로우해머 공격 방어 기법에 관한 연구

하승진[1], 백윤흥[1]
[1]서울대학교 전기·정보공학부, 서울대학교 반도체 공동연구소
sjha@sor.snu.ac.kr, ypaek@snu.ac.kr

A Study on Defense of the Rowhammer Attack

Seung-jin Ha[1], Yun-heung Paek[1]
[1]Dept. of Electrical and Computer Engineering and Inter-University Semiconductor Research Center(ISRC), Seoul National University

요 약

컴퓨터 보안의 위협 중 하나인 로우해머 공격은 DRAM 메모리 모듈에 영향을 미치며, 인접 셀에서 “비트 플립”이 발생하여 중요한 데이터에 무단으로 접근하거나 시스템을 손상시킬 수 있다. 하드웨어 기반 방어 기법은 메모리 컨트롤러 및 메모리 모듈 기반으로 나뉘며, 소프트웨어 기반 방어 기법은 기계 학습 알고리즘을 사용하여 공격을 감지하거나 예측하여 방지한다. 본 논문은 로우해머 공격과 그 대응 방안에 대한 연구 동향을 설명한다.

1. 서론

최근 몇 년 동안, 기술의 급속한 발전으로 인해 스마트폰, 노트북 및 기타 혁신적인 시스템이 등장하면서 우리의 일상 생활에서 필수불가결한 존재가 되었다. 이러한 시스템은 정보 접근 및 다른 사람들과 소통이 쉬워졌지만, 이러한 기술 발전과 함께 시스템 보안에 대한 다양한 위협이 존재한다. 이 중에서도 로우해머(Rowhammer) 공격은 주요 위협 중 하나이다.

로우해머 공격[1]은 현대 메모리 시스템의 취약성을 이용하는 메모리 기반 공격으로, 특정 메모리 행에 반복적으로 접근하여 인접 메모리 행의 내용을 변경함으로써 작동한다. 이러한 공격으로 중요한 데이터에 무단으로 접근하거나 전체 시스템을 제어하는 것이 가능하다. 특히, 클라우드 컴퓨팅의 증가로 인해 로우해머 공격은 여러 시스템의 보안을 동시에 손상시킬 수 있기 때문에 더욱 위험하다.

이러한 로우해머 공격을 방지하기 위해 하드웨어 및 소프트웨어 기반의 여러 방어 기법이 제안되었다. 하드웨어 기반 솔루션에는 오류 수정 코드(Error-Correcting Codes, ECC) 사용 또는 행의 주기적인 새로고침이 포함된다. 반면에, 소프트웨어 기반 솔루션은 기계 학습 알고리즘을 사용하여 메모리 접근 패턴을 감시하거나 공격을 예측하고 방지하는 것을 포함한다.

본 논문은 로우해머 공격에 대한 개요과 그 영향을 완화하기 위해 제안된 다양한 방어 기법에 대한 연구 동향을 알아보는 것을 목표로 한다.

2. 로우해머 공격

로우해머 공격[1]은 최신 컴퓨터의 DRAM 메모리 모듈에 영향을 미치는 보안 취약점으로, DRAM 셀의 물리적 근접성으로 인해 발생한다. 특정 행에 반복적으로 접근할 때 인접 셀에서 “비트 플립(bit flip)”이 발생할 수 있으며, 이를 이용하여 공격자는 중요한 데이터에 무단으로 접근하거나 시스템을 손상시킬 수 있다. 로우해머 공격은 소프트웨어를 통해 수행될 수 있어[2] 상당한 위협으로 작용한다.

이 공격은 2014년에 처음 제안되었으며, 이후 여러 변형이 발견되었다. 이에는 인접한 두 행의 셀을 대상으로 하는 “양면 로우해머” 공격[3]과 네트워크를 통해 원격으로 수행할 수 있는 “쓰로우해머(Throwhammer)” 공격[4]이 포함된다.

2. 로우해머 공격에 대한 방어 기법

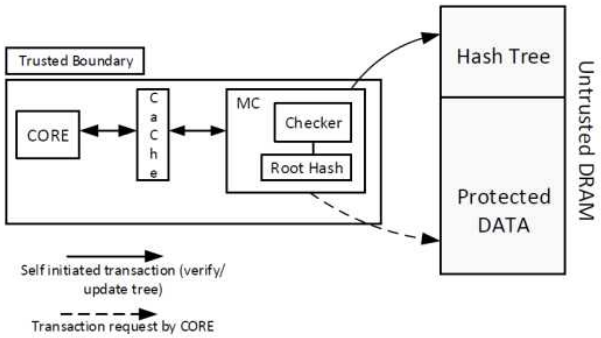
2-1. 하드웨어 기반 방어

로우해머 공격을 막기 위한 하드웨어 기반 방어 기법은 최근 몇 년 동안 많은 연구가 이루어졌다. 이러한 방어 기법은 하드웨어 수준에서 로우해머 공

격을 탐지하고 완화하는 것을 목표로 한다.

