클라우드 컴퓨팅 환경의 데이터 신뢰 확보

정회원 정 임 영*, 조 인 순**, 준회원 유 영 진***

Trust Assurance of Data in Cloud Computing Environment

Im Y. Jung*, Insoon Jo** *Regular Members*, Youngjin Yu*** *Associate Member*

요 약

가상화를 통해 원하는 만큼의 컴퓨팅 파워와 데이터 저장 공간을 제공하면서도 관련된 IT자원의 유지보수 비용과 관리에서는 사용자를 해방시켜주는 많은 장점에도 불구하고, 클라우드 컴퓨팅이 가까운 미래에 실질적인 서비스로 자리잡고 활성화를 위해서는 먼저 넘어야 할 장벽들이 있다. 즉, 사용자의 제어 너머에 있는 클라우드 컴퓨팅 환경이 IT서비스와 인프라에 대해 사용자에게는 이용권만을 주기 때문에 비롯되는 여러 문제들이 생기게 된다. 가장 큰 이슈 중의 하나는 클라우드에 저장되는 정보의 보호 및 신뢰성 확보이다. 본 논문에서는 provenance 통한 클라우드 상의 데이터 신뢰확보에 대한 효과적이고 유용한 해법을 제안한다.

Key Words : Cloud Computing, Trust of Data, Provenance, Data Integrity, Audit Integrity

ABSTRACT

Cloud Computing Environment provides users with a blue print of IT Utopia with virtualization; unbounded computing power and data storage free from the cost and the responsibility of maintenance for the IT resources. But, there are several issues to be addressed for the Cloud Computing Environment to be realized as the blue print because users cannot control the IT resources provided by the Cloud Computing Environment but can only use them. One of the issues is how to secure and to trust data in the Cloud Computing Environment. In this paper, an efficient and practical trust assurance of data with provenance in Cloud Computing Environment.

Ⅰ. 서 론

가상화를 통해 사용자가 필요로 하는 만큼의 맞춤형 IT 자원을 제공할 수 있는 클라우드 컴퓨팅(Cloud Computing)은 미래 IT의 청사진을 보여주는 패러다임으로 학계와 산업계의 주목을 받고있다[5]. 클라우드(Cloud)란 사용자가 필요로 하는 모든 IT 서비스, 즉, 응용프로그램인 상위 IT 자원부터 IT서비스를 개발할 수 있는 하위 IT자원에 이르는 모든 것이 서비스로 제공되는 가상화된 공간을 의미한다.

클라우드 컴퓨팅이 우리의 일상으로 들어오면 개인의 랩탑 혹은 여러 데이터 저장소에 분산되어있는 이메일, 각종 문서들, 큰 저장용량을 필요로하는 데이터들이 클라우드에 저장될 수 있다[6]. 사용자는 저장소의 용량을 개인이 항상 확인하면서 필요시 더 큰 저장용량의 저장소로 옮기거나 주기적으로 데이터들을 삭제하지 않아도 된다. 또한, 클라우드 컴퓨팅 환경은 개개인이 필요한 IT 자원을 구입하고 유지보수하는 비용을 절약해주어 경제적이고, 원하는 만큼의 IT자원을 언제든지 확보할 수 있는 가용성을 제공한다. 따라서, 사용자로 하여금 클라우드 컴퓨팅이 활성화된 미래에 대한 상당한 기대를 가지게 한다[4,10,19,20]. 그러나, 가상화를 통한 거의 무한한 컴퓨팅 파워와 데이터 저장공간 제공 등의 많은 장점에도 불구하고[3], 클라

※ 본 연구는 방송통신위원회의 방송통신정책센터운영지원사업(KCA-2011-1194100004-110010100)의 연구결과로 수행되었음.
* 서울대학교 컴퓨터공학부 분산컴퓨팅시스템 연구실(iyjung@dcslab.snu.ac.kr, ischo@dcslab.snu.ac.kr, yjyu@dcslab.snu.ac.kr)
논문번호 : KICS2011-05-228, 접수일자 : 2011년 5월 25일, 최종논문접수일자 : 2011년 9월 20일

우드 컴퓨팅 환경이 청사진대로 활성화되기 위해서는, 기밀성과 비밀성을 보장하는 정보의 보호 및 클라우드에 저장된 정보에 대한 신뢰성 확보란 과제를 해결해야 한다[10].

