클라우드 컴퓨팅과 안전성을 가진 다자간 연산*

은하수 [1†], 이훈정 [1], 손정갑 [1], 오희국 [1], 김상진 [2‡]
[1] 한양대학교 컴퓨터공학과
[2] 한국기술교육대학교 컴퓨터공학과
e-mail: hseun@hanyang.ac.kr

Cloud Computing and Secure Multi-Party Computation*

Hasoo Eun[1†], Hoonjung Lee[1], Junggab Son[1], Heekuck Oh[1], Sangjin Kim[2‡]
[1]Dept. of Computer Science, Hanyang University
[2]Dept. of Computer Science, Korea University of Technology and Education

요 약

클라우드 컴퓨팅 인프라를 사용할 때 사용자의 민감한 정보가 포함된 데이터를 사용하게 될 수 있다. 데이터를 아웃소싱하여 처리하는 경우 클라우드 제공자가 데이터 처리자로서 사용자의 데이터에 접근해야 한다. 사용자는 데이터를 처리하는 과정에서 행하는 클라우드 제공자의 동작을 알 수 없으므로 클라우드 컴퓨팅을 사용하는 것을 불안해하게 되고 공개를 해도 되는 일부의 데이터만을 사용하게 된다. 본 연구에서는 클라우드 컴퓨팅을 통해 연산을 수행하는 환경에서 사용자의 데이터를 보호하기 위한 연구의 일환으로써, 시스템 및 환경을 정의하고 주로 발생할 수 있는 정보보호 위협을 정리하였다. 또한 현재 연구가 진행되고 있는 SMPC(Secure Multi-Party Computation)을 소개하고 이를 클라우드 컴퓨팅을 통해 연산을 수행하는 환경에 적용하기 위해 고려해야 할 사항들을 제시하며, 향후 연구 방향을 모색한다.

1. 서론

클라우드 컴퓨팅이란 인터넷 기술을 활용하여 '가상화된 IT 자원을 서비스'로 제공하는 컴퓨팅으로써, 사용자는 IT 자원(소프트웨어, 스토리지, 서버, 네트워크)을 필요한 만큼 빌려서 사용하고, 서비스 부하에 따라서 실시간 확장성을 지원받으며, 사용한 만큼 비용을 지불하는 시스템이다[1]. 클라우드 사용자는 클라우드 컴퓨팅 인프라를 사용하여 자신이 원하는 연산을 수행한다. 이때 사용자의 데이터가 입력 값으로 사용된다. 기업의 경우 Private Cloud 를 사용하여 직접 이를 관리하므로 입력 데이터에 대한 정보유출 걱정이 조금 덜 한 편이지만, 이러한 환경을 구축할 수 없는 일반 사용자의 경우 자신이 입력한 데이터가 유출될 것을 두려워하여 정작 우수한 컴퓨팅 자원이 있음에도 불구하고 사용을 꺼려하게 된다.

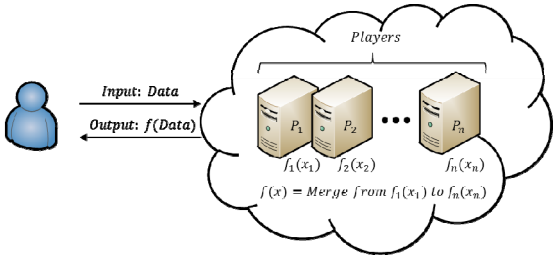
본 연구에서는 이러한 클라우드 컴퓨팅 환경의 제한 사항을 극복하고자 클라우드 서비스 제공자로부터 데이터를 보호할 수 있는 기법을 연구하고자 한다. 이를 위한 초석으로 본 논문에서는 위와 같은 문제가 발생할 수 있는 클라우드 컴퓨팅 시스템 및 환경을 정의하고 정보보호 위협을 정리한다. 또한 현재 연구가 진행되고 있는 SMPC 를 소개하고 이를 클라우드 컴퓨팅을 통해 연산을 수행하는 환경에 적용하기 위해 고려해야 할 사항들을 제시한다.

이후 본 논문의 구성은 다음과 같다. 2 장에서 클라우드 컴퓨팅을 통한 데이터 연산 환경을 정의하고 그러한 환경에서 발생할 수 있는 보안 위협을 정리한다. 3 장에서는 SMPC 를 소개하고 4 장에서는 이를 클라우드 컴퓨팅에 적용할 수 있는 방안을 제시한다. 5 장에서는 결론을 맺고 향후 연구 방향을 모색한다.

2. 환경정의

2.1. 클라우드 컴퓨팅을 통한 데이터 연산

클라우드 컴퓨팅에서의 아웃소싱은 클라우드 컴퓨팅에서 발생하는 연산을 외부 업체에 맡기는 것을 말한다. 사용자는 업체로부터 컴퓨팅자원을 제공받으며, 그 업체는 사용자의 데이터를 받아 연산을 처리한다.



