2010년 12월

10ZS1120-01-7060P

60

유비쿼터스 환경에서의 정보보호 서비스를 위한 프라이버시 강화 암호 기술 개발

Development of Privacy Enhancing Cryptography on Ubiquitous Computing Environment

ETRI 한국전자통신연구원
Electronics and Telecommunications Research Institute

인 사 말 씀

20세기 후반 시작된 정보화 혁명은 IT 환경의 급격한 발전을 이끌었고, 이를 바탕으로 사회 전반적인 변화가 이루어지고 있습니다. 특히 최근 iPhone을 시작으로 한 스마트 폰의 열풍으로 인한 사회·경제적인 파급 효과가 폭넓게 확산되고 있으며, Clouding Computing, RFID/USN, 차량 네트워크, 사물 통신 등의 신규 정보 통신 서비스 환경이 도래하고 있습니다.

이에 따라 인간의 사회 활동 및 경제 활동, 사적인 생활에 있어서 인터넷, 모바일 기기 등에 대한 의존성이 높아지면서, 정보 가치의 상승, 정보의 대량화와 함께 정보의 활용도가 높아지고 있습니다. 하지만 이에 따른 역기능으로 통신 도청, 사이버 공간 상에서의 개인 정보 유출 등으로 인한 개인 프라이버시 침해와 같은 다양한 형태의 사회 문제가 발생하고 있습니다. 특히 개인 정보가 실시간으로 수집·저장·가공되어 서비스에 활용되면서, 서비스 이용 과정에서 개인 정보 유출 및 불법적인 접근에 대한 우려가 커지고 있는 실정입니다. 이는 정보보호 환경이 단순한 통신 상의 정보보호에서 생활 속의 정보보호로 바뀌고 있다는 것을 의미하며, 보안이 적용되어야 할 대상이 급격히 확대되고 있다는 것을 의미합니다. 이에 미국, 유럽 등 선진국은 프라이버시 보호에 대한 중요성을 인지하고, 프라이버시 보호 관련 다양한 프로젝트를 통해 관련 기술을 개발하고 있는 중입니다.

이에 본 과제에서는 스마트폰, 자동차, 다양한 임베디드 기기 형태의 유비쿼터스 디바이스에 정보보호 서비스를 제공하기 위한 프라이버시 강화 암호 기술 개발을 수행합니다. 연구 결과들은 다양한 정보 통신 서비스의 기반 프리미티브 기술로 활용될 것으로 기대하고 있습니다. 또한 다양한 신규 정보 통신 서비스의 신뢰성을 향상시켜 관련 사업의 활성화 및 국가 경쟁력 강화에 기여할 것으로 확신합니다.

끝으로 연구개발 과제에 참여한 연구원 및 공동연구 기관 관계자 여러분들의 노고를 치하하는 바입니다. 앞으로 여러분 개개인의 열정으로 개발된 연구 결과물이 우리나라 정보통신 및 정보보호 기술 발전에 큰 기여가 있기를 기대합니다.

2010 년 12 월

한국전자통신연구원 원장 김 흥 남

제 출 문

본 연구보고서는 주요사업인 "유비쿼터스 환경에서의 정보보호 서비스를 위한 프라이버시 강화 암호 기술 개발"의 결과로서, 본 과제에 참여한 아래의 연구팀이 작성한 것입니다.

2010 년 12 월

주관연구기관 : 한국전자통신연구원

연구책임자 : 책임연구원 홍도원

연구참여자 : 책임연구원 황승구

책임연구원 조현숙

책임연구원 은성경

책임연구원 장구영

선임연구원 김건우

선임연구원 이상수

선임연구원 김영수

선임연구원 길연희

선임연구원 조수형

선임연구원 이주영

선임연구원 최정운

선임연구원 조남수

연구원 윤택영

공동연구기관 : 고려대학교 산학협력단

연구책임자 : 책임연구원 홍석희

연구참여자 : 선임연구원 정익래

선임연구원 성재철

연구원 이제상

연구원 이유섭

연구원 강진건

연구원 문덕재

연구원 정기태

연구원 김희석

연구원 김성경

연구원 서석충

연구원 조영인

연구원 천지영

연구원 노건태

연구원 김윤규

요 약 문

Ⅰ. 제 목

유비쿼터스 환경에서의 정보보호 서비스를 위한 프라이버시 강화 암호 기술 개발

Ⅱ. 연구목적 및 중요성

가. 연구개발의 목적

자동차, 스마트폰, 다양한 임베디드 기기 형태의 유비쿼터스 디바이스에 정보보호 서비스를 제공하기 위한 경량 암호 핵심 기술 및 프라이버시 보존형 데이터 처리 기술을 포함하는 프라이버시 강화 암호 기술 개발

나. 연구개발의 중요성

- 20세기 후반 시작된 정보화 혁명은 IT 환경의 급격한 발전을 이끌었고, 이를 바탕으로 지금까지와는 다른 IT 패러다임과 이에 따른 삶의 변화가 진행되고 있음

- 새로운 패러다임 및 신규 서비스 환경의 등장에 따라 프라이버시 침해와 같은 역기능에 대한 우려가 점차 커지고 있으며, 이에 이를 방지하기 위한 새로운 핵심 암호 기술의 필요성이 커지고 있음

○ 따라서 진화하고 있는 차세대 IT 서비스 환경에서의 안전한 서비스 제공을 위한 경량 암호 기술 및 프라이버시 보존형 데이터 처리 기술 등과 같은 프라이버시 강화 암호 기술의 개발이 중요함

Ⅲ. 연구내용 및 범위

본 과제는 2010년부터 2012년까지 3년간 진행되며 최종 목표는 다음과 같다.

○ 유비쿼터스 환경에서의 프라이버시 침해 방지를 위한 프라이버시 강화 암호 기술 개발
- 경량 암호 핵심 기술 개발
- 프라이버시 보존형 데이터 처리 기술 개발
- 양자 서명 프로토콜 개발

연도별 내용 및 범위는 다음과 같다.