Provenance는 데이터의 생성부터 현재까지의 처리 이력 및 중요사항들을 기록해 놓은 메타데이터로 정의된다[9]. 즉, 데이터의 provenance는 데이터의 최초 버전을 누가 생성했고 현재의 데이터 상태에 이르기까지 누가 그 내용을 수정하고 어떤 처리를 가했는지에 대해 알려준다. Provenance는 데이터의 공유가 광범위하고 익명적으로 발생하는 클라우드 환경에서 상당히 중요한 정보가 된다. Provenance가 없이는 클라우드 환경에서 데이터의 출처와 신원을 밝히는 것이 불가능하기 때문이다[1]. 구체적인 예로, Amazone의 "Public Data Sets on AWS"는 GenBank[8]와 US census data, PubChem[7] 의 데이터 집합(data sets)을 위한 무료 저장소를 제공한다. 연구자들이 이 데이터를 활용해 연구업적을 내기 위해서는 이들 데이터의 출처 및 처리과정을 정확히 규명해야 하는 과제를 가지게 된다. Provenance는 현재의 데이터가 존재하기까지의 모든 처리과정과 데이터의 원천에 대해 규명할 수 있는 필수 정보를 제공한다. 또한 provenance는 실험 결과를 개선시켜 연구의 질을 높이는데도 기여할 수 있는 유용한 정보가 된다. 이런 provenance는 시간의 흐름과 데이터의 처리방향에 따라 방향성있는 싸이클없는 그래프(Direct Acyclic Graph: DAG)로 표현될 수 있다. DAG로 표현된 Open Provenance Model(OPM)[11]은 시간에 따른 데이터의 진화와 관련 정보를 파악하기 쉽도록 provenance를 정형화하고 있다. OPM에서 provenance를 나타내는 다이어그램의 구성요소로서는 데이터나 문서 같은 provenance 정보의 기술대상이 되는 article들, 이런 artifact의 상태 변화의 원동력이 되는 process 들과 article들의 어떤 관계와 상태전이를 표현하는 arc가 있다. 구체적으로 어떤 관계나 어떤 성격의 상태변이인지에 대한 부분은 provenance의 세부 기술에 의해 나타나게 된다. Arc는 artifact들 간의 의존관계 및 인과관계(causality)를 나타내고, 동일한 신원(identity)을 가지는 데이터는 상태를 변화시키는 process에 의해 서로 다른 artifact로 OPM에서 나타나게 된다. 즉, artifact는 데이터의 버전(version, state)으로 볼 수 있다. 하나의 원천에서부터 파생되어 현재의 artifact까지 이르는 arc와 artifact들의 연결을 provenance chain이라고도 부른다[9].

한편, provenance는 데이터의 신뢰를 가늠할 수 있도록 해주는 유용한 메타정보이지만, 이 정보 역시 그 유효성에 대한 검증과 확인이 필요하다. 특히 provenance를 기술하는 정보는 데이터가 진화할수록 그 양이 많아지는 특성이 있다. 데이터는 버전이 변화하고 상태 전이 전의 버전이 남아있지 않은 경우가 일반적이지만, provenance는 데이터의 진화이력이기 때문에 추가만이 가능하고 삭제되거나 변경이 일어나지 않는 정보이다. 시간이 흐를수록 provenance의 양은 계속 늘어나게 되어, provenance의 확인을 통한 데이터의 처리이력 확인 및 데이터의 원천을 확인하는 작업은 시간이 많이 걸리는 부담스러운 일이 된다. 즉, 정보통신의 발달로 정보는 전자문서의 형태로 저장, 검색, 추출되는 것이 일상이 되었고, 정보의 복사 및 쉬운 편집도 삶을 편리하게 해주는 요소가 되었다. 이런 편리함은 데이터의 신뢰 문제를 제기하였고, 데이터에 대한 신뢰를 판단할 수 있는 근거로 provenance가 제안되었으나, provenance를 통한 데이터의 신뢰확인 부분은 데이터의 처리 이력이 증가함에 따라 이를 나타내는 provenance chain도 길어지게 되고, 이의 추적을 통한 데이터의 출처 확인과 현재 버전의 데이터 신뢰성 확인은 또 하나의 문제를 제기하기에 이르렀다. Provenance가 데이터 신뢰 확인의 근간이 되는 가치있는 정보가 되기 위해서, 또 provenance를 통한 데이터의 신뢰확인이 의미를 가지기 위해서는 데이터 뿐만아니라 provenance의 신뢰확인부터 해결이 되어야 한다. 따라서, 본 논문에서는 클라우드 컴퓨팅 환경에서 provenance를 통한 데이터 신뢰확인에서 이슈가 되는 효과적인 provenance의 신뢰확인을 통한 데이터의 신뢰확보 방안을 제안한다.

