정 보 보 호 학 회 지
제 33 권 제5호, 2023. 10

분리 메모리 시스템의 보안 기술 연구 동향

용 예 원*, 김 창 대**, 김 태 훈**

요 약

최근 대규모 인공지능 데이터 처리를 위한 메모리 용량 한계 극복과 데이터센터의 메모리 효율성 향상을 위해 분리 메모리 시스템 기술이 각광 받고 있다. 그런데, 분리 메모리 시스템은 컴퓨팅 노드의 메모리 외의 디바이스 혹은 원격 노드의 메모리를 활용해 확장된 메모리를 제공하기 때문에 새로운 보안 위협이 발생한다. 본 논문은 분리 메모리 시스템의 보안 위협을 분석하고, 분리 메모리 시스템에 적용 가능한 보안 기술의 최근 연구 동향을 소개한다.

Ⅰ. 서 론

사용자가 생성하는 데이터의 급속한 증가와 함께, 인공지능이 생성해내는 데이터까지 더해지면서 세계의 데이터가 폭증하고 있다. 이에 더해, 응용들도 점점 복잡해지면서, 응용들의 메모리 요구량이 나날이 커지고 있다. 하지만, 시스템 내에 장착 가능한 DRAM 용량은 기술의 한계로 인해 더디게 증가하고 있어서, 시스템 내 메모리가 부족해지는 현상이 발생하고 있다. 특히 클라우드 환경에서 메모리 부족으로 인해 시스템 활용률이 정체되는 현상도 목격되는 등 시스템 메모리 용량 문제가 중요하게 대두되고 있다 [1].

이와 같은 메모리 부족 현상을 해결하기 위해 분리 메모리 시스템 기술 (Disaggregated Memory System)이 각광을 받고 있다. 분리 메모리 기술은 다른 서버의 메모리나 고성능 SSD 등의 디바이스를 활용해 시스템의 메모리를 확장하는 기술이다 [2, 3, 4, 5]. 인피니밴드에서 400Gbps 속도의 네트워크 기술이 발표되고 Compute Express Link (CXL)이 현실화되는 등 분리 메모리 기술에 필요한 인터커넥트 기술이 발전하면서 분리 메모리 기술에 대한 기대감은 더욱 높아지고 있다 [6].

하지만, 분리 메모리 시스템은 기존에 없던 보안 위협을 야기한다. 서버들 간의 메모리가 독립적으로 작동했던 기존 기술과는 달리, 분리 메모리 시스템은 여러 시스템의 메모리가 함께 사용되기 때문에, 한 시스템의 보안 위협이 다른 시스템으로 퍼져나갈 수 있기 때문이다. 하나의 시스템이 침해당했을 때 분리 메모리 시스템으로 연결된 다른 시스템의 데이터를 탈취하거나 변조할 수 있게 되고, 그를 통해 다른 시스템으로 침입하는 것도 더 쉬워지게 된다.

본 논문에서는 이와 같은 분리 메모리 시스템의 보안 문제에 대한 연구 동향을 알아보고자 한다. 분리 메모리 시스템의 보안 위협에 대해 분석하고, 최근 발표되고 있는 보안 기술에 대한 논문 중 분리 메모리 시스템에 적용할 수 있는 기술들에 대해 알아보고자 한다. 그리고 분리 메모리 시스템의 보안을 제공하기 위한 향후 연구 주제에 대해서도 논한다.

Ⅱ. 배경지식

이 장에서는 분리 메모리 시스템에서의 보안 기술 연구 동향을 살펴보기에 앞서 관련 연구를 설명하기 위한 배경지식에 대해서 알아보고자 한다.

2.1. 분리 메모리 시스템

분리 메모리 시스템 (Disaggregated Memory System)은 다른 서버의 메모리나 고성능 SSD 등의

본 연구는 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2018-0-00503, 메모리 중심 차세대 컴퓨팅 시스템 구조 연구)
* 한국전자통신연구원 고성능컴퓨팅시스템연구실, 충남대학교 컴퓨터공학과 (학연협동과정학생, yongye1@etri.re.kr)
** 한국전자통신연구원 고성능컴퓨팅시스템연구실 (선임연구원, cdkim@etri.re.kr, taehoon.kim@etri.re.kr)

디바이스를 활용해서 시스템 메모리를 확장하는 기술이다. 분리 메모리 시스템 내 노드들은 컴퓨팅 노드와 메모리 노드로 구분할 수 있으며, 하나의 노드가 컴퓨팅 노드와 메모리 노드의 역할을 동시에 수행할 수도 있다. 컴퓨팅 노드에서 응용 혹은 가상머신이 실행되며, 컴퓨팅 노드 내의 메모리가 부족할 경우 분리 메모리 시스템의 시스템 소프트웨어나 전용 하드웨어가 메모리 노드의 메모리를 활용할 수 있도록 해준다. 이때, 자주 사용하는 데이터들을 컴퓨팅 노드 내의 메모리에 두고 자주 사용하지 않는 데이터들을 메모리 노드로 보내는 방식이 많이 쓰인다.