가. 1차년도 (2010년) : 프라이버시 강화 암호 핵심 설계 논리 개발

○ 경량 hash 알고리즘 설계 논리 개발
- 경량 hash 후보 알고리즘 3~4종 설계

○ 경량 그룹 서명 프로토콜 설계

○ 순서 보존 암호화 설계 논리 개발
- 순서 보존 암호화 핵심 프리미티브 설계

나. 2차년도(2011년) : 프라이버시 강화 암호 프리미티브 설계

- ○ 경량 hash 알고리즘 설계
  - • 경량 hash 알고리즘 최종안 설계
- ○ 경량 그룹 서명 프로토콜 개발
- ○ 순서 보존 암호화 기술 개발
- ○ 양자 암호 서명 프로토콜 설계

다. 3차년도(2012년) : 프라이버시 강화 암호 기술 개발

- ○ 경량 hash 함수 개발
  - • 최종 알고리즘에 대한 심층 안전성 평가 및 개선
  - • 최종 알고리즘에 대한 SW/HW 구현 및 효율성 평가
- ○ 경량 인증 기술 개발
- ○ 암호 데이터 연산 기술 개발
- ○ 양자 암호 서명 프로토콜 개발

Ⅳ. 연구결과

본 과제의 수행을 통하여 확보된 당해연도(2010년) 주요 연구개발 결과는 아래와 같다.

- ○ 경량 hash 알고리즘 핵심 설계 논리 개발
  - • 경량 hash 후보 알고리즘 4종(LH1, LH2, LH3, LH4) 설계
- ○ 경량 그룹 서명 프로토콜 설계

- Lattice 기반의 경량 그룹 서명 프로토콜(링 서명, 그룹 서명) 설계

○ 순서 보존 암호화 설계 논리 개발

- Bucket 기반 순서 보존 암호화 및 pivot을 사용한 순서 보존 암호화 기법 초안 설계

○ 논문

- SCI(E) 논문 4건 게재

○ 특허

- 국내 특허 6건 출원

Ⅴ. 기대성과 및 건의

가. 기술적 측면

○ 유비쿼터스 환경에서의 중요성이 크게 부각되는 경량 암호 기술 및 프라이버시 보호 핵심 암호 기술 개발을 통해 차세대 IT 환경을 위한 핵심 기술 선점이 가능

○ 경량 모바일 기기의 사용이 증가하면서, 요구가 증대되는 경량 암호 기술 및 핵심 특허 선점을 통해 미래 시장 경쟁에서 주도적인 위치를 확보할 수 있음

○ 프라이버시 침해 가능성 우려로 도입되지 못한 IT 서비스와 결합하여 국내 IT 기술의 신뢰성 향상이 가능

○ 양자 서명 기술 개발을 통해 QKD에 제한되어 왔던 양자 암호 프로토콜의 응용 분야를 확대하고, 무조건적인 안전성이 보장되는 다양한 네트워크 통신 시스템의 실현

나. 경제 산업적 측면

- ○ 개발 기술 활용으로 프라이버시 침해를 방지하여 다양한 유비쿼터스 네트워크 서비스 산업의 활성화에 기여
- ○ 차세대 IT 환경을 위한 핵심 암호 기술을 통해 Cloud Computing, RFID/USN, 차량 네트워크, 위치 기반 서비스, 사물 통신 등 신규 응용 서비스의 신뢰성을 향상시켜 관련 산업 활성화 및 수출 확대에 기여
- ○ 프라이버시 침해에 대한 우려로 구축할 수 없었던 민감한 정보를 다루는 정보 통신 서비스에 대한 실용화를 가능하도록 하여 관련 사업의 활성화 기대
- ○ 양자 서명 기술 연구를 통해 양자 암호 프로토콜의 응용 범위를 확대하고, 기존의 디지털 산업, 이동통신 산업, 스마트카드 산업 및 근거리 통신 산업과의 융합을 통해 새로운 시장 및 제품 창출 가능성이 높음
- ○ 특허 기술, IP 형태의 핵심 기술 선점으로 국가 경쟁력 확보가 가능하며, 개발된 기술의 국내 업체 이전을 통한 국내 업체 경쟁력 강화 및 지적 재산권에 대한 간접 수익 기대 가능
- ○ 국내 IT 보안 산업 시장은 2009년 2,230억원 규모에서 연평균 10.5% 성장하여 2013년 3,480억원으로 성장 예상 (IDC Korea 2009.7, KISA 2008.12)
- ○ 세계 IT 보안 산업 시장은 연평균 9.3%의 성장률을 보여 2013년 190억 달러 이상에 이를 것으로 전망(Gartner 2009.7)

ABSTRACT

Ⅰ. TITLE

Development of Privacy Enhancing Cryptography on Ubiquitous Computing Environment

Ⅱ. THE OBJECTIVE AND IMPORTANCE

A. Objective

Development of privacy enhancing cryptography including core technologies of light-weight cryptography and privacy preserving data processing technologies for providing information security services on various ubiquitous devices such as vehicles, smart-phone, and other embedded devices

B. Importance

- ○ Information revolution, which had started at late 20th century, leaded rapid development of information-technology(IT) environment. As a result, we are faced with a new IT-paradigm and changes of lifestyle
- ○ Arising of new paradigm and new service environments, concerns for an

adverse effect such as privacy violation are growing. In order to prevent the problem the necessity of new cryptographic technologies is also getting stronger

○ Therefore, development of privacy enhancing cryptography such as core technologies of light-weight cryptography and privacy preserving data processing technologies for providing secure services on evolving next-generation IT service environment is important

Ⅲ. THE CONTENTS AND SCOPE OF THE STUDY

This project has been under development since 2010 and will be carried out by 2012 for 3 years. The goals are expressed as follows.

○ Development of privacy enhancing cryptography for preventing privacy violation in ubiquitous computing environment
  - Development of core technologies of light-weight cryptography
  - Development of privacy preserving data processing technologies
  - Development of quantum signature protocols

Contents and extent by the year are listed as follows.

