Understanding and Strengthening Security for CXL Memory

Recent advances in large ML models, such as Generative Large Language Models (LLMs) and Mixture of Experts (MoE), require cloud data centers to increase memory bandwidth and capacity to unlock their performance [1, 2]. Compute Express Link (CXL), an open standard, was developed to provide a high-speed, low-latency, cache-coherent interconnect for processors, accelerators, and memory expansion. The release of CXL 3.0 in August 2022 [3] introduced support for fabric topologies that connect multiple hosts to shared GFAM (Global Fabric Attached Memory) devices. This supports disaggregated memory, where an arbitrary number of endpoints connected in any topology can request, use, and coherently share varying amounts of memory [4], offering a cost-effective and high-performance solution to expand memory bandwidth and capacity.

Data security in data centers is critical for users, yet current solutions fail to address the complexities in the distributed CXL memory environment. Trusted Execution Environments (TEEs) offer hardware-protected enclaves that ensure program integrity and confidentiality [5, 6]. However, these enclaves are limited to individual processors or virtual machines, and do not accommodate the migration of data across different virtual machines, processors, and server nodes in a CXL memory setting.

CXL memory actually poses some principled challenges for protection in comparison to traditional memory in several ways:

1. The architecture of CXL memory is inherently extensible. Developing a security abstraction supporting dynamic resource allocation and deallocation cross server nodes remains a challenge.

2. Current methodologies fail to protect data in use across different virtual machines, processors, and server nodes, with existing solutions restricted to single processor or virtual machine environments.

3. The shared nature of data in CXL memory among multiple virtual machines, processors, and server nodes introduces unique security risks. Vulnerabilities in one process can compromise the security of others, highlighting the emergence of cross-process, cross-processor, and cross-server security vulnerabilities specific to the CXL memory model.

The combination of extensibility, migratability, and shared access distinctly positions CXL memory relative to traditional memory and storage, not only in terms of vulnerability and the potential consequences of security breaches but also in offering opportunities for innovative security solutions.

Keywords: CXL Memory, Security, Architecture, Operating Systems, Compiler

Intellectual Merit The goal of this proposed research is to significantly advance the understanding of CXL memory protection, establish the foundation for *CXL-conscious memory protection*, create a set of novel techniques for protecting CXL memory. Specifically, this proposed research brings architecture, programming language/model, and operating system techniques to establish *CXL-conscious memory protection* as a new paradigm of memory protection to fit the distinctive properties of CXL memory. It consists of three major thrusts:

- Understanding of CXL security and building up a conceptual framework of *CXL-conscious memory protection*. (i) Benchmarks. (ii) Relations with various attack models. (iii) Implications to each level of the security stack. (iv) Key concepts.

- Establishing Extensible and Migratable Protection as a principled approach to CXL memory protection. (i) New abstraction; (ii) architecture design; (iii) OS support; (iv) compiler support.

- New protocols for validating, recovering data modifications from different users.

Broader Impact As CXL memory becomes mainstream in computing devices, this research is crucial for ensuring CXL memory safety. Its success could help mitigate potential security breaches that may result in substantial losses across various sectors including industry, defense, scientific research, and health, among others that depend on future computing devices.

The PI will integrate the research into education through curriculum development, virtual machine platforms, workshops, and tutorials. The PI has a history of mentoring Ph.D., master's, and undergraduate students from underrepresented groups. The PI will continue to promote diversity and engage in outreach activities. As a long-term collaborator with Google, the PI will continue to pursue opportunities to disseminate and apply the results from this work through collaborations with industrial colleagues.

References

[1] OpenAI. Chatgpt, 2024.

[2] Zhuohan Li, Lianmin Zheng, Yinmin Zhong, Vincent Liu, Ying Sheng, Xin Jin, Yanping Huang, Zhifeng Chen, Hao Zhang, Joseph E Gonzalez, et al. {AlpaServe}: Statistical multiplexing with model parallelism for deep learning serving. In *17th USENIX Symposium on Operating Systems Design and Implementation (OSDI 23)*, pages 663–679, 2023.

[3] Compute Express Link. Compute express link, 2024.

[4] Samuel W Stark, A Theodore Markettos, and Simon W Moore. How flexible is cxl's memory protection? replacing a sledgehammer with a scalpel. *Queue*, 21(3):54–64, 2023.

[5] Intel. Software guard extensions programming reference. `https://software.intel.com/sites/default/files/managed/48/88/329298-002.pdf`. Online; accessed August, 2020.

[6] Amd sev-snp. `https://www.amd.com/system/files/TechDocs/SEV-SNP-strengthening-vm-isolation-with-integrity-protection-and-more.pdf`.