분리 메모리는 90년대부터 제시되었으나 [7], 네트워크 속도 발전과 더불어, 다른 서버의 메모리를 해당 노드의 CPU 개입 없이 접근할 수 있도록 해주는 Remote Direct Memory Access (RDMA) 기술이 등장하면서 주목받기 시작했다. Infiniswap은 Linux의 스왑 시스템을 수정하여 RDMA 기반의 메모리 확장 기술을 제시했다 [2]. Fastswap은 Linux의 프론트-스왑 시스템을 활용함으로써 성능 개선을 이루었고, 분리 메모리가 클러스터의 처리량을 높일 수 있음을 보였다 [3]. DCM은 가상머신을 위한 분리 메모리 시스템으로, Linux의 스왑을 수정하지 않고 직접 메모리를 관리함으로써 효율을 높였다 [4]. 또한, 다른 서버의 메모리와 더불어 고성능 SSD로의 메모리 확장도 지원한다. 최근에는 KAIST의 정명수 교수 연구실을 중심으로 CXL 기술을 활용한 분리 메모리 기술이 개발되고 있다 [8].

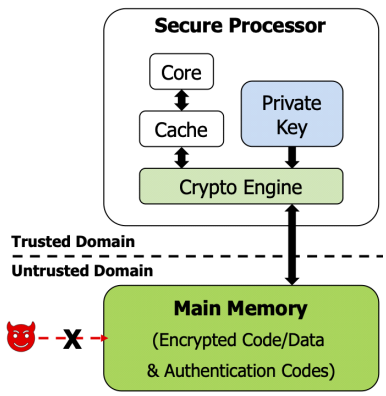
분리 메모리 시스템이 메모리 용량 문제를 해결해 주지만, 메모리 노드로의 접근 속도가 상대적으로 느리기 때문에 성능 문제가 생긴다. 이를 해결하기 위한 연구도 많이 진행되고 있다. Leap은 분리 메모리 시스템을 위한 프리페치 기술을 제시하여 컴퓨팅 노드 내의 메모리 접근 비율을 더 높이고자 했으며 [9], AIFM은 응용 라이브러리 형태의 분리 메모리 시스템을 개발하여 응용의 특성을 활용한 성능 최적화를 시도했다 [10]. DiLOS는 라이브러리OS를 사용하여 가상메모리 (Virtual Memory) 기반 분리 메모리 시스템의 성능 부하를 줄이는 방법을 제시했고 [11], DeHype은 메모리 노드로 접근하기 위한 I/O 속도를 최적화하기 위한 여러 시스템 소프트웨어 기술을 제시했다 [5].

2.2. 보안 프로세서

보안 프로세서 (Secure Processor)는 신뢰 컴퓨팅 기반 (Trusted Computing Base, TCB)을 프로세서로 최소화하고, 프로세서 내부에 수행되는 연산 보호 및 연산을 수행하는 데이터에 대한 기밀성과 무결성을 보장해주는 기술이다 [12, 13, 14]. 그림 1과 같이 프로세서 내의 보안키와 보안 모듈을 통해서 프로세서 외부로 나가는 데이터에 대해 암호화를 수행하고, 데이터의 해시 값 및 메시지 인증 코드 (Message Authentication Code, MAC) 정보를 사용하여 데이터의 무결성을 검증한다.

메모리와 통신하는 데이터의 빠른 암호화를 위해서 주로 카운터 모드 암호화 기술을 사용한다 [15]. 데이터 전송과 암호화/복호화를 병렬적으로 수행하기 위해서, 데이터의 주소나 카운터를 활용하여 암호화된 패드를 미리 준비하고, 데이터가 프로세서 내로 전송되면 XOR연산을 통해서 빠른 복호화를 수행하는 기술이다.