A. 1st Year (2010) : Development of design logic for privacy enhancing cryptography

○ Development of design logic for light-weight hash algorithm

- Design of 3~4 candidates for light-weight hash algorithm

○ Design of light-weight group signature protocol

○ Development of design logic for order-preserving encryption algorithm

- Design of primitive modules for order-preserving encryption algorithm

B. 2nd Year (2011) : Design of primitive modules for privacy enhancing cryptography

○ Design of light-weight hash algorithm

- Design of final plan for light-weight hash algorithm

○ Development of light-weight group signature protocol

○ Development of order-preserving encryption algorithm

○ Design of quantum signature protocol

C. 3rd Year (2012) : Development of privacy enhancing cryptography

○ Development of light-weight hash algorithm

- In-depth evaluation and improvement for security of the final algorithm
- SW/HW implementation and efficiency evaluation for the final algorithm

○ Development of light-weight authentication protocols

○ Development of protocols for secure computation of encrypted data

○ Development of quantum signature protocols

Ⅳ. RESULTS

The main results procured by performing the project in this year(2010) are as follows.

- ○ Development of design logics for light-weight hash algorithm
  - • Design of 4 candidates for light-weight hash algorithm (LH1, LH2, LH3, and LH4)
- ○ Design of light-weight group signature protocol
  - • Design of lattice-based light-weight group signature protocols (ring signature and group signature)
- ○ Development of design logics for order-preserving encryption algorithm
  - • Design of draft algorithms using bucket-based technique and pivoting method
- ○ Papers
  - • 4 SCI(E) papers published
- ○ Patents
  - • 6 domestic patents pending

Ⅴ. EXPECTED RESULT & PROPOSITION

A. Technical Aspect

- ○ Preoccupancy of core technologies for next generation

IT-environment by developing light-weight cryptography and privacy preserving cryptography whose importance is increasing in ubiquitous computing environment

- ○ Acquisition of advantageous position for competing in future market, by preoccupying core technologies and patents for light-weight cryptography whose need is increasing according to growing use of mobile devices
- ○ Combining IT services which could not be introduced by the reason of privacy violation, improving credibility of domestic IT technologies becomes possible
- ○ Through development of quantum signature technologies, enlargement of applicable area of quantum cryptographic protocols and realization of various network communication systems which guarantee unconditional security

B. Economical and Industrial Aspect

- ○ By preventing privacy violations, developed technologies will contribute to activation of various ubiquitous network service businesses
- ○ Fundamental cryptographic technologies for next generation IT environment will improve credibilities of newly developed application services such as Cloud Computing, RFID/USN, vehicular network, location based service, and M2M(machine-to-machine) network, and consequently contribute to activation of related businesses and augmentation of exportation
- ○ By enabling commercialization of information communication services

dealing with sensitive data which could not be realized from worries about privacy violation, activation of related businesses is expected

- Through researches of quantum signature technologies, applicable area of quantum cryptographic protocols will enlarge and new market and product will be created by converging to existing digital industry, mobile communication business, smart card business, and local area network communication business
- Acquisition of national competitive power by preoccupying intellectual properties such as patents and fundamental technologies, buildup of domestic company's competitive power by transferring developed technologies, and expectation of indirect profits from intellectual property rights
- Domestic market size of IT security business is expected to grow average 10.5% every year from ₩223 billion in 2009 to ₩348 billion in 2013 (IDC Korea 2009.7, KISA 2008.12)
- Global market size of IT security business is expected to grow average 9.3% every year to be over than $19 billion in 2013 (Gartner 2009.7)

CONTENTS

List of Tables

List of Figures

목 차

표 목 차

그 림 목 차

제 1 장 서 론

제 1 장 서 론

제 1 절 연구 개발의 필요성

20세기 후반 시작된 정보화 혁명은 IT 환경의 급격한 발전을 이끌었고, 이를 바탕으로 지금까지와는 다른 IT 패러다임과 이에 따른 삶의 변화가 진행되고 있다. 특히 통신 체계 및 하드웨어의 발달과 더불어, 세계적으로 IT 보안 시장의 흐름이 통신상의 정보보호에서 생활 속의 정보보호로 패러다임의 변화가 이루어지고 있다. 이에 따라 보안이 적용되어야 하는 대상 및 다양한 신규 정보 통신 서비스가 급속히 증가하고 있는 실정이다.

이러한 정보통신 환경의 진화에 따라 '언제 어디서나' 정보에 대한 접근이 가능하다는 것은, '언제 어디서나' 정보 유출 및 변조 가능성을 의미하며, 이에 따라 현재의 정보화 시대에서 경험하고 있는 각종 역기능들이 지금보다 더욱 광범위하게 심각한 위협으로 다가올 것으로 예상된다. 특히 새로운 서비스의 도입으로 인해 사용자 정보에 대한 무차별적 수집, 이용, 관리, 재분배가 심화되어, 수집된 데이터에 대한 불법적인 접근 및 유출에 대한 우려가 더욱 증폭될 것으로 예상된다. 또한 향후 IT 서비스는 주로 소형, 경량화된 모바일 기기를 중심으로 이루어질 것으로 예상되며, 이러한 장치들은 시스템 내부 계산 능력의 한계로 암호화 및 인증을 위한 고비도의 암호 적용이 어려워 전체 네트워크의 보안 위협 요소로 작용할 수 있다.

개인의 프라이버시에 대한 관심이 점점 고조되고 있는 이때 정보보호나 프라이버시 침해를 고려하지 않은 새로운 서비스의 도입은 심각한 부작용을 초래하여 서비스 활성화를 방해하는 큰 장애 요인이 될 수 있다. 이에 자원 제약적인 환경에서 정보보호 서비스를 수행할 수 있도록 하는 필수 보안 기술에 대한 고속 경량화 기술 및 프라이버시 침해를 방지하면서 중요 데이터에 대한 연구, 통계, 마케팅

등의 활용성을 높일 수 있는 프라이버시 보호 데이터 처리 기술 등과 같은 핵심 암호 기술 개발의 중요성이 크게 부각되고 있다. 또한 양자 컴퓨터를 포함한 컴퓨팅 능력의 급격한 발전 및 새로운 암호 공격 기법의 등장 가능성에 따라 향후 현재의 암호 기술 체계가 흔들릴 가능성이 존재하며, 이에 대비해 컴퓨터 계산 능력에 상관없이 무조건적 안전성을 가지는 양자 암호 기술 개발의 필요성이 주목을 받고 있다.