본 논문의 구성은 2장에서 클라우드 컴퓨팅 환경에서 provenance 관리 모델을 기술하고, 효과적인 provenance와 데이터의 신뢰검증 방안을 제안한다. 3장에서는 본 제안의 오버헤드를 여러 관점에서 분석하여 제안의 효용과 가치를 평가한다. 4장에서는 관련 연구 분석을 통해 본 제안의 의미를, 5장에서는 결론과 향후 연구계획을 기술한다.

Ⅱ. 클라우드 컴퓨팅 환경상의 Provenance를 통한 데이터 신뢰확보

2.1 Provenance 관리를 위한 클라우드 컴퓨팅환경

데이터의 신뢰확인을 위한 provenance의 관리 및 검증 장치를 포함하는 클라우드 컴퓨팅 환경은 그림 1과 같다. C_0는 하나의 클라우드 컴퓨팅 환경을 나타낸다. C_0를 관리하는 믿을 수 있는 System Manager (SM)가 있고, C_0의 클라우드 서비스를 제공하는 Service

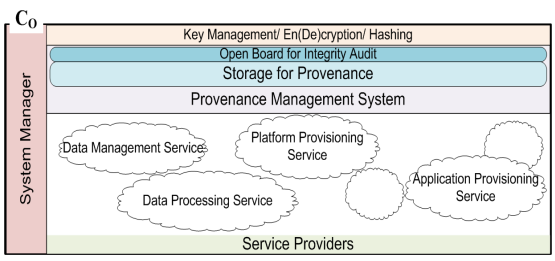


그림 1. Provenance 관리를 수용한 클라우드 컴퓨팅 환경

Provider(SP)들이 있다. 그리고, SP에 의해 C_0상에서 제공되는 데이터 관리서비스(Data Management Service), 응용프로그램을 개발할 수 있는 플랫폼 제공 서비스(Platform Provisioning Service), 고사양의 자원을 많이 요구하는 데이터 처리 서비스(Data Processing Service), 사용자가 원하는 응용 프로그램 제공 서비스(Application Provisioning Service) 등에 대한 각 사용자의 서비스 이용이력 및 데이터의 저장 처리 이력을 관리하는 Provenance Management System이 있다. 본 논문은 데이터 관리 서비스의 provenance 관리 시스템을 대상으로 논한다. 데이터와 provenance의 무결성과 보안관리를 위한 C_0의 키관리 및 암복호화 관리를 담당하는 부분이 provenance management system과 연계되어 SM의 관리를 받는다. 관리가 되는 키는 C_0의 개인키 Pr_{C0}가 된다. 그리고, hash function, H와 C_0의 공개키 Puc_0는 공개된다. Provenance Management System 상에는 provenance를 저장하는 저장소와, 저장된 provenance와 provenance에 매칭되는 각 data 상태, 즉, 각 artifact(데이터 버전)의 무결성에 대한 검증이 가능한 공개 보드로서 Open Board for Integrity Audit(OBIA)이 존재한다.