데이터의 무결성을 검증하기 위해서 주로 데이터의 해시 값이나 데이터의 버전넘버를 이용해서 머클 트리를 구성한다 [12, 16]. 이는 신뢰할 수 있는 보안 프로세서 내부에 모든 데이터의 해시 값들을 저장할 수 없기 때문에 해시 값을 트리 형태로 구성하고, 머클 트리의 루트 노드만은 항상 신뢰 영역인 보안 프로세서에 저장하는 기술이다. 데이터의 무결성 확인을 위해 머클 트리의 해시 값을 전송된 데이터의 해시 값과 비교하여 검증한다.

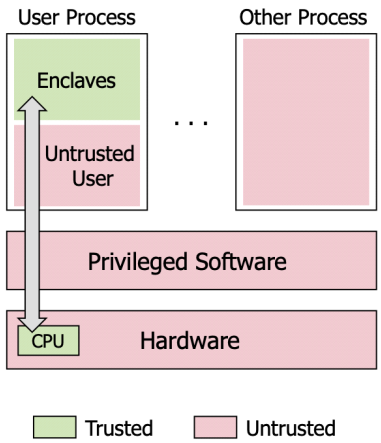


〔그림 1〕 보안 프로세서의 구성

2.3. 신뢰 실행 환경

Intel Secure Guard Extension (SGX), ARM TrustZone, AMD Secure Encrypted Virtualization (SEV), RISC-V Keystone 등 최근 프로세서는 보안 기능을 바탕으로 하드웨어 기반 신뢰 실행 환경 (Trusted Execution Environment, TEE)을 제공하고 있다.

그림 2와 같이 CPU가 제공하는 TEE를 통해서 운영체제, 하이퍼바이저와 같은 높은 권한의 시스템 소프트웨어 및 다른 컨텍스트로부터 실행 컨텍스트를 격리시킴으로써 보호된 실행환경을 제공한다. 또한, 격리된 컨텍스트에서 사용하는 데이터는 2.2장에서 설명한 보안 프로세서에서 제공하고 있는 데이터 보호 기법과 유사하게 데이터 암호화 및 무결성 검증 기능을 제공한다. 이와 더불어 로컬 및 원격 인증 (Attestation)을 통해서 해당 프로세서에서 수행하는 환경에 대한 검증 기능을 제공한다.



〔그림 2〕 TEE의 예 (e.g., Intel SGX)

Ⅲ. 분리 메모리 시스템의 보안 위협 모델 분석

분리 메모리 시스템은 다른 서버의 메모리를 활용하거나 호스트의 메모리가 아닌 다른 디바이스를 활용하여 메모리를 확장하는 방식을 사용한다. 따라서, 기존에 보안 프로세서나 TEE가 가정하는 단일 호스트 내의 데이터 보호 이외의 새로운 보안 위협이 존재한다.

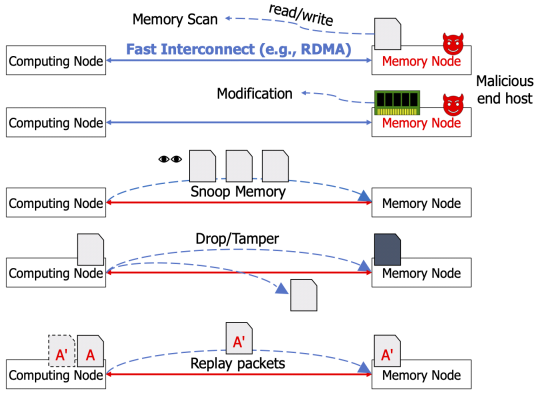
이 장에서는 분리 메모리 시스템에서 발생할 수 있는 보안 위협에 대해서 알아보고자 한다.

3.1. 메모리 노드에서의 위협

분리 메모리 시스템에서는 다양한 위협이 존재한다. 특히, 메모리 노드에 데이터를 저장하고 다수의 컴퓨팅 노드가 해당 메모리 노드를 공유하기 때문에, 악의적인 컴퓨팅 노드 혹은 악의적인 관리자에 의해서 위협에 노출될 수 있고, 이에 따라 메모리 노드가 악의적으로 행동할 수 있다. 따라서, 그림 3의 첫 번째, 두 번째 경우와 같이, 메모리 노드에 존재하는 데이터에 대한 기밀 누출 혹은 변조가 발생할 수 있다.

첫 번째 공격은 데이터의 기밀성과 관련된 공격으로 악의적인 공격자가 메모리 노드에 저장되어있는 컴퓨팅 노드의 데이터에 접근 가능하다는 것이다. 이 공격을 통해서 컴퓨팅 노드가 사용하는 데이터의 정보 혹은 컴퓨팅 노드가 어떠한 동작을 수행하는지 파악할 수 있게 된다. 또한, 악의적인 다른 컴퓨팅 노드가 해당 데이터에 접근할 수 있도록 허용할 수 있다.