[그림 1-1] 기술 개발 개념도

이에 차세대 IT 서비스에서의 프라이버시 침해 방지를 위한 경량 암호 기술, 프라이버시 보호 핵심 암호 기술 및 양자 암호 프로토콜을 포함하는 프라이버시 강화 암호 기술 개발이 필요하다. 이러한 프라이버시 강화 암호 기술 개발을 통해 Cloud Computing, RFID/USN, 차량 네트워크, 위치 기반 서비스, 사물 통신 등과 같은 신규 정보 통신 서비스의 신뢰성을 향상시켜 유관 산업 활성화 및 수출 확대에 기여할 것으로 전망된다.

제 2 절 연구내용 및 범위

1. 연구개발의 목표

유비쿼터스 환경에서의 정보보호 서비스를 위한 프라이버시 강화 암호 기술 개발 사업의 최종 연구목표 및 세부목표는 다음과 같다.

가. 과제의 최종목표

○ 유비쿼터스 환경에서의 프라이버시 침해 방지를 위한 프라이버시 강화 암호 기술 개발

- 경량 암호 핵심 기술 개발
- 프라이버시 보존형 데이터 처리 기술 개발
- 양자 서명 프로토콜 개발

나. 과제의 세부목표

○ 경량 암호 핵심 기술 개발

- 경량 hash 알고리즘 개발
- 경량 그룹 서명 프로토콜 개발
- 경량 인증 기술 개발

○ 프라이버시 보존형 데이터 처리 기술 개발

- 순서 보존 암호화 기술 개발
- 암호 데이터 연산 기술 개발

○ 양자 암호 서명 프로토콜 개발

2. 당해연도 연구내용

유비쿼터스 환경에서의 정보보호 서비스를 위한 프라이버시 강화 암호 기술 개발 사업의 1차년도(2010년) 연구목표 및 주요 연구내용은 다음과 같다.

가. 연구목표

○ 프라이버시 강화 암호 핵심 설계 논리 개발

- 경량 hash 알고리즘 설계 논리 개발
- 경량 그룹 서명 프로토콜 설계
- 순서 보존 암호화 설계 논리 개발

나. 주요 연구내용

○ 경량 hash 알고리즘 설계 논리 개발

- SHA-3 hash 후보 알고리즘에 대한 설계 논리 분석 및 분류
- 경량 hash 알고리즘에 대한 요구 사항 분석 및 정의
- 설계 논리에 따른 연산 효율성 분석
- 경량 hash 후보 알고리즘 4종(LH1, LH2, LH3, LH4) 설계
  - hash 함수와 블록 암호를 동시에 지원하는 최초의 경량 hash 알고리즘 설계(LH1, LH2)
  - LFSR 기반의 스트림 암호 설계 논리를 도입한 경량 hash 함수 설계 (LH3, LH4)
  - 하드웨어 구현 관점에서 저전력, 저면적의 초경량 hash 알고리즘 설계

○ 경량 그룹 서명 프로토콜 설계

- 경량 그룹 서명에 대한 요구 사항 분석 및 안전성 모델 정립
- 경량 그룹 서명 프로토콜 설계

- Lattice 기반의 효율적인 그룹 서명 기법 설계
- Lattice 기반의 강한 위조 불가능성을 만족하는 효율적인 링 서명 기법 설계
- 안전한 delegation 기반의 익명 인증 시스템 설계

○ 순서 보존 암호화 설계 논리 개발

- 순서 보존 암호화 알고리즘 기본 프리미티브 분석 및 장단점 파악
- 신규 순서 보존 암호화 기법에 대한 안전성 분석 및 개념 정립
- 순서 보존 암호화 프리미티브 설계
  - Bucket 기반 순서 보존 암호화 기법 설계
  - Pivot을 사용한 순서 보존 암호화 기법 설계

제 3 절 국내외 기술개발 동향

1. 세계 기술현황

○ 2000년 초반 기존 hash 함수 안전성에 대한 심각한 문제가 제기되면서 미국 NIST를 중심으로 새로운 hash 함수의 국제 표준을 제정하려는 SHA-3 프로젝트가 진행 중에 있으나, 이는 hash 함수의 안전성에 중점을 두고 있으며 미래 컴퓨팅 환경에서 요구되는 경량 hash 함수에 대한 연구는 아직 진행되고 있지 않음

- 전 세계적으로 가장 널리 사용되고 있는 사용되고 있는 전용 hash 알고리즘은 MD 계열 hash 알고리즘이며, MD2, MD4, MD5, HAVAL, RIPEMD, RIPEMD-128/160, SHA-0, SHA-1, SHA-2 등이 있음
- 2005년 중국의 Wang 교수팀은 처음으로 MD 계열 hash 알고리즘에 대한 일

반화된 공격 기법을 소개하고, MD4, MD5, HAVAL, RIPEMD, SHA-0/1에 대한 충돌 쌍 공격을 수행함. Wang 교수팀은 일반화된 공격 기법을 이용하여 가장 널리 사용되고 있는 MD5, SHA-1에 대한 실제 충돌 쌍을 제시함

- Wang 교수팀이 발견한 MD5, SHA-1 등의 hash 알고리즘에 대한 취약성을 기초로 HMAC-MD5, HMAC-SHA-1, APOP-MD5, X.509 인증서에 대한 충돌 쌍 생성, Postscript 파일 충돌 쌍 분석 등에 관한 결과가 연이어 발표됨
- SHA-0와 SHA-1에 대한 분석 결과에 따라 이후 SHA-2에 취약점이 발생할 것을 대비하여 미국 국립기술표준원(NIST)는 신규 hash 알고리즘 SHA-3 개발을 목표로 2007년부터 2012년까지 6개년의 "SHA-3 개발 프로젝트" 계획을 수립하여 현재 진행 중에 있음
- SHA-3 후보 알고리즘은 다양한 환경에 적합하도록 설계되어 RFID와 같은 자원이 제한된 장치에는 효율적이지 않아, 이후 이러한 환경에 적합한 hash 알고리즘 개발에 관한 연구가 활발할 것으로 예상됨