2.2 클라우드 컴퓨팅 환경상의 data와 provenance

그림 2는 C_0상에 A라는 신원(identity)을 가지는 데이터가 시각 t_0에 생성되어 $A_C[t_0]$버전으로 저장된 것부터 시작해서 현재의 시각 t_c에 $A_C[t_c]$버전으로의 진화를 보여주는 provenance chain을 OPM으로 나타내고 있다. A의 시각에 따른 각 버전은 원형표기의 artifact를, A의 버전 변화, 즉 A의 진화를 유도한 process, $p(A_C[t_i])$들은 사각형으로 나타나고 있다. 각 artifact들 사이의 전이는 OPM에서 정의된 대로 arc로 나타나고 있다. 사용자는 OPM의 모든 요소에 대해 중요한 또는 필요한 정보를 provenance로 남기게 된다. 이는 누가 언제 어떤 데이터를 저장했는지를 자동으로 기록하는 일반 로그보다 데이터 및 데이터의 진화과정의 semantic을 강조한 메타 정보가 더 풍부하

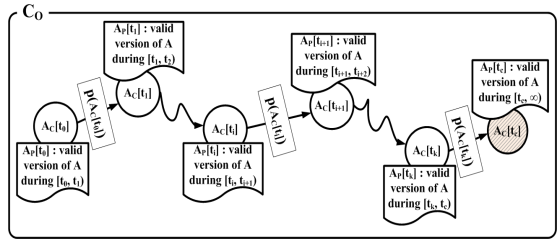


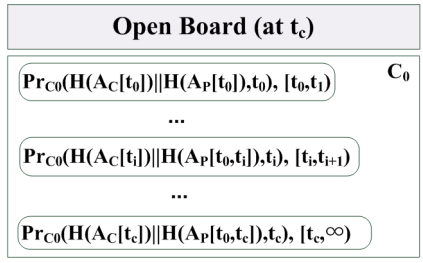
그림 2. 클라우드 컴퓨팅 환경상의 OPM

게 들어있고 기록하는 사람에 따라 다른 정보 구성이 될 수 있는 특징을 가지게 된다. 이런 provenance를 기록하는 부분은 출력다각형으로 표현되어 있다; 그림 2의 경우는 artifact들에 대한 provenance 만을 나타내고 있는데, 예를들면, $A_C[t_0]$의 provenance는 $A_P[t_0]$으로 표현된다. 사용자는 데이터 보관과 관리를 위한 대용량의 저장소와 증가만 하는 데이터의 처리 이력 관리를 손쉽게 하고, 언제 어디서나 인터넷을 통해서 저장된 데이터에 접근할 수 있는 편의성 때문에 클라우드가 제공하는 데이터 관리 서비스를 사용한다. 그런데, 가장 큰 문제는 내가 보관한 데이터 모두가 제대로 보관되고 있는지를 확인하고 싶은데, 단순히 클라우드를 믿는 방법 이외에는 이를 확인할 방법이 현재의 클라우드에서는 제공되고 있지 않는 부분이 문제가 된다. 본 논문에서는 데이터와 provenance를 클라우드에 보관을 하지만, 이에 대한 저장 무결성에 대한 확인 근거는 공개적으로 포스팅을 함과 동시에 사용자가 보관을 하는 방법을 이용해서 저장 무결성을 확인하는 방법을 제안한다.

3. 효율적인 무결성 확인에 기반한 provenance를 통한 데이터의 신뢰 확인

그림 2의 Provenance chain 상의 각 데이터 버전, 즉, 각 artifact와 이들에 관련된 provenance의 검증은 Open Board를 통해 이뤄지게 된다. 즉, 데이터의 상태전이가 t_i에 일어나서 C_0에 데이터를 저장할 때, C_0는 $A_C[t_i]$를 데이터 저장소에 저장함과 동시에 $A_P[t_i]$를 Provenance Management System을 통해 입력받는다. 이 때, $A_P[t_i]$는 provenance 저장소에 저장되고, $H(A_C[t_i])$와 $H(A_P[t_i])$가 계산된다. 그리고, t_i에는 이미 $[t_{i-1},\infty)$에 유효한 artifact, $A_C[t_{i-1}]$에 대한 $H(A_C[t_{i-1}])||H(A_P[t_0,t_{i-1}])$가 Open Board에 게시가 된 상태이다.

$$H(A_P[t_0,t_i])=H(A_P[t_0,t_{i-1}])||H(A_P[t_i])$$
$$=H(A_P[t_0])||H(A_P[t_1])||\cdots$$
$$||H(A_P[t_{i-1}])||H(A_P[t_i])$$



$$H(A_P[t_0,t_c]) = H(A_P[t_0])||H(A_P[t_1])||\cdots||H(A_P[t_i])||H(A_P[t_{i+1}])||...||H(A_P[t_k])||H(A_P[t_c])$$