두 번째 공격은 데이터의 무결성과 관련된 공격이다. 악의적인 공격자는 메모리 노드에 저장되어있는 컴퓨팅 노드의 데이터를 변조시킬 수 있으며, 이렇게 변조된 데이터의 사용으로 컴퓨팅 노드를 위협에 노출할 수 있다.



〔그림 3〕 분리 메모리 시스템에서의 위협 모델

3.2. 통신 채널에서의 위협

컴퓨팅 노드와 메모리 노드 사이에 통신을 위한 인터커넥트에서도 위협이 존재한다. 예를 들어, 원격 노드에 존재하는 메모리 노드를 가정해보자. 원격 노드로의 빠른 접근은 RDMA 기술을 통해서 수행할 수

있다. 현재의 RDMA 기술은 보안보다는 성능을 위해서 사용되고 있기 때문에 암호화나 인증에 대한 기능을 제공하고 있지 않다. 하지만, 접근 토큰 방식을 통해서 접근 격리를 제공하며, 권한이 없는 시스템의 접근에 대해서는 보호할 수 있는 기능을 제공한다 [17].

이에 따라서, 그림 3의 아래 세 가지 경우와 같이 해당 인터페이스를 통해서 전달되는 패킷을 스캔하고 새로운 패킷을 주입하는 방식으로 데이터를 누출시킬 수 있으며, 변조 혹은 패킷을 누락 시킬 수 있다. 또한, 패킷에 대한 무결성 검증을 하더라도 이전에 검증된 데이터를 새로운 데이터와 교체하여 보내는 방식을 사용하여 리플레이 공격을 수행할 수 있다.

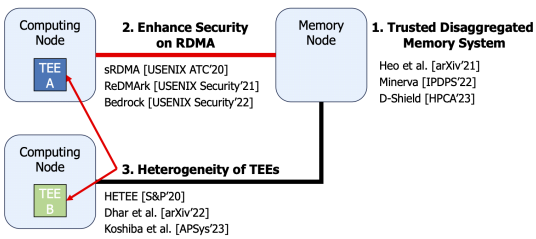
3.3. 부채널 공격

부채널 공격 (Side-channel Attacks)도 가능하다. 컴퓨팅 노드에서 자주 사용하는 라이브러리를 수행하는 경우 해당 문맥을 파악하고, 원격의 메모리 노드에 있는 데이터에 대한 접근 패턴을 파악하여 해당 데이터의 내용도 유추가 가능하다 [18]. 또한, RDMA 기술을 사용하는 경우, RDMA의 접근 토큰이 평문 형식으로 전달되기 때문에 통신 패킷이 어느 위치로 전송되는 되는지 파악이 가능하다.

또한, 악의적인 공격자는 다양한 형태의 Covert channel 공격도 가능하다. 예를 들어, 메모리 노드에서의 데이터 패턴을 파악하기 위해서 같은 위치에 패킷을 주기적으로 보내고, 해당 패킷의 지연시간에 따라서 데이터의 접근 패턴을 파악할 수 있다 [17].

Ⅳ. 분리 메모리 시스템의 보안 기술 연구 동향

이 장에서는 최근 발표된 분리 메모리 시스템의 보안 기술에 대해 소개하고자 한다. 본 논문에서는 그림 4와 같이 분리 메모리 시스템의 보안 기술 연구를 세 가지 주제로 분류하였다. 해당 분류의 연구들에 대해서 자세히 알아보고자 한다.



〔그림 4〕 분리 메모리 시스템의 보안 기술 연구

4.1. 신뢰할 수 있는 분리 메모리 시스템

첫 번째 분류로 신뢰할 수 있는 분리 메모리 시스템에 관한 연구를 소개한다. TDMEM [18]은 분리 메모리 시스템 환경에서 컴퓨팅 노드의 커널을 포함하여 모든 노드에 배치한 신뢰할 수 있는 FPGA를 통해서 보안 문제를 해결하고자 하였다. 컴퓨팅 노드의 커널 수준에서 하드웨어 가속을 통한 암호화 및 무결성 검증을 수행하고, FPGA를 통해서 메모리 노드에 대한 접근 권한을 검증하는 방식을 사용하여 데이터를 보호하였다. 마지막으로, 메모리 접근 패턴을 난독화하기위해서 커널의 스왑 인터페이스를 사용하고, 더불어 FPGA에서 매번 새로운 위치로 할당하는 방식을 사용하였다.