○ 유비쿼터스 컴퓨팅 환경의 확산과 함께, 제한된 컴퓨팅 환경에 적용 가능한 경량 암호 기술에 대한 요구가 늘어나고 있으며, 이에 따라 USN과 RFID를 대상으로 경량 암호 기술의 연구가 활발히 진행 중임. 그러나 휴대폰, GPS, 스마트 카드 및 각종 모바일 기기 등과 같이 어느 정도의 안전성을 요구하면서 비교적 작은 계산량을 지니는 디바이스에 대한 경량 암호 기술 연구는 미비한 실정임

- 경량 암호 기술 연구는 90년대 센서 네트워크와 RFID 등의 계산량이 극히 제한된 환경에서 암호 응용 기술을 사용하기 위해 처음 연구되었고, 2000년대에 들어서 유비쿼터스 디바이스가 생활 전반에 도입되면서 더욱 다양한 경량 암호 기술에 대한 요구가 생겨나고 있음
  - 현재까지 USN과 RFID를 위한 경량 암호 기술이 주로 연구되고 있으며, 주요 연구 주제로는 경량 암호 알고리즘, 경량 인증, 경량 키 분배 등이 있음

- 계산량, 비밀 키의 저장량 그리고 각 디바이스 사이의 메시지 전송량을 최적화하여 제한된 환경에서 암호 프로토콜의 수행을 가능하도록 하는 것이 경량 암호 기술 연구의 주된 목표임
- 경량 암호 기술은 일반적인 암호 기술에 비해서 낮은 안전성을 지니며 주로 효율성과 안전성의 trade-off를 통해서 경량화를 실현하고 있음

• 유비쿼터스 디바이스가 널리 사용되면서 다양한 컴퓨팅 능력을 지니는 장비들 사이에 정보의 전달이 이루어지고 있지만, 앞으로의 환경에서 요구되는 안전성과 효율성이 동시에 만족하는 경량 암호 기술에 대한 연구는 미비한 실정임
- 경량 인증, 경량 그룹 서명 프로토콜 및 경량 키 분배에 대한 연구가 진행 중 이지만, 현재까지는 이론적인 결과들로 실용적인 효율성에는 도달하지 못하고 있는 실정임

• 앞으로의 연구 방향은 RFID와 USN과 같은 초경량 및 동일한 성능을 지닌 디바이스로 구성된 네트워크를 위한 경량 암호 기술의 연구를 벗어나 다양한 디바이스로 구성된 유비쿼터스 네트워크를 위한 경량 암호 기술의 연구가 중심을 이룰 것으로 기대됨
- 생활 전반에 걸쳐서 다양한 성능의 각기 다른 디바이스들로 구성된 유비쿼터스 네트워크의 사용이 늘고 있으며, 각 응용 환경에 따라 다양한 제한 조건 및 안전성 요구 조건을 지니고 있음
- 각 응용 환경에 따른 여러 요구 조건을 만족시키기 위한 다양한 경량 암호 기술이 연구될 것으로 전망
- 유비쿼터스 네트워크를 통해 다양한 정보 서비스가 제공됨에 따라 비교적 높은 안전성을 요구하는 경량 암호 기술에 대한 연구 또한 필요함

○ 프라이버시 보호에 대한 관심과 중요성이 부각되면서, EU 및 미국 등 선진국을 중심으로 개인 정보 침해 방지를 위한 법률적, 제도적 정비와 더불어 이를 기술적으로 지원하기 위한 다양한 연구가 진행되고 있음

- Yao에 의해 1970년대 기본 개념이 탄생하였고 80년대와 90년대를 거쳐 컴퓨터 과학자와 데이터베이스 개발자 및 암호 학자에 의해 많은 연구 진행 중임
- 프라이버시 보호를 핵심 암호 기술인 다자간 비밀 계산의 중요성이 부각되면서, 2000년대에 들어서 데이터마이닝, 침입 탐지, 과학 계산, 기하학적 계산, 통계적 분석, 인터넷 경매 등의 다양한 분야에 대한 다자간 계산 프로토콜이 연구되고 있음
  - 현재 이상적인 안전성(ideal security)을 만족하면서 효율성을 높이려는 연구 방향과 랜덤화 기법 등을 이용하여 실용성을 확보한 후 안전성을 강화하려는 연구가 주를 이루고 있음
  - 이상적인 안전성을 보장하는 다자간 비밀 계산 프로토콜의 효율성 개선에 대한 연구가 활발히 수행되고 있으나, 아직 효율성 측면에 문제가 있어 실용화 단계에는 아직 이르고 있지 못한 상태임
  - 한편 랜덤화 기법 등을 이용하여 다자간 비밀 계산의 실용성을 높이는 연구가 진행되고 있으나, 완벽한 수준의 안전성을 보장하지 못하는 상태임
- 암호화된 데이터에 대한 검색, 연산, sorting 등의 프라이버시 보호 암호 데이터 처리 기술에 대한 연구가 활발히 진행되고 있으나, 안전성과 실용성을 동시에 만족하는 핵심 암호 기술 개발 부분에서는 아직 미진한 상태임
  - 순서 보존 암호화 기술은 평문의 순서 노출로 인해 선택 평문 공격에 안전하지 않으며, 결과적으로 아직까지는 학술적인 논의 단계에 있어 상용화 제품으로 적용하기에는 해결해야할 문제가 남아 있는 상태임
  - 암호화된 문서에서의 연산에 관한 연구는 주로 대학 및 연구 기관에서 이론 중심으로 이루어져 왔으며, 실제로 적용하기에는 효율성이 부족하고 가능한 연산도 한정적임