그림 3. 데이터와 provenance의 무결성 확인을 위한 OBIA

로 $H(A_P[t_0,t_i])$가 계산이 되면, Provenance Management System은 $H(A_C[t_i])||H(A_P[t_0,t_i])$를 C_0의 private key, Pr_{C0}로 암호화해서 이를 $A_P[t_i]$버전의 유효기간 $[t_i,\infty)$와 같이 Open Board에 게시를 하게 된다. 이 때, $H(A_C[t_{i-1}])||H(A_P[t_0,t_{i-1}])$의 유효기간은 $[t_{i-1},t_i)$으로 정정된다. 데이터를 저장하는 사용자는 $Pr_{C0}(H(A_C[t_i])||H(A_P[t_0,t_i]),t_i),[t_i,\infty))$를 local computer에 다운을 받아 이후 $A_C[t_i]$의 무결성을 검증하고자 할 때를 위해 저장을 해 둘 수 있다. 이 정보로 알 수 있는 것은 $A_C[t_i]$의 무결성과 더불어 $A_C[t_i]$가 있기까지의 provenance의 무결성이다. Open Board에 게시된 정보는 누구에게나 공개되는 정보이지만, 해쉬의 특성상 원래의 데이터를 유추할 수가 없기 때문에 데이터의 privacy를 보호할 수 있다. 또한 해쉬값은 길이가 원본 데이터에 비해 작고 일정하고 이를 클라우드의 SM이 C_0의 개인키로 인증을 해주는 과정에서 암호화되어 빠르고 안전하게 검증할 수 있는 방안이 된다. 공개키는 공개가 되는 키이기 때문에 Open Board의 정보를 검증해보고자 하는 사용자는 누구나 C_0의 공개키로 게시된 정보의 신뢰성을 확인할 수 있고, hash 값의 비교로 각 artifact의 무결성은 물론, provenance의 무결성까지 확인할 수 있다.

Ⅲ. 평 가

본 제안의 운영 오버헤드를 분석해보면 다음과 같다.

실험환경은 Pentium 4 CPU 3.20 GHz, 3G RAM 상에서 crypto++ library 5.5.2[21]와 QuickHash library[22]를 이용하여, hash와 RSA PKI 알고리즘을 적용하였다. 각 artifact의 provenance는 artifact의 생성자, 생성시각, 생성환경, 참고자료와 메모로 구성되고, 1KB이하로 두었다. Artifact는 데이터 사이즈 300MB, 200MB, 1000MB, 20MB, 2MB, 81KB 에 대해 측정을 해보았다. 본 제안에서의 시간적 오버헤드는 hashing time, PKI 기반의 encryption time, 그리고, PKI 기반의 decryption time, hash verification time을 들 수 있다. 이들에 대한 구체적인 오버헤드는 다음과 같이 측정되었다.

Hashing time은 SHA1을 기준으로 데이터 사이즈 100MB ~ 300MB는 평균 2 sec, 20MB는 0.2 sec, 2MB 이하는 0.03 sec 이하가 되고, SAH256을 기준으로는 5 sec, 0.5 sec, 0.06 sec 이하로 측정되었다. Hash 검증 시간은 같은 조건에서 SHA1의 경우, 3 sec, 0.2 sec, 0.03 sec 이하, SHA 256 기준으로는 6 sec, 0.5 sec, 0.07 sec 이하가 되었다. 그리고, hash 값에 대한 C0의 encryption time은 SAH1의 경우, 키 길이 1024의 RSA 1024의 경우는 0.003 sec 이하, 2048의 RSA 2048의 경우 ~0 sec로 측정되었다. Decryption time은 RSA 1024의 경우 0.14 sec 이하, RSA 2048의 경우는 0.03 sec 이하로 측정되었다.

그리고, 공간적 오버헤드는 데이터의 전이가 일어나서 provenance가 추가될 때 Open board에 게시되는 정보의 양으로 볼 수 있는데, 새로운 artifact 당 Open board에 추가되는 정보의 양은 표 1과 같다.