Minerva [19]는 보안 프로세서 기술을 활용하여 호스트 메모리에 저장된 데이터 이외에 CXL, Gen-Z, CCIX와 같은 고속의 인터페이스로 연결된 Fabric attached Memory (FAM)에 저장된 데이터까지 보호하는 기술을 제안하였다. 해당 인터페이스는 캐시 일관성을 제공하기 때문에 일관성 유지에 필요한 메시지들로 인하여 발생하는 확장성 문제를 보완하였다. 또한, FAM으로 연결되는 메모리는 비휘발성 메모리를 포함하기 때문에 크래시 일관성을 제공하였다.

D-Shield [20]는 Minerva와 마찬가지로 보안 프로세서 기술을 활용하여 메모리 이외에 NVMe SSD의 데이터 보호를 제공하는 기술을 제안하였다. 디스크에 저장된 큰 용량의 데이터의 무결성 검증을 위해서 메모리를 무결성 검증 트리의 캐시로 사용하는 등의 최적화 기법을 사용하였다. 분리 메모리 시스템에서는 NVMe SSD도 메모리 확장용 디바이스로 사용가능하기 때문에 해당 연구도 분리 메모리 시스템의 보안 기술로 포함될 수 있다.

4.2. 안전한 RDMA 프로토콜을 통한 통신 보호

원격 메모리 노드를 사용하는 분리 메모리 시스템은 인피니밴드와 같은 고속 인터페이스를 활용하기 위해서 RDMA 프로토콜을 사용한다. 최근 RDMA 프로토콜에 보안 기능 추가를 제안하는 연구들이 진

행되고 있어 본 논문에서 소개하고자 한다.

ReDMArk [17]는 RDMA 프로토콜을 통한 통신 채널에서 발생할 수 있는 다양한 취약점에 대한 고찰을 수행하고, 해당 취약점을 완화시킬 수 있는 기술을 소개하였다. 특히, 분리 메모리 시스템 중 하나인 Infiniswap [2]의 RDMA 통신 채널에서 일어날 수 있는 취약점들에 대해서 다음과 같이 설명하고 있다: 1) RDMA를 통해서 패킷을 누락 혹은 주입, 2) DDoS공격, 3) 메모리 노드에서 새로 할당되는 메모리 위치나 rkey를 예측, 4) 악의적 컴퓨팅 노드가 다른 컴퓨팅 노드의 원격 데이터 영역에 접근, 5) 하나의 컴퓨팅 노드가 모든 리소스 자원을 사용하여 성능 일관성 침해와 같은 위협이 존재한다. 따라서, 분리 메모리 시스템의 보안을 위해서는 안전한 통신 채널을 구성하는 것이 필요하다.

sRDMA [21]는 RDMA 프로토콜에 보안 기능을 강화하기 위해서 신뢰할 수 있는 스마트 RNIC (smart RDMA Network Interface Card)을 활용하였다. 스마트 RNIC을 통해서 통신 패킷에 보안 기능을 제공하기 위한 헤더 정보를 추가하고, 패킷 암호화를 제공하였다.

Bedrock [22]은 sRDMA에서 제공하는 보안 기능들을 RNIC이 아닌 프로그램 가능한 스위치에서 제공하는 방법을 사용하였다. 이를 통해서 네트워크 트래픽에 대한 모니터링과 로깅 기능을 제공하며, RNIC의 사용 대비 성능을 향상시켰다.

4.3. 이종 TEE를 위한 단일화된 신뢰 환경 제공 기술

분리 메모리 시스템은 다양한 하드웨어를 통해서 구성될 수 있다. 따라서, 다양한 종류의 TEE가 존재할 수 있다. 예를 들어서, 단일 랙 내에서도 Intel CPU를 사용하는 서버가 있을 수 있고, AMD CPU를 사용하는 서버가 있을 수 있다. 이러한 환경에서는 다양한 종류의 하드웨어 기반 TEE가 공존하게 되고, 서로의 신뢰 환경에 대한 컨센서스를 맞추어야 한다. 따라서, 다수의 연구가 분리 메모리 시스템 혹은 분산 시스템 환경에서의 해당 문제를 파악하고 문제 해결 방법을 제안하였다.