- EU 및 미국 등의 선진국은 프라이버시 보호에 대한 중요성을 인지하고, 프라이버시 보호 관련 다양한 프로젝트를 수행하고 있음
  - 미국의 경우, Stanford, Yale 등의 대학과 MS, Google, HP 등의 기업체 및 법률 및 의료 관련 기관이 연합하여 2003년부터 5년간 PORTIA(Privacy, Obligations, and Rights in Technologies Information of Assessment) 프로젝트를 통하여, 프라이버시를 보호하는 데이터마이닝 기법, 정보보호 정책 강화를 위한 도구 개발 등의 연구를 수행함
  - 유럽의 경우, 2004년부터 4년간 수행된 PRIME(Privacy and Identity Management for Europe) 프로젝트를 통해 프라이버시 보호 및 ID 관리 기술을 개발했으며, 현재 PRIME 프로젝트의 후속인 PrimeLife(Privacy and Identity Management in Europe for Life) 프로젝트를 통해 2008년부터 3년간 프라이버시에 대한 연구를 지속적으로 수행하고 있음
  - 또한 ECRYPT 프로젝트의 후속인 ECRYPT II(European Network of Excellence in Cryptology II) 프로젝트를 통해 2008년부터 4년간 프라이버시 보호 관련한 다양한 응용 프로토콜 연구를 수행하고 있음

○ 양자 암호는 1984년 Bennett과 Brassard에 의해 알려진 양자 키 분배 프로토콜의 무조건적 안전성이 증명된 이후 현재 상용화 단계에 이를 정도로 매우 급격한 성장을 이룬 반면, 인증, 서명 및 다자간 암호 프로토콜 등 양자 네트워크에서 수행될 수 있는 여러 양자 암호 프리미티브들의 경우 기술적 제약으로 인해 아직 이론적 접근에 머물고 있으며, 다양한 연구 결과물들이 도출되고 있지는 못함

- id Quantique, MagiQ Technologies, SmartQuantum, Quintessence Labs 등의 기업들에서 QKD 프로토콜의 상용화에 이미 성공하였으며, 전자 선거 및 은행 간의 데이터 전송에 실제 적용한 사례가 있고, 현재 140km 이상 거리에서 자유공간 및 광케이블을 통한 실험에 성공하고 있으며, 심지어

인공위성과의 통신에서 긍정적인 결과를 얻고 있음

- 유럽 연합의 SECOQC 프로젝트의 경우, 양자 네트워크에 대한 안전성 모델 정립 및 표준화를 바탕으로 양자 네트워크의 상용화를 목표로 하고 있음
- 양자 상태의 불확정성 그 자체를 양자 일방향 함수로서 인식하고 있으며, 이를 통해 다양한 암호 프로토콜의 무조건적 안전성을 보장하기 위한 노력이 지속되고 있음. 2001년 D. Gottesman과 I. L. Chuang이 세계 최초로 이러한 성질이 양자 서명에 적용 가능함을 보임
- 이후, quantum fingerprinting, quantum stabilizer coding 및 GHZ 양자 상태 등의 다양한 접근을 통해 양자 서명에 대한 새로운 연구 결과들이 발표되고 있으며, 최근 고전 메시지뿐만 아니라, 양자 메시지에 대한 서명이 가능하다는 것이 증명됨
- 양자 공개키 암호 시스템의 경우는 유니타리 변환을 통한 인코딩 방법을 이용한 기초적인 수준의 프로토콜이 제시되고 있지만, 프로토콜 전체의 구조적, 통합적 안전성은 아직 확보되지 못함. 아직 초보적 단계이지만 QPKI 구성 및 안전성 확보를 위한 여러 연구들이 진행되고 있음

2. 국내 기술현황

○ 미국 SHA-3 프로젝트와 더불어 국내에서는 고려대학교 등을 중심으로 새로운 hash 알고리즘 선정 작업에 동참하고 있음

- 국내에서 개발된 hash 알고리즘으로는 HAS-160, FORK-256, ARIRANG 등이 있으며, 높은 안전성과 효율성을 보유하고 있는 것으로 알려짐
- 기존에 개발된 hash 알고리즘들은 경량화 구현에 적합하지 않으며, 이에 따라 경량 hash 알고리즘 개발에 관한 연구가 세계적으로 활발해질 것으로 예상됨

○ 국내에서는 최근 u-City 사업 등을 통해 사회 전반에 유비쿼터스 네트워크

구성을 추진하고 있음. 유비쿼터스 네트워크의 필수 요소인 보안 기술에 대한 연구로는 RFID와 USN에 대한 연구가 주를 이루고 있으며 기타 유비쿼터스 디바이스에 적용 가능한 경량 암호 프로토콜에 대한 연구가 시급한 실정임

- 국내 경량 암호 기술 연구는 한국전자통신연구원 및 일부 대학을 중심으로 RFID와 USN 보안 기술이 주를 이루고 있음
- 암호학적인 기법을 바탕으로 한 경량 인증, 그룹 서명 및 키 분배 프로토콜에 대한 연구는 이론적인 연구들이 대부분이며, 일부 대학을 중심으로 연구가 진행되어 연구 결과를 국제 학회 및 저널에 발표하고 있음
- KISA에서 제안한 추적이 가능한 익명 인증 기술이 IETF 표준(RFC 5636)으로 채택되는 등 익명 인증 기술에 대한 연구가 이루어지고는 있으나 PKI를 이용하기 때문에 경량화된 디바이스를 사용하는 유비쿼터스 환경에 그대로 적용시키기 어려움
- 제한된 성능을 지닌 유비쿼터스 디바이스를 통한 정보의 교류가 사회 전반에 걸쳐 폭넓게 사용되면서, 높은 안전성을 지니고 동시에 효율성 및 다양한 환경에 적용 가능한 경량 암호 프로토콜에 대한 연구가 필요
  - 유비쿼터스 디바이스의 사용 범위가 넓어지고 다양한 정보 서비스가 제공되면서, 기존에 비해 높은 안전성을 지닌 경량 암호 프로토콜이 요구됨
  - 이론적인 연구를 벗어나 실용적인 효율성을 지닌 경량 인증, 경량 그룹 서명 프로토콜 및 경량 키 분배에 대한 연구가 필요하며, 또한 다양한 환경 및 요구 조건에 적합한 경량 암호 프로토콜에 대한 연구가 필요

