of DRAM-PCM hybrid main memory. Experimental results reveal that SeNonDiv improves performance (CPI) by 39% over full encryption, while the improvement shown by SeNonDiv$_{adv}$ is 17%. However, SeNonDiv$_{adv}$ protects sensitive data contents from bus-snooping based attack at the cost of a slight degradation in performance compared to the original SeNonDiv.

3. Cold Boot Attack : DRAM chips are reported to retain data for longer time duration provided they are cooled to a very low temperature [29], [30]. This form of attacks are known as cold boot attacks. Our technique SeNonDiv is not designed to address cold boot attack. However, using data scrambling based technique [29] in DRAM portion of the hybrid memory in conjunction with SeNonDiv can ensure guarantee of sensitive data protection in DRAM.

C. Overhead Analysis

We take a conventional PTE size [31] for our experiments, where a PTE entry requires 41 bits (=1-bit valid bit+ 20 bit Page Number + 20 bit Frame Number). We extend this PTE to include one bit Critical Flag (CF). Therefore, our extended PTE includes 1-bit as overhead over 41 bits. Therefore, storage overhead for keeping this extra CF bit is 2.44% only.

V. Conclusion

Non-volatile memories are promising candidates for constructing high density and energy-efficient main memories. However, the non-volatility feature of NVMs opens door to many data confidentiality attacks like stolen DIMM, bus snooping attack, trojan launch, etc. The traditional encryption-based techniques provide safeguard against sensitive data contents. However, the encryption induced write-activities catalyzes the wear-out process of these memories. Unless treated properly, these techniques can severely degrade the lifetime of the NVMs.

In this paper, we propose a data diversion based technique called SeNonDiv using a DRAM-PCM based hybrid main memory. SeNonDiv places the security-sensitive data pages coming from secondary storage in the volatile DRAM portion of main memory. The other non-sensitive data pages are stored in the remaining memory consisting of the remaining DRAM

region and the PCM. The sensitive data contents stored in DRAM are lost once the system is powered off, and the threats associated with prolonged data retention in PCM are nullified. Also, avoidance of encryption helps in improving the lifetime of PCM. Experimental results show that apart from providing security against the stolen DIMM attacks, SeNonDiv improves the lifetime of PCM by 33% and 140% over ParEncr and FullEncr, respectively. Thus, intelligent improvisation of DRAM-PCM based hybrid memories can provide resourceful solutions in protecting data-privacy without exacerbating the lifetime issues inherent to the non-volatile memories.

REFERENCES

[1] R. A. Bheda et al., "Energy efficient Phase Change Memory based main memory for future high performance systems," in *2011 International Green Computing Conference and Workshops*, July 2011, pp. 1–8.

[2] S. Swami and K. Mohanram, "ACME: Advanced counter mode encryption for secure non-volatile memories," in *2018 55th ACM/ESDA/IEEE Design Automation Conference (DAC)*. IEEE, 2018, pp. 1–6.

[3] A. Nath and H. K. Kapoor, "WELCOMF: wear leveling assisted compression using frequent words in non-volatile main memories," in *Proceedings of the ACM/IEEE International Symposium on Low Power Electronics and Design*, 2020, pp. 157–162.

[4] V. Young, P. J. Nair, and M. K. Qureshi, "DEUCE: Write-efficient encryption for non-volatile memories," *ACM SIGARCH Computer Architecture News*, vol. 43, no. 1, pp. 33–44, 2015.

[5] P. Zuo, Y. Hua, M. Zhao, W. Zhou, and Y. Guo, "Improving the performance and endurance of encrypted non-volatile main memory through deduplicating writes," in *IEEE/ACM MICRO)*. IEEE, 2018, pp. 442–454.

[6] W. Wei et al., "HAP: Hybrid-Memory-Aware Partition in Shared Last-Level Cache," *ACM Trans. Archit. Code Optim.*, vol. 14, no. 3, pp. 24:1–24:25, Sep. 2017.

[7] D. Zhang et al., "Write-back aware shared last-level cache management for hybrid main memory," in *Proceedings of the 53rd Annual Design Automation Conference*, ser. DAC '16. New York, NY, USA: ACM, 2016, pp. 172:1–172:6.

[8] D. Chen, H. Jin, X. Liao, H. Liu, R. Guo, and D. Liu, "MALRU: Miss-penalty aware lru-based cache replacement for hybrid memory systems," in *Proceedings of the Conference on Design, Automation & Test in Europe*, ser. DATE '17. 3001 Leuven, Belgium, Belgium: European Design and Automation Association, 2017, pp. 1086–1091. [Online]. Available: http://dl.acm.org/citation.cfm?id=3130379.3130637

[9] A. Nath, S. Agarwal, and H. K. Kapoor, "Reuse distance-based victim cache for effective utilisation of hybrid main memory system," *ACM Transactions on Design Automation of Electronic Systems (TODAES)*, vol. 25, no. 3, pp. 1–32, 2020.