HETEE [23]는 랙-스케일의 컴퓨팅 환경에서 새로운 형태의 TEE를 만들어 단일 랙 내의 모든 컴퓨팅 자원들을 동적으로 관리하고, 각 컴퓨팅 작업들을 격리 시키는 기술을 제안하였다. 신뢰할 수 있는 환경을 제공하기 위해서 랙 내부에 중앙화된 HETEE 시스템을 구축하고 Security Controller (SC)를 통해서 보안에 필요한 암호화 및 인증 기능을 제공하였다.

Dhar 등 [24]은 HETEE와 유사하게 SC를 활용하되, 실제 TEE를 제공하는 CPU의 기능을 이용해서 TEE를 제공하지 않는 노드 혹은 디바이스에 대한 보호 기술을 제안하였고, Koshiba 등 [25]은 FPGA를 통해 단일화된 하드웨어 모듈과 마이크로 커널을 활용하여 virtual TEE (vTEE)를 제공해주고, 다수의 노드가 vTEE 별로 별도의 접근 권한을 가지고 노드 혹은 통신 채널에 대해서 데이터를 보호하는 기술을 제안하였다.

Ⅴ. 향후 연구 주제

앞에서 분리 메모리 시스템에 보안을 제공하기 위한 다양한 기술들에 대해 살펴보았다. 이 장에서는 분리 메모리 시스템을 위한 보안 기술과 관련한 향후 연구 주제들을 정리해보고자 한다.

첫째, 기존 연구들은 특정 디바이스 혹은 인터커넥트로의 메모리 확장을 위한 보안 기술만을 제시하였다. RDMA 기반 메모리 확장을 보호해주는 기술이 많았고 [17, 18, 19, 21, 22], 일부 NVMe SSD의 데이터를 보호해주는 기술이 있었다 [20]. 하지만, 앞으로 분리 메모리에서는 속도와 용량이 다양한 디바이스를 동시에 활용하여 시스템의 효율을 높이는 기술이 발전할 것으로 보인다. Infiniswap도 빠르게 접근할 수 있는 RDMA로 연결된 메모리와 속도는 느리지만 가용성이 높은 SSD를 동시에 활용한 바 있으며 [2], DCM도 RDMA로 연결된 메모리와 고성능 SSD 등을 동시에 지원한다 [4]. 따라서 각 디바이스 종류별 보안이 아닌 분리 메모리 시스템을 위해 다양한 디바이스를 지원할 수 있는 보안 기법에 대한 연구가 필요하다.

둘째, 기존 연구들은 분리 메모리 시스템에 보안을 제공하기 위해 특별한 하드웨어를 요구한다. 프로세서를 일반적인 프로세서가 아닌 보안 프로세서를 사용해야 하거나 [19, 20], 전용 FPGA를 장착해야 하거나 [18, 25], 프로그램 가능한 스위치 혹은 네트워크 카드를 요구하기도 했다 [21, 22]. 하지만, 분리 메모리 시스템은 위와 같은 시스템이 없이도 구현이 가능하기 때문에, 기존 시스템에 분리 메모리 기능을 적용

할 경우 보안을 제공하기 어렵다는 단점이 있다. 따라서, 소프트웨어 기반의 분리 메모리 시스템 보안 기술 연구가 필요하다. TDMEM 논문에서 보고한 바에 따르면, 기존 소프트웨어로 CPU를 사용해 4KB 데이터를 암/복호화하는데 2μs 정도가 걸린다고 한다 [18]. 또한, 분리 메모리 시스템에서는 고정된 크기의 데이터를 입출력하는 경우가 대부분이므로, 암/복호화 속도를 더 최적화할 여지도 있을 것이다. 따라서, 소프트웨어만을 사용한 분리 메모리 보안 시스템 구현이 가능할 것으로 보인다.

Ⅵ. 결 론

분리 메모리 시스템은 대용량의 메모리 확장 기능을 바탕으로 LLM과 같은 대규모 인공지능 모델 처리, 대규모 데이터 분석을 효과적으로 제공하며, 데이터센터 내의 메모리 활용률 향상을 위해서 필요한 기술이다. 하지만, 새로운 환경에 따른 보안 취약점을 해결하지 못한다면 널리 사용되기 힘들 것이다.

본 논문에서는 분리 메모리 시스템에서의 보안 위협에 대한 분석과 최근 연구되고 있는 분리 메모리 시스템 보안 기술 연구 동향에 대해 살펴보았다. 또한, 분리 메모리 시스템에 쉽게 적용될 수 있는 향후 연구 주제에 대해서 소개하였다. 해당 보안 기술을 바탕으로 분리 메모리 시스템이 좀 더 많은 컴퓨팅 환경에 적용되기를 기대한다.