(그림 1) 클라우드 컴퓨팅을 통한 데이터 연산

* 본 연구는 지식경제부 및 정보통신산업진흥원의 대학 IT 연구센터 지원사업의 연구결과로 수행되었음(NIPA-2011-C1090-1111-0010).
* 이 논문은 2011 년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임(No.2011-0000189).
‡ 교신저자, sangjin@kut.ac.kr

클라우드 컴퓨팅을 통해 연산을 수행하게 되면 Thin Client 환경에서도 빠르게 연산을 수행할 수 있다. 가까운 예로써 클라우드 컴퓨팅을 통해 동영상 인코딩 서비스를 제공해주는 국내 클라우드 서비스를 들 수 있다. 이 서비스는 사용자의 PC 를 이용하여 업로드 한 700mb 가량의 동영상을 2 분만에 인코딩하여 이동 단말을 통해 감상할 수 있게 해준다. 기존 PC 의 연산 능력에 비하면 10 배가량 빠른 속도로 연산이 가능하다[2]. 하지만 이러한 편의를 받기 위해서는 사용자가 클라우드 제공자에게 자신의 콘텐츠를 전송해야 하며, 콘텐츠가 노출될 것을 고려한다면 제한적으로 서비스를 이용할 수 밖에 없다.

2.2. 보안 위협

캐나다 온타리오의 IPC(정보 및 프라이버시 위원회 사무국)는 'Privacy by Design(PbD)' 이라는 개념에서 클라우드 컴퓨팅 개인정보보호 위협을 <표 1>과 같이 분류하였다[3].

3. 안전성을 가진 다자간 연산, SMPC

최근 클라우드 컴퓨팅에서 데이터를 보호하기 위한 연구가 진행되고 있다. SMPC 자체적인 문제는 기존과 같이 제 3 의 신뢰기관 두거나 서비스 제공자를 전적으로 신뢰하는 방법으로 간단히 해결될 수 있으나, 클라우드 컴퓨팅 환경에서는 서비스 제공자가 사용자의 완전한 데이터를 얻을 수 있으므로 서비스 제공자의 악의적인 동작을 배제할 수 없다. SMPC 문제는 제 3 의 신뢰기관을 가정하지 않고 프라이버시를 보호하는 다자간 협력 계산방식을 개발하기 위해 정의되었다. 일반적으로 SMPC 란 여러 개체가 각자의 프라이버시 노출 없이 서로 협력하여 어떤 결과를 수행하는 것을 말한다[4]. 이러한 SMPC 는 1982 년 A. Yao 에 의해 소개되었다[5]. 서로의 연봉은 공개하지 않고 누가 가장 부자인지 계산하는 Millionaire-Problem 은 아직까지도 SMPC 를 가장 잘 표현한 예로 알려져 있다[6].

<표 1> PbD 클라우드 컴퓨팅 개인정보보호 위협

RA1. 사법권(Jurisdiction)
국가별 데이터 보호와 관련한 법과 접근법이 상이하므로 다중의 사법권에 포함될 수 있으며 이에 대한 중재가 요구됨
RA2. 새로운 데이터 흐름의 생성 (Creation of New Data-streams)
클라우드 모델은 방대한 규모의 새로운 데이터를 생성하고, 이는 정보 중재자 및 제공자에게 노출될 수 있으며, 본래의 목적을 넘어서 사용될 수 있음
RA3. 보안(Security)
클라우드 서비스 제공자는 데이터와 흐름을 보호하기 위하여 현재 온라인 뱅킹이나 소매업에서 사용되는 암호화 기법을 사용해야 하지만 대부분이 그렇지 못함
RA4. 데이터 침해(Data Intrusion)
클라우드 서비스 제공자, 정부 및 관련 기관에 의한 사용자 데이터의 접속 및 활용 등이 가능할 수 있으며, 사용자는 이러한 침해에 대해 인지하지 못하는 경우가 발생할 수 있음
RA5. 합법적인 접근(Lawful Access)
합법적인 접근 자체로는 문제가 없지만 해당 목적을 넘어선 접근이 이루어 질 수 있으며, 이에 대해 인지하지 못할 수 있음
RA6. 처리(Processing)
처리를 아웃소싱하는 경우, 사용자는 데이터에 대한 통제권자로써 접속 • 수정 • 삭제 절차가 적절하고 적합함을 보장받아야 함
RA7. 처리 데이터의 오용(Misuse of Processing Data)
클라우드 제공자가 처리자로써 처리하는 활동과 처리 이외의 활동으로 구분하여 접속을 제한 • 관리하여야 함
RA8. 데이터 영속성(Permanence of Data)
계약이 완료된 후에 데이터는 클라우드 인프라에서 영구적으로 제거되고 언제 제거가 완료되는지 확인하여야 함
RA9. 데이터 소유권(Ownership of Data)
새로운 데이터 흐름을 통해 생성된 데이터의 소유권이 불확실해질 수 있으며, 해당 데이터의 생산 및 존속에 대해 고려하여야 함