[10] J. Daemen and V. Rijmen, "Reijndael: The advanced encryption standard." *Dr. Dobb's Journal: Software Tools for the Professional Programmer*, vol. 26, no. 3, pp. 137–139, 2001.

[11] N. Binkert et al., "The gem5 simulator," *ACM SIGARCH Computer Architecture News*, vol. 39, no. 2, pp. 1–7, 2011.

[12] M. Poremba et al., "NVMain: An architectural-level main memory simulator for emerging non-volatile memories," in *ISVLSI*. IEEE, 2012, pp. 392–397.

[13] S. Rajarajan et al., "A study on the challenges and prospects of PCM Based Main Memory Architectures," *Middle-East Journal of Scientific Research*, vol. 18, no. 6, pp. 788–795, 2013.

[14] B. C. Lee, E. Ipek, O. Mutlu, and D. Burger, "Architecting phase change memory as a scalable dram alternative," in *Proceedings of the 36th annual international symposium on Computer architecture*, 2009, pp. 2–13.

[15] F. McKeen et al., "Innovative instructions and software model for isolated execution." *Hasp@ isca*, vol. 10, no. 1, 2013.

[16] D. Aucsmith, "Tamper resistant software: An implementation," in *International Workshop on Information Hiding*. Springer, 1996, pp. 317–333.

[17] J. M. McCune, B. J. Parno, A. Perrig, M. K. Reiter, and H. Isozaki, "Flicker: An execution infrastructure for tcb minimization," in *Proceedings of the 3rd ACM SIGOPS/EuroSys European Conference on Computer Systems 2008*, 2008, pp. 315–328.

[18] H. K. Kapoor, G. B. Rao, S. Arshi, and G. Trivedi, "A security framework for noc using authenticated encryption and session keys," *Circuits, Systems, and Signal Processing*, vol. 32, no. 6, pp. 2605–2622, 2013.

[19] S. Swami, J. Rakshit, and K. Mohanram, "Secret: Smartly encrypted energy efficient non-volatile memories," in *Proceedings of the 53rd Annual Design Automation Conference*, 2016, pp. 1–6.

[20] M. Jalili and H. Sarbazi-Azad, "Endurance-aware security enhancement in non-volatile memories using compression and selective encryption," *IEEE Transactions on Computers*, vol. 66, no. 7, pp. 1132–1144, 2017.

[21] S. Ghosh, M. N. I. Khan, A. De, and J. Jang, "Security and privacy threats to on-chip non-volatile memories and countermeasures," in *ICCAD 2016, Austin, TX, USA, November 7-10, 2016*, F. Liu, Ed. ACM, 2016, p. 10.

[22] S. Motaman, S. Ghosh, and N. Rathi, "Cache bypassing and checkpointing to circumvent data security attacks on sttram," *IEEE Transactions on Emerging Topics in Computing*, 2017.

[23] N. Rathi, S. Ghosh, A. Iyengar, and H. Naeimi, "Data privacy in non-volatile cache: Challenges, attack models and solutions," in *ASP-DAC 2016, Macao, Macao, January 25-28, 2016*. IEEE, 2016, pp. 348–353.

[24] K. Sajeesh and H. K. Kapoor, "An authenticated encryption based security framework for noc architectures," in *2011 International Symposium on Electronic System Design*. IEEE, 2011, pp. 134–139.

[25] D. M. Ancajas, K. Chakraborty, and S. Roy, "Fort-nocs: Mitigating the threat of a compromised noc," in *Proceedings of the 51st Annual Design Automation Conference*, 2014, pp. 1–6.

[26] J. L. Henning, "Spec cpu2006 benchmark descriptions," *ACM SIGARCH Computer Architecture News*, vol. 34, no. 4, pp. 1–17, 2006.

[27] W. Shi, H.-h. S. Lee, M. Ghosh, C. Lu, and A. Boldyreva, "High efficiency counter mode security architecture via prediction and precomputation," in *32nd International Symposium on Computer Architecture (ISCA'05)*. IEEE, 2005, pp. 14–24.

[28] D. Lee, D. Jung, I. T. Fang, C.-C. Tsai, and R. A. Popa, "An off-chip attack on hardware enclaves via the memory bus," in *29th {USENIX} Security Symposium ({USENIX} Security 20)*, 2020.

[29] M. Gruhn and T. Müller, "On the practicability of cold boot attacks," in *2013 International Conference on Availability, Reliability and Security*. IEEE, 2013, pp. 390–397.

[30] S. Lindenlauf, H. Höfken, and M. Schuba, "Cold boot attacks on ddr2 and ddr3 sdram," in *2015 10th International Conference on Availability, Reliability and Security*. IEEE, 2015, pp. 287–292.

[31] J. Y. Hur, "Representing contiguity in page table for memory management units," in *2017 IEEE 11th International Symposium on Embedded Multicore/Many-core Systems-on-Chip (MCSoC)*, 2017, pp. 21–28.