참 고 문 헌

[1] Jing Guo, Zihao Chang, Sa Wang, Haiyang Ding, Yihui Feng, Liang Mao, and Yugang Bao. “Who limits the Resource Efficiency of My Datacenter: An Analysis of Alibaba Datacenter Traces,” *In IEEE/ACM 27th International Symposium on Quality of Service (IWQoS)*, pages 1-10, 2019.

[2] Juncheng Gu, Youngmoon Lee, Yiwen Zhang, Mosharaf Chowdhury, and Kang G. Shin. “Efficient Memory Disaggregation with Infiniswap,” *In Proceedings of the 14th USENIX Symposium on Networked Systems Design and Implementation (NSDI)*, pages 649-667, March 2017.

[3] Emmanuel Amaro, Christopher Branner-Augmon, Zhihong Luo, Amy Ousterhout, Marcos K. Aguilera, Aurojit Panda, Sylvia Ratnasamy, and Scott Shenker. “Can Far Memory Improve Job Throughput?,” *In Proceedings of the 15th European Conference on Computer Systems (EuroSys)*, pages 1-16, April 2020.

[4] Kwangwon Koh, Kangho Kim, Seunghyub Jeon, and Jaehyuk Huh. “Disaggregated Cloud Memory with Elastic Block Management,” *In IEEE Transactions on Computers*, 68(1):39-52, 2019.

[5] Taehoon Kim, Kwangwon Koh, Changdae Kim, Eunji Pak, Yeonjeong Jeong, and Sang-Hoon Kim. “DEHype: Retrofitting Hypervisors for a Resource-Disaggregated Environment,” *In IEEE International Conference on Cluster Computing (CLUSTER)*, Oct 2023.

[6] Miryeong Kwon, Junhyeok Jang, Hanjin Choi, Sangwon Lee, and Myoungsoo Jung. “Failure Tolerant Training With Persistent Memory Disaggregation Over CXL,” IEEE Micro, 43(2):66-75, 2023.

[7] Evangelos P. Markatos and George Dramitinos. “Implementation of a Reliable Remote Memory Pager,” *In Proceedings of USENIX Annual Technical Conference (USENIX ATC)*, January 1996.

[8] Donghyun Gouk, Sangwon Lee, Miryeong Kwon, and Myoungsoo Jung. "Direct Access, High-Performance Memory Disaggregation with DirectCXL," *In Proceedings of USENIX Annual Technical Conference (USENIX ATC)*, July 2022.

[9] Hasan Al Maruf and Mosharaf Chowdhury. “Effectively Prefetching Remote Memory with Leap,” *In Proceedings of USENIX Annual Technical Conference (USENIX ATC)*, July 2020.

[10] Zhenyuan Ruan, Malte Schwarzkopf, Marcos K. Aguilera, and Adam Belay. "AIFM:

High-Performance, Application-Integrated Far Memory," *In Proceedings of USENIX Conference on Operating Systems Design and Implementation (OSDI)*, Nov 2020.

[11] Wonsup Yoon, Jisu Ok, Jinyoung Oh, Sue Moon, and Youngjin Kwon. “DiLOS: Do Not Trade Compatibility for Performance in Memory Disaggregation,” *In Proceedings of European Conference on Computer Systems (EuroSys)*, May 2023.

[12] G. Edward Suh, Dwaine Clarke, Blaise Gassend, Marten van Dijk, and Srinivas Devadas. “AEGIS: Architecture for Tamper-Evident and Tamper-Resistant Processing,” *In Proceedings of the 17th International Conference on Supercomputing (ISC)*, June 2003.

[13] David Lie, Chandramohan Thekkath, Mark Mitchell, Patrick Lincoln, Dan Boneh, John Mitchell, and Mark Horowitz. “Architectural Support for Copy and Tamper Resistant Software,” *In Proceedings of the 9th International Conference on Architectural Support for Programming Languages and Operating Systems (ASPLOS)*, pages 168–177, November 2000.

[14] Junghoon Lee, Taehoon Kim, and Jaehyuk Huh, "Reducing the Memory Bandwidth Overheads of Hardware Security Support for Multi-Core Processors," In IEEE Transactions on Computers, 65(11):3384-3397, 2016.

[15] G. Edward Suh, Dwaine Clarke, Blaise Gassend, Marten van Dijk, and Srinivas Devadas. “Efficient Memory Integrity Verification and Encryption for Secure Processors,” *In Proceedings of the 36th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pages 339–350, 2003.