본 제안의 시간과 공간 오버헤드를 전반적으로 분석해보면, 클라우드에 저장되는 데이터의 크기에 따라 달라지기는 하지만, 20MB 정도의 데이터라고 하더라도 각 artifact당 1초 미만, 2MB 이하는 0.1 sec가 채 되지 않는 오버헤드를 가진다. 사양이 좋은 클라우드 상의 IT자원을 이용한다면 이 오버헤드는 인터넷을 통한 클라우드에의 데이터 저장시간에 비해 추가적인 오버헤드를 발생시키지 않을 것임이 예상된다.

본 논문의 기여점은 다음과 같다.

첫째, 클라우드 컴퓨팅환경에서 데이터의 신뢰 확인을 위한 provenance를 도입할 수 있는 모델을 제안했다. 둘째, provenance를 통한 데이터의 신뢰확인은 provenance의 무결성부터 검증이 되어야 의미가 있기 때문에, 본 제안에서 데이터의 무결성은 provenance

표 1. hashing 후 C_0의 개인키로 암호화한 후의 정보 크기

	Hash Algorithm		PKI Algorithm	
	SHA1	SHA256	RSA 1024	RSA 2048
Size (B)	20	32	256	256

의 무결성과 함께 검증되어 데이터 신뢰성 확인에 더 신뢰를 주게 된다. 셋째, provenance를 통한 데이터의 신뢰확인은 provenance chain 상의 artifact의 인과관계(causality)를 따라, 즉, arc를 따라 검증을 해가면, provenance chain의 길이만큼의 검증 오버헤드가 든다. 그러나, 본 제안에서는 각 artifact의 해쉬값을 저장함으로써 과거 데이터의 무결성을 증명하고, provenance 역시 데이터의 진화에 따라 그 양이 커져 검증이 힘든 부분을 해쉬를 통한 증빙으로 그 무결성을 검증할 수 있다. 넷째, 본 제안에서는 open board를 통한 해쉬값을 게시함으로써 누구나 검증이 가능한 투명한 검증을 제공하고, 해쉬값은 사용자의 확인하에 게시가 되는 것이기 때문에 게시된 정보에 대한 고의적인 변경이 힘들도록 했다. 사용자도 게시된 정보는 local computer에 저장을 해서 이후에 검증을 위한 자료로 활용할 수 있기 때문에 고의적인 삭제가 의미가 없도록 했다. 다섯째, 클라우드 컴퓨팅 환경에서 provenance management system이외의 다른 장치를 특별히 요구하지 않기 때문에 적용이 쉽고, 본 제안의 운용을 위한 시간적, 공간적, 자원 요구량 관점에서의 오버헤드는 크지 않아 효율적으로 provenance를 통한 데이터의 신뢰확인을 할 수 있는 장점이 있다.

Ⅳ. 관련연구

클라우드 컴퓨팅 환경에서의 데이터 신뢰 문제는 클라우드 컴퓨팅의 성공을 좌우하는 중요한 문제이다. 현재까지 클라우드 컴퓨팅의 실현 문제, 즉, 실제 사용자에게 제공 가능한 서비스인가를 놓고 많은 타진을 해왔었고, 이제 그 실현을 목전에 두고 있다. 클라우드 컴퓨팅이 활성화되기 위해서 사용자의 중요한 정보를 훼손없이 잘 보관하고 사용자가 원하는 때 제공할 수 있는 확신을 주는 일이 급선무이다.

분산 워크플로우 기반 그리드 환경에서 provenance에 대한 연구들이 있었다[23-26]. 그런데, 기존 연구들은 기반 system component들을 바꿀 수 있는 가정 하에 데이터의 신뢰확인을 위해 provenance를 이용하는 방안을 보여주고 있다. 클라우드 환경은 기반 시스템 환경 및 서비스들을 제어할 수 없는 환경이다. 즉, 클라우드 컴퓨팅 환경의 기반 서비스와 인프라에 변경을 가하지 않으면서 provenance를 이용하여 데이터 신뢰를 확인하는 방안이 바람직하다. 본 제안에서는 클라우드 환경의 서비스나 인프라에 어떤 제한도 가하지 않으면서 provenance를 관리하는 시스템을 모델화하고 있다. 그리고, 연구 [1]은 단순히 Cloud에 provenance를 저장하고 이를 access하는 protocol과 모델을 제시하는데 그치고 있다.