○ 최근 빈번한 개인 프라이버시 정보 유출 사례 등을 통해 개인 프라이버시 정보 보호에 대한 사회적 요구가 늘고 있으며, 현재 국내에서는 연구소와 학교를 중심으로 프라이버시 보존형 데이터 처리 기술에 대한 일부 연구가 이루어지고 있음

- 최근 프라이버시 침해에 대한 사회적 우려가 높아져 프라이버시 보존형

데이터 처리 기술에 대한 관심이 증가하고 있으며, 이용자 자신의 선택에 따른 자율적 프라이버시 보호 기술을 개발하려 하고 있음

- 한국전자통신연구원을 중심으로 프라이버시 보호 ID 관리 기술, 암호 데이터 검색 기술, 프라이버시 보존형 데이터 마이닝 기술 등과 같은 일부 프라이버시 보존형 데이터 처리 기술에 대한 연구가 이루어지고 있음
- 고려대학교, 국민대학교, 인하대학교, 세종대학교 등을 중심으로 검색 가능 암호 시스템 및 PPDM 관련 연구가 진행되고 있으나, 단편적인 연구 결과에 머무르고 있음

○ 양자 암호 분야 국내 연구는 90년대 말부터 시작되었으며, 한국전자통신연구원, 고등과학원, KAIST를 중심으로 실제 구현을 위한 이론적 실험적 연구와, 서울대학교 및 고려대학교 등을 통한 수학적, 암호학적 연구가 개별적으로 이뤄지고 있음

- 양자 암호 관련 국내 연구는 양자 키 분배의 구현 및 상용화에 집중되어 있으며, 한국전자통신연구원을 중심으로 고등과학원과 서울대학교의 부분적 지원을 통해 이론적, 실험적인 연구가 체계적으로 이뤄지고 있음
- 양자 키 분배를 제외한 다른 양자 암호 프리미티브들에 대한 연구는 서울대학교와 고려대학교에서 개별적으로 연구가 진행되어 왔으나 선진국에 비해 미미한 수준에 그치고 있음
- 한국전자통신연구원은 양자 키 분배, 후처리, 인증, 비트 위임, 불확정 전송 등 다양한 양자 암호 프리미티브에 대한 이론적 연구를 수행하고 있음

3. 국내외 표준화 현황

○ 최근 선진 각국은 표준 암호 알고리즘을 정부 주도하에 개발하고 있음

- 미국의 표준 블록 암호 알고리즘인 DES를 대체하기 위해 NIST 주관으로

차세대 미국 표준 암호 제정을 위한 AES 프로젝트가 1998년에 시작되어 3년여의 평가 과정 끝에 2000년 10월 벨기에 루벤 대학에서 제안한 Rijndael을 차세대 미국 표준 암호로 선정하고, 2001년 11월에 미 연방 표준 FIPS-197로 제정하였음

- 유럽은 미국의 AES 프로젝트와 유사한 작업으로 NESSIE 프로젝트를 2000년 1월부터 2003년 3월까지 진행하였으며, 블록 암호, 스트림 암호, 공개키 암호, 디지털 서명, MAC, hash알고리즘 등 정보보호에 필요한 다양한 원천 기술에 관한 표준 제정을 목적으로 함
- 일본 자국 내에서는 유럽의 NESSIE 프로젝트와 유사한 CRYPTREC 프로젝트를 추진하여 2003년에 안전하고 효율적인 공개키 암호, 디지털 서명, 블록 암호, 스트림 암호, hash 함수, 난수 발생기 등을 추천하고 있음
- Hash 함수의 안전성에 문제가 제기된 이후, NIST는 2012년 말까지 신규 hash 알고리즘 SHA-3 개발을 목표로 “SHA-3 개발 프로젝트(Cryptographic Hash Project)"를 현재 진행 중에 있음

○ 디지털 서명과 관련된 국제 표준화 ISO/IEC JTC1의 SC 27에서 주도적으로 수행하고 있으며, 디지털 서명의 국제 표준은 어느 특정 서명 알고리즘을 국제 표준으로 채택하지는 않고 전반적인 프레임워크(framework)만을 규정함

○ 선진 각국은 프라이버시 보호 기술의 중요성을 인식하고, 관련 기술을 개발하고자 PORTIA, PRIME, PrimeLife, ECRYPT, ECRYPT II 등의 다양한 연구 프로젝트를 수행하고 있음

○ EU(SECOQC), 미국(ARDA)은 양자암호 네트워크 원천 기술 획득 및 상용화 추진을 위한 국가 중장기 로드맵을 마련하여 추진 중에 있음

제 4 절 연구 수행방법 및 보고서 체계

1. 연구 추진체계 및 수행방법

한국전자통신연구원은 전체 연구 방향 설정 및 기술 개발 총괄

- 한국전자통신연구원 주도로 프라이버시 강화 암호 핵심 기술 개발
- 고려대학교 산학협력단과의 공동 연구 추진
- 공동 연구 기관 및 관계 기관과의 유기적인 협력을 통한 프라이버시 강화 암호 기술의 발전 방향 및 목표에 대한 점검을 하여 세계적 수준의 연구 개발 수행
- 세계적인 우수 연구 기관과 협력하여 개발된 핵심 알고리즘에 대한 평가를 수행하며, 또한 선진 기술 습득 및 세계 최고 기술을 개발하는데 필요한 기반 기술을 단기간 내에 확보하도록 함