한 가지 방법은 하드웨어 카운터를 사용하여 DRAM의 새로고침 주기 동안 행별로 활성화된 횟수를 셀 수 있다. 이 방법에는 “타겟 행 리프레시” (Target Row Refresh, TRR)[5] 및 “적응형 행 리프레시” (Adaptive Row Refresh, ARR)[6]와 같은 기술이 제안되었으며, 이는 비트 플립을 방지하기 위해 대상 행에 인접한 행을 주기적으로 새로 고치는 것을 포함한다.

또 다른 방법으로 데이터 무결성 검사를 통해 비트 플립을 감지하거나 수정할 수 있다. 이 방법에는 로우해머 공격으로 인한 단일 비트 오류를 감지하고 수정할 수 있는 오류 수정 코드(Error-Correcting Code, ECC)[7] 메모리를 추가하는 것이 포함된다. 또한, 메모리 컨트롤러가 잠재적으로 로우해머 공격을 받은 행을 파악하여 비트 플립을 감지할 수 있다.[8]



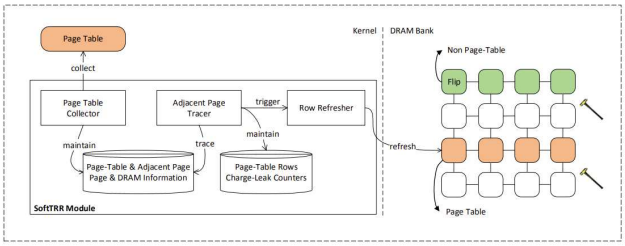
(그림 1) 메모리 컨트롤러를 활용한 비트 플립 감지[8]

전반적으로 하드웨어 기반 방어는 시스템 성능에 영향을 미치지 않고 빠르고 효과적인 완화를 제공할 수 있기 때문에 로우해머 취약점에 대한 유망한 솔루션을 제공한다. 이러한 방어 기법들은 로우해머 공격을 막는 데 매우 유용하며, 계속해서 연구와 개발이 이루어질 것으로 예상된다.

2-2. 소프트웨어 기반 방어

로우해머 공격에 대한 소프트웨어 기반 방어 기법이 최근 몇 년 동안 활발히 연구되고 있다. 이러한 기법은 소프트웨어 기반 공격 탐지 및 메모리 영역 분리와 같은 기술을 활용하여 로우해머 공격을 탐지하고 방지하는 데 중점을 둔다.

한 가지 접근 방식은 잠재적인 로우해머 공격에 대항하여 특정 메모리 위치에서 비트 플립을 의도적으로 유도하는 것을 탐지하기 위해 메모리 접근 방식을 감시하는 것이다. 한 위치에 너무 자주 접근하거나[9] CPU 캐시 미스가 너무 많이 발생하는 경우[10] 로우해머 공격을 나타낼 수 있으며, 공격을 완화하기 위해 적절한 조치를 취할 수 있다.



(그림 2) SoftTRR 구조도[11]

예를 들어, SoftTRR[11]은 페이지 테이블 항목의 비트를 이용하여 페이지 테이블을 담당하는 행에 인접한 행에 대한 접근을 지속적으로 추적한다. 이러한 추적된 행의 접근 횟수가 미리 설정된 임계값에 도달하면, 해당하는 페이지 테이블이 있는 행에 대해 새로 고침이 수행된다. 이러한 방법은 하드웨어나 운영체제를 수정할 필요 없이 소프트웨어 기반으로 구현할 수 있으며, 최소한의 오버헤드로 로우해머 공격을 탐지하는 데 효과적인 것으로 나타났다.

또 다른 유망한 접근 방식은 취약한 메모리 영역을 분리하여 비트 플립이 시스템의 다른 부분에 영향을 미치지 않도록 하는 로우해머 공격에 대한 방어 기법이다.[12] 이 방법은 로우해머 공격이 종종 특정 메모리 영역을 대상으로 한다는 사실을 활용한다. 이러한 영역을 격리하고 공격자의 접근 능력을 제한함으로써 로우해머 공격의 영향을 줄일 수 있다.

소프트웨어 기반 방어는 로우해머 취약점을 완화하기 위한 유연하고 비용 효율적인 접근 방식을 제공하며, 하드웨어 기반 방어와 함께 사용하여 보호 강화에 효과적이다.