Provenance를 이용하여 데이터의 신뢰를 확인하는 방법으로는 보통은 provenance chain 상의 데이터의 causality를 이용하여 provenance와 데이터를 확인하는 방법을 사용한다[27,28]. 이 방법은 provenance chain을 현재의 데이터 버전으로부터 역으로 이전 버전의 데이터를 추적하는 일이기 때문에 데이터 처리가 가역적이지 않으면 이전 버전의 데이터를 확인하기 힘들다는 문제가 있다. 또한, 가역적인 데이터 처리로 이전 버전의 데이터를 복원할 수 있다고 하더라도 causality의 역추적으로 provenance chain 상의 모든 artifact들을 원하는 데이터 버전이 나올 때까지 추적을 해야 데이터의 무결성을 확인할 수 있는 시간적인 오버헤드가 큰 작업이 된다. 또한, provenance의 무결성은 확인할 수 없는 허점도 있다.

따라서, 본 논문의 제안은 데이터와 provenance의 무결성을 모두 확인하여 provenance를 이용한 데이터의 신뢰확인의 방법적인 면에서도 신뢰를 획득할 수 있고, 시간적, 공간적 부담도 크지 않고 적용도 쉬운 효율적인 방안이 된다. 그리고, Cloud 컴퓨팅 환경에서 데이터 신뢰확인이란 과제가 풀리지 않은 현시점에 provenance를 이용하여 데이터의 신뢰를 확인하는 선구적인 모델과 방법으로서의 의의를 가지게 된다. 그리고, 데이터의 신뢰확인에 provenance 이외의 정보는 이용하지 않는 방법으로 타 전문가 시스템 및 reputation system의 평가를 필요로 하지 않는 연구[29]와 같이 가벼운 방법이 된다.

Ⅴ. 결론 및 향후 계획

본 논문에서는 클라우드 컴퓨팅 환경에서 provenance를 통한 데이터의 신뢰확인을 위한 실용적이고 효율적인 방안을 제안하였다. 클라우드 컴퓨팅환경은 IT자원의 가상화를 통해 사용자의 다양한 요구를 수용할 수 있는 비전을 제시하여 학계 뿐 아니라 산업계의 주목을 받고 있다. 그러나, 클라우드 환경이 활성화 되기 위해서는 정보보안 문제가 해결되어야 한다. 본 논문은 클라우드 상의 정보에 대한 provenance를

통한 효과적인 신뢰확인 방안을 제안하였다. 일반적으로 Provenance를 통한 데이터의 신뢰확인이 상당한 부담이 되는 이유는 provenance chain을 따라 데이터의 causality를 검증하는 것이기 때문인데, 본 논문은 데이터와 관련 provenance의 무결성 검증을 해쉬와 클라우드 개인키로의 인증, open board 상의 검증 데이터의 공개 게시를 통해 보안성, 효율성을 동시에 확보하고 있다.

향후 연구는 클라우드 컴퓨팅 환경에서 데이터 신뢰 확인에 대한 질적, 양적 지표 설정으로 본 제안을 확장할 것이다.

참 고 문 헌

〔1〕 Kiran-Kumar Muniswamy-Reddy, Peter Macko, and Margo Seltzer, “Provenance for the Cloud”, in Proceedings of 8th USENIX Conference on File and Storage Technologies (FAST ’10), Feb 2010

〔2〕 Kiran-Kumar Muniswamy-Reddy and David A. Holland, “Causality-Based Versioning”, ACM Transactions on Storage (ACM TOS), Dec 2009

〔3〕 Wikipedia, http://en.wikipedia.org/wiki/Cloud_computing

〔4〕 H. Erdogmus, “Cloud computing: Does nirvana hide behind the nebula?”, IEEE Software 11, 2 (March-April 2009), pp.4-6.

〔5〕 I. Foster, Y. Zhao, I. Raicu, and S. Lu, “Cloud computing and grid computing 360-degree compared”, In Proceedings of Grid Computing Environments Workshop, GCE’08 (Austin, TX, 2008), pp.1-10.

〔6〕 R. Gellman, “Privacy in the clouds: Risks to privacy and confidentiality from cloud computing”, Tech. rep., February 2009

〔7〕 Pubchem, http://pubchem.ncbi.nln.nih.gov/

〔8〕 Genbank, Nucleic Acids Research 36 (Database Issue) (Jan. 2008)

〔9〕 R. Hasan, R. Sion, and M. Winslett, “Introducing secure provenance: problems and challenges”, In Proceedings of ACM workshop on Storage security and survivabilit, StorageSS ’07 (Alexandria, Virginia, USA, October 2007), pp.13-18.