※ R = Risk, A = Article Number

4. 클라우드와 SMPC

4.1. 공격자 정의

클라우드 컴퓨팅에 SMPC 를 적용하는 경우 공격자는 연산을 수행하는 Player 들이라 할 수 있다. 이들은 사용자의 데이터 중 일부를 받아서 연산하므로 전체 데이터를 얻을 수는 없다. 따라서 공격을 통해 다른 사용자들의 데이터를 얻거나 연산을 방해하려 할 것이다. 최근 제안된 논문에서 정의하고 있는 공격자는 다음과 같이 넷으로 구분된다 [7].

- **수동적 공격자(Passive Adversary):** 프로토콜을 따르되, 그 이상의 정보를 얻으려고 하는 자
- **능동적 공격자(Active Adversary):** 프로토콜의 정의를 벗어나거나, 정직한 Player 의 연산을 방해 혹은 오답을 유도하려는 자
- **전환적 공격자(Convert Adversary):** 프로토콜의 정의를 벗어났으면서도 이를 알지 못하게 은폐하려는 자
- **집합적 공격자(Monolithic Adversary):** 여러 Player 를 포획하여 정보를 얻으려 하는 자

한번 공격이 발생하게 되면 서버에서는 공격자가 누구인지 알 수 있고, 해당 Player 를 배제하게 된다. 따라서 Corrupt Player 를 구별해내는 시점이 중요하며 다음과 같이 둘로 구분할 수 있다.

- **정적 공격자(Static Adversary):** Corrupt Player 를 프로토콜 수행 전에 미리 알 수 있음
- **동적 공격자(Adaptive Adversary):** 프로토콜을 수행하는 도중에 Corrupt Player 를 알 수 있음

초기의 SMPC 에서는 두 사용자 사이에서 연산을 수행하였으므로, 공격자를 가려내거나 배제하는데 있어서 현실적이었으나 현재의 환경에 적용하기에는 비효율적이다. 2005 년 이후부터 n 명의 사용자 사이에서 SMPC 를 사용하기 위한 연구가 진행되고 있으며, 현재는 정적 공격자의 수를 t 라 했을 때 $t<n/2$ 인 경우 안전성을 보장할 수 있다는 분석이 발표되었다[7].

4.2. 기존 연구 분류

4.2.1. Secret Sharing 을 이용한 기법[7]

하나 이상의 서비스 제공들에게 사용자의 데이터를

보내어 Secret Sharing 을 통해 연산을 수행하는 기법이다. 전체의 데이터를 모두 모아야 완벽한 비밀값을 얻을 수 있으므로, 데이터 위/변조에 취약하다. 이를 보호하기 위해 위/변조 된 값을 찾고 복원하기 위한 알고리즘이 반드시 필요하다.

4.2.2. Homomorphic Encryption 을 이용한 기법[8]

단일 서버에 대하여 최근 발견된 Homomorphic Encryption 을 적용한 기법이다. 하지만 아직까지 이론적인 연구의 결과이며 가까운 장래에 적용되기는 힘들 것으로 보인다. 게다가 단일 서버를 사용하는 경우 능동적 공격자에 의해 시스템 자체가 쉽게 공격당할 수 있다.

4.2.3. 단일 서버와 TRH 를 사용한 기법[9]

단일 서버에서 TRH 를 이용하여 서버 내에서 비밀연산을 수행하는 기법이다. 데이터는 암호화되어 서버로 전송되며 TRH 내에서 연산이 수행된다. 이를 위해 TRH 자체적으로 연산 능력이 무척 좋아야 한다.

4.3. 클라우드 컴퓨팅을 통한 연산 수행에 SMPC 적용을 위한 고려사항

4.3.1. 데이터 송신 시 고려사항

사용자의 데이터가 클라우드로 송신하는 과정에서 외부 공격자의 개입이 발생할 수 있다. 따라서 데이터를 암호화해서 전송할 필요가 있다[RA3]. 이를 위해 사용자는 서비스 제공자와 보호된 채널을 통해 데이터를 주고 받아야 한다.

데이터 전송을 위한 창구 역할을 하는 서비스 제공자가 데이터 전체를 습득하는 상황을 고려해야 한다. 이를 위한 보호대책으로 스트림 데이터를 보호하기 위해 사용되고 있는 스크램블링, DRM 등을 들 수 있다.