[16] Brian Rogers, Siddhartha Chhabra, Milos Prvulovic, and Yan Solihin. “Using Address Independent Seed Encryption and Bonsai Merkle Trees to Make Secure Processors OS- and Performance-Friendly,” *In Proceedings of the 40th Annual IEEE/ACM International Symposium on Microarchitecture (MICRO)*, pages 183–196, 2007.

[17] Benjamin Rothenberger, Konstantin Taranov, Adrian Perrig, and Torsten Hoefler. “ReDMArk: Bypassing RDMA Security Mechanisms,” *In Proceedings of 30th USENIX Security Symposium (USENIX Security)*, pages 4277-4292, 2021.

[18] Taekyung Heo, Seunghyo Kang, Sanghyeon Lee, Soojin Hwang, and Jaehyuk Huh. “Hardware-assisted Trusted Memory Disaggregation for Secure Far Memory,” arXiv, 2021

[19] Mazen Alwadi, Rujia Wang, David Mohaisen, Clayton Hughes, Simon David Hammond, and Amro Awad. “Minerva: Rethinking Secure Architectures for the Era of Fabric-Attached Memory Architectures,” *In IEEE International Parallel and Distributed Processing Symposium (IPDPS)*, pages 258-268, 2022

[20] Md Hafizul Islam Chowdhuryy, Myoungsoo Jung, Fan Yao, and Amro Awad, "D-Shield: Enabling Processor-side Encryption and Integrity Verification for Secure NVMe Drives," *In IEEE International Symposium on High-Performance Computer Architecture (HPCA)*, pages 908-921, 2023.

[21] Konstantin Taranov, Benjamin Rothenberger, Adrian Perrig, and Torsten Hoefler. “sRDMA: Efficient NIC-based Authentication and Encryption for Remote Direct Memory Access,” *In Proceedings of USENIX Annual Technical Conference (USENIX ATC)*, pages 691-704, 2020.

[22] Jiarong Xing, Kuo-Feng Hsu, Yiming Qiu, Ziyang Yang, Hongyi Liu, and Ang Chen. “Bedrock: Programmable Network Support for Secure RDMA Systems,” *In Proceedings of 31st USENIX Security Symposium (USENIX Security)*, pages 2585-2600, 2022.

[23] Jianping Zhu, Rui Hou, XiaoFeng Wang, Wenha Wang, Jiangfeng Cao, Boyan Zhao, Zhongpu

Wang, Yuhui Zhang, Jiameng Ying, Lixin Zhang, and Dan Meng. “Enabling Rack-scale Confidential Computing using Heterogeneous Trusted Execution Environment,” *In 2020 IEEE Symposium on Security and Privacy (SP)*, pages 1450-1465, 2020.

[24] Aritra Dhar, Supraja Sridhara, Shweta Shinde, Srdjan Capkun, and Renzo Andri. “Empowering Data Centers for Next Generation Trusted Computing,” arXiv, 2022.

[25] Atsushi Koshiba, Felix Gust, Julian Pritzi, Anjo Vahldiek-Oberwagner, Nuno Santos, and Promod Bhatotia. “Trusted Heterogeneous Disaggregated Architectures,” *In Proceedings of the 14th ACM SIGOPS Asia-Pacific Workshop on Systems (APSys)*, pages 72-79, 2023.

〈저자소개〉

용 예 원 (Yewon Yong)

2017년 2월 : 한남대학교 정보통신공학과 학사
2022년 3월~현재 : 충남대학교 컴퓨터공학과 석사과정 (한국전자통신연구원 학연협동과정)
<관심분야> 분리 메모리, 시스템 보안

김 창 대 (Changdae Kim)

2010년 1월 : 한국과학기술원 전산학과 학사
2012년 2월 : 한국과학기술원 전산학부 석사
2017년 8월 : 한국과학기술원 전산학부 박사
2017년 9월~2018년 10월 : 한국과학기술원 박사후연구원
2018년 11월~현재 : 한국전자통신연구원 선임연구원
<관심분야> 컴퓨터 구조, 운영체제, 메모리 시스템

김 태 훈 (Taehoon Kim)

2012년 2월 : 서강대학교 컴퓨터공학과 학사
2014년 2월 : 한국과학기술원 전산학부 석사
2020년 2월 : 한국과학기술원 전산학부 박사
2020년 3월~2022년 2월 : 한국전자통신연구원 연구원
2022년 3월~현재 : 한국전자통신연구원 선임연구원
<관심분야> 컴퓨터 구조, 메모리 시스템, 시스템 보안