3. 결론

결론적으로 로우해머 공격은 메모리 시스템의 취약점을 악용하기 때문에 현대 컴퓨팅 시스템의 보안에 심각한 위협이 된다. 최근 하드웨어 및 소프트웨어 솔루션 측면에서 이 공격의 영향을 완화하기 위한 다양한 방어 기법이 제안되었다. 주기적인 새로고침이나 비트 플립 감지 및 수정과 같은 하드웨어 기반 솔루션은 로우해머 공격의 영향을 줄이는 데

효과적이었다. 반면, 소프트웨어 기반 공격 탐지 및 메모리 영역 분리와 같은 소프트웨어 기반 솔루션은 공격을 탐지하고 방지하는 데 있어 가능성을 보여주었다. 그러나 기술이 발전함에 따라 로우해머 공격은 계속 진화하고 더욱 정교해질 수 있다. 따라서 향후 연구에서는 로우해머 공격의 위협을 완화하고 컴퓨팅 시스템의 보안을 보장하기 위한 새롭고 혁신적인 솔루션을 계속해서 탐구해야 한다.

4. ACKNOWLEDGEMENT

이 논문은 2023년도 BK21 FOUR 정보기술 미래인재 교육연구단에 의하여 지원되었음. 이 논문은 2023년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No.2021-0-00528, 하드웨어 중심 신뢰계산기반과 분산 데이터보호박스를 위한 표준 프로토콜 개발)

참고문헌

[1] Kim, Y., Daly, R., Kim, J., & Fallin, C. "Flipping bits in memory without accessing them: An experimental study of DRAM disturbance errors." In ACM SIGARCH Computer Architecture News, 42, 3, 361-372, 2014.

[2] Gruss, D., Lipp, M., Schwarz, M., & Mangard, S. "Rowhammer.js: A remote software-induced fault attack in JavaScript." In Proceedings of the 13th International Conference on Detection of Intrusions and Malware, and Vulnerability Assessment, San Sebastián, Spain, 2016.

[3] Gruss, D., Lipp, M., Schwarz, M., Genkin, D., Juffinger, J., O'Connell, S., ... & Yarom, Y. "Another flip in the wall of rowhammer defenses." In 2018 IEEE Symposium on Security and Privacy (SP), 2018, 245-261.

[4] Tatar, A., Konoth, R. K., Athanasopoulos, E., Giuffrida, C., Bos, H., & Razavi, K. "Throwhammer: Rowhammer attacks over the network and defenses." In 2018 USENIX Annual Technical Conference, 2018, 213-226.

[5] Marazzi, M., Jattke, P., Solt, F., & Razavi, K. "PROTRR: Principled yet optimal in-DRAM target row refresh." In 2022 IEEE Symposium on Security and Privacy (SP), 2022, 735-753.

[6] M. Son, H. Park, J. Ahn, and S. Yoo, "Making DRAM stronger against row hammering," in Design Automation Conference, 2017, pp. 1 - 6.

[7] W. Ryan and S. Lin, Channel codes: classical and modern. Cambridge university press, 2009.

[8] Vig, S., Bhattacharya, S., Mukhopadhyay, D., & Lam, S. K. "Rapid detection of Rowhammer attacks using dynamic skewed hash tree." In Proceedings of the 7th International Workshop on Hardware and Architectural Support for Security and Privacy, 2018, 1-8.

[9] Z. Zhang, Y. Cheng, M. Wang, W. He, W. Wang, N. Surya, Y. Gao, K. Li, Z. Wang, and C. Wu, "Softtrr: Protect page tables against rowhammer attacks using software-only target row refresh," arXiv preprint arXiv:2102.10269, 2021.

[10] Z. B. Aweke, S. F. Yitbarek, R. Qiao, R. Das, M. Hicks, Y. Oren, and T. Austin, "ANVIL: Software-based protection against next generation rowhammer attacks," in Architectural Support for Programming Languages and Operating Systems, 2016, pp. 743 - 755.

[11] Zhang, Z., Cheng, Y., Wang, M., He, W., Wang, W., Nepal, S., ... & Wu, C. "{SoftTRR}: Protect Page Tables against Rowhammer Attacks using Software-only Target Row Refresh." In 2022 USENIX Annual Technical Conference (USENIX ATC 22), 2022, 399-414.

[12] F. Brasser, L. Davi, D. Gens, C. Liebchen, and A.-R. Sadeghi, "CAn't Touch This: Software-only mitigation against rowhammer attacks targeting kernel memory," in USENIX Security Symposium, 2017.