4.3.2. 연산 시 고려사항

사용자의 데이터가 연산되는 과정에서 공격자가 사용자의 데이터를 위/변조 할 때 이를 구별해 낼 수 있어야 한다. 이를 위해 해당 연산에 관여한 Player 의 정보와 그가 이 연산을 수행했다는 증거가 필요하다[RA6]. 이는 간단하게 서명 등을 통해 자신이 연산했음을 보일 수 있다.

전송된 데이터에 대하여 정당한 Player 만이 사용할 수 있도록 접근제어를 제공해 주어야 한다[RA7]. 이를 위해 데이터의 암호화에 사용자의 키와 Player 의 키가 함께 사용되도록 하여 보호할 수 있는 암호기법이 필요하다. 일반적으로 신뢰기관을 두는 경우 이들의 키도 함께 사용될 수 있으나, 그리하면 사용자가 인지 못하는 상황에서 신뢰기관에 의해 데이터 유출이 발생할 수 있다[RA4].

4.3.3. 결과 값 반환 시 고려사항

사용자의 데이터는 클라우드 서비스 제공자에 의해 Player 로 분산되고, 모이게 된다. 따라서 각각 안전하게 연산을 했다 하더라도 서비스 제공자의 악의적인 행동에 의해 사용자의 데이터 유출이 발생할 수 있다[RA5]. 따라서 클라우드 서비스 제공자를 사이에 두고도 안전하게 데이터를 주고받을 수 있는 기법이 필요하다. 이는 앞서 이야기한 Secret Sharing 을 하더라도 데이터를 결합하며 복호화하는 과정에서 사용자의 데이터가 클라우드 서비스 제공자에게 노출된다. 이를 사용자 단말에서 수행함으로써 간단히 해결할 수 있지만 연산에 따른 부담이 사용자 단말로 전가되는 한계가 있다.

5. 결론 및 향후 연구 방향

본 논문에서는 클라우드 컴퓨팅을 이용해 연산을 수행하는 환경에서 사용자의 데이터를 보호하기 위한 연구의 일환으로써, 시스템 및 환경을 정의하고 정보보호 위협을 정리하였다. 또한 현재 연구가 진행되고 있는 SMPC(Secure Multi-Party Computation)을 소개하고 이를 클라우드에 적용하기 위해 고려해야 할 사항들을 제시하였다. 현재까지 제안된 클라우드 컴퓨팅관련 SMPC 논문들은 대부분 클라우드 환경에 대하여 강력한 가정(Consensus Broadcasting 등)을 두고 SMPC 프로토콜 자체에 치중하는 경향이 있다. 따라서 향후에는 본 논문의 내용을 기반으로 기존 연구와 더불어 가정을 줄여도 안전한 클라우드 컴퓨팅 연산 시스템을 구성하려 한다.

참고문헌

[1] 김학범, 전은정, 김성준, “클라우드 컴퓨팅 환경에서의 보안관리에 관한 연구,” 경영컨설팅리뷰, 제 2 권, 제 1 호, pp. 127-144, 2011 년 2 월.

[2] KBench, “클라우드컴퓨팅 이용, 동양상 인코딩에 단 2 분,” http://www.kbench.com/hardware/?no=94103

[3] 박대하, 백태석, “클라우드 컴퓨팅 개인정보보호 연구동향과 과제,” 한국정보보호학회논문지, 제 21 권, 제 5 호, pp. 37-44, 2011 년 8 월.

[4] 강주성, 이옥연, 홍도원, “선형계를 위한 실용적인 프라이버시 보존형 다자간 계산 프로토콜,” 한국정보보호학회논문지, 제 16 권 제 2 호, pp. 13-24, 2006 년 4 월.

[5] A. Yao, “Protocols for Secure Computations,” *Proceedings of the 23th Annual IEEE Symposium on Foundations of Computer Science*, 1982.

[6] Secure Multi-Party Computation, http://en.wikipedia.org/wiki/Secure_multi-party_computation

[7] J. Loftus and N. Smart, “Secure Outsourced Computation,” *AFRICACRYPT 2011, LNCS 6737*, pp. 1-20, 2011.

[8] M. Dijk, C. Gentry, S. Halevi, V. Vaikuntanathan, “Fully Homomorphic Encryption Over the Integers,” *EUROCRYPT 2010, LNCS 6110*, pp. 24-43, 2010.

[9] A. Sadeghi, T. Schneider, and M. Winandy, “Token-based Cloud Computing: Secure Outsourcing of Data and Arbitrary Computations with Lower Latency,” *TRUST 2010, LNCS 6101*, pp. 417-429, 2010.