〔10〕 L. M. Kaufman, “Data security in the world of cloud computing”, IEEE Security & Privacy 7, 4 (July-Aug. 2009), pp.61-64.

〔11〕 Luc Moreau, Ben Clifford, Juliana Freire, Joe Futrelle, Yolanda Gil, Paul Groth, Natalia Kwasnikowska, Simon Miles, Paolo Missier, Jim Myers, Beth Plale, Yogesh Simmhan, Eric Stephan, Jan Van den Bussche, “The Open Provenance Model core specification (v1.1)”, Future Generation Computer Systems, 2010, doi:10.1016/j.future. 2010. 07. 005

〔19〕 T. Sterling, and D. Stark, “A high-performance computing forecast: Partly cloudy”, Computing in Science & Engineering 11, 4 (July-Aug. 2009), pp.42-49.

〔20〕 J. Voas, and J. Zhang, “Cloud computing: New wine or just a new bottle?”, IT Professional 11, 2 (March-April 2009), pp.15-17.

〔21〕 Crypto++ library 5.5.2, available at http://www.cryptopp.com/

〔22〕 QuickHash Library, available at http://www.slavasoft.com/quickhash/index.htm

〔23〕 Z. CHEN, AND L. MOREAU, “Implementation and evaluation of a protocol for recording process documentation in the presence of failures.” In Proceedings of Second International Provenance and Annotation Workshop (IPAW’08).

〔24〕 I. FOSTER, J. VOECKLER, M. WILDE, AND Y. ZHAO, “The Virtual Data Grid: A New Model and Architecture for Data-Intensive Collaboration.”, In CIDR (Asilomar, CA, Jan. 2003).

〔25〕 P. GROTH, L. MOREAU, AND M. LUCK, “Formalising a protocol for recording provenance in grids. In Proceedings of the UK OST e-Science Third All Hands Meeting 2004 (AHM’04) (Nottingham,UK, Sept. 2004). Accepted for publication.

〔26〕 Y. L. SIMMHAN, B. PLALE, AND D. GANNON, “A framework for collecting provenance in data-centric scientific workflows”, In ICWS ’06: Proceedings of the IEEE International Conference on Web Services (2006).

〔27〕 R. Bose, “A conceptual framework for

composing and managing scientific data lienage", In Proceedings of the 14th International Conference on Scientific and Statistical Database Management, pp.15 - 19, 2002.

〔28〕 A. de Keijzer and M. van Keulen, "Quality measures in uncertain data management", Scalable Uncertainty Management, 4772:pp.104 - 115, 2007.

〔29〕 A. Chapman, B. Blaustein and C. Elsaesser, "Provenance based belief", in Proceedings of the 2nd Workshop on the Theory and Practice of Provenance, July 2010.

정 임 영 (Im Young Jung) 정회원

1993년 2월 포항공과 대학교 화학과 졸업
1999년 2월 서울대학교 전산과학과 졸업
2001년 2월 서울대학교 컴퓨터공학부 석사
2010년 8월 서울대학교 컴퓨터공학부 박사
<관심분야> 클라우드 컴퓨팅, 데이터 및 시스템 보안, 스토리지 시스템, 분산컴퓨팅시스템

조 인 순 (Insoon Jo) 정회원

2004년 2월 서울대학교 컴퓨터공학부 졸업
2006년 8월 서울대학교 컴퓨터공학부 석사
2006년 9월~현재 서울대학교 컴퓨터공학부 박사과정
<관심분야> 클라우드 컴퓨팅, 분산컴퓨팅시스템, 인터넷보안

유 영 진 (Youngjin Yu) 준회원

2006년 2월 서울대학교 전기컴퓨터 공학부 졸업
2008년 2월 서울대학교 전기컴퓨터 공학부 석사
2008년 3월~현재 서울대학교 전기컴퓨터공학과 박사과정
<관심분야> 클라우드 컴퓨팅, 스토리지 시스템, 분산컴퓨팅시스템