구분	참여기관별 역할 분담
ETRI 지식정보보안연구부	o 프라이버시 강화 암호 기술 개발 총괄 - 경량 암호 핵심 기술 개발 : 경량 hash 알고리즘, 경량 그룹 서명 및 인증 프로토콜 - 프라이버시 보존형 데이터 처리 기술 개발 : 순서 보존 암호화 기술, 암호 데이터 연산 기술 - 양자 암호 서명 프로토콜 개발
고려대학교 산학 협력단	o 경량 암호 기반 기술 연구
우수 국제 연구 기관	o 설계된 핵심 알고리즘에 대한 검증
위탁 연구 기관	o 안전성 증명 기술 연구 및 최신 연구 동향 파악

- 체계적인 연차별 기술 개발을 통해 연구 결과물이 최종 시스템에 유기적으로 결합될 수 있도록 연구개발 수행

- 개발된 프라이버시 강화 암호 기술의 산업화를 위해 연구 결과물의 실용적 이용 방안에 대한 면밀히 분석

○ 핵심 원천 기술 개발 및 지적 재산권 확보
- 프라이버시 강화 암호 기술에 대한 핵심 원천 기술 선점 및 확보
- 핵심 원천 기술 개발을 통한 다수의 우수 IPR 확보
- 국내의 보안 관련 업체와 관련 산업 분야에 대한 정보 공유 및 본 과제를 통해 개발된 핵심 원천 기술의 IPR 이전을 통한 교류의 활성화를 추진함

○ 국내외 동향 및 환경 변화를 파악하여 과제에 적극 반영
- 선진국의 비중 있는 정보보호 연구 기관과 협력을 통해 요구 사항과 기술 발전 상태를 면밀히 파악하여 이를 수용
- 관련 연구 성과의 산업화를 유도하기 위해 산·학·연 연구 네트워크를 통해 산업 현장에서 요구되는 특성들을 신속히 파악하고 공유함으로써 연구 개발에 능동적으로 반영하는 기회로 활용
- 정부 기관의 정책사항이 잘 반영되도록 사업 관리 협조 체제 구축

2. 보고서 체계

본 보고서는 2010.1.1 ~ 2010.12.31 기간에 수행된 “유비쿼터스 환경에서의 정보보호 서비스를 위한 프라이버시 강화 암호 기술 개발” 1차년도 사업의 최종 연구개발 내용 및 결과에 대하여 요약 작성하였다. 1장은 서론으로 연구 개발의 중요성, 연구내용 및 범위, 국내외 기술개발 동향을 소개한다. 2장과 3장은 경량 암호 기술 개발의 결과물인 경량 해쉬 함수와 경량 그룹 서명 프로토콜에 대한 연구 결과를 기술한다. 4장은 프라이버시 보존형 데이터 처리 기술 개발의 결과물인 순서 보존 암호화 기술에 대한 연구 결과를 설명한다. 마지막으로 5장에서는 본 사업의 결론을 맺는다.

제 2 장 경량 해쉬 함수 개발

제 2 장 경량 해쉬 함수 개발

제 1 절 경량 해쉬함수 연구 개발 필요성

1. 최근 해쉬함수 동향

암호학적 해쉬함수(Cryptographic Hash Function)는 인터넷 뱅킹, 전자서명, 메시지 인증 코드, 키교환 알고리즘, 키생성 알고리즘, 의사난수생성기 뿐 아니라 다양한 암호 시스템에서 사용되며, 이들의 안전성은 해쉬함수의 안전성에 기반을 둔다는 점에 있어서 안전한 해쉬함수를 설계하는 것은 매우 중요하다. 국제적으로 널리 사용되는 해쉬함수로는 90년대 초반에 제안된 MD5, SHA-1 등이 대표적이며 국내 표준 해쉬함수 HAS-160 역시 MD5, SHA-1의 설계 사상을 기초로 설계되었다. 국제 표준 블록암호인 AES의 경우 5년에 걸친 엄격한 공개 검증과정을 통해 선정된 반면, MD5, SHA-1 등의 해쉬함수들은 공개 검증과정 없이 채택되었고, 10년 이상 전 세계적으로 가장 널리 사용되어져 왔다. 그러나 2005년 중국의 Wang 교수 연구팀의 MD5, SHA-1 등의 해쉬함수에 대한 취약성 발견[SA09,WLF05,WY05,WYY05a, WYY05b,WYY05c,WYY05d]과 이를 기초로 HMAC-MD5, HMAC-SHA-1, APOP-MD5, X.509 인증서에 대한 충돌쌍 생성, Postscript 파일 충돌쌍 분석 등에 관한 결과가 연이어 발표됨에 따라, 엄격한 공개검증이 결여된 해쉬함수의 사용은 엄청난 사회적, 경제적 손실을 가져올 수 있음을 경고하고 있다[SA09,SLW09].

MD5, SHA-1 등의 해쉬함수의 취약성이 발표됨에 따라 미국 국립기술표준원(NIST, National Institute of Standards and Technologies)은 MD5를 사용하지 말도록 각 기관에 요청하였으며, 특히 SHA-1은 전자 서명 시 사용하지 말도록 권고하였다[NIS08]. 그리고 미국 국립기술표준원은 신규 해쉬 알고리즘 SHA-3 개발을 목표로 2007년부터 2012년까지 6개년 “SHA-3 개발 프로젝트(Cryptographic Hash

Project)" 계획을 수립하고 현재 진행 중에 있다[NIS10].

NIST의 6개년 프로젝트에서 보듯이 새로운 해쉬함수의 개발은 매우 필요한 상황이다. 국내에서는 TTA 표준 해쉬함수로 제안된 HAS-160[TTA00]만이 널리 사용되고 있는 실정이다. HAS-160은 MD5, SHA-1 등의 설계 논리와 유사하기 때문에 안전성을 장담할 수 없다. 또한 NIST는 특정 길이의 출력 크기를 갖는 해쉬함수가 아닌 가변 길이의 출력을 제공해야 한다는 요구 조건을 제시하고 있기 때문에 이러한 조건에 부합하는 신규 국제 또는 국내 표준 해쉬함수의 개발이 요구된다. 하지만 해쉬함수에 대한 설계 이론 및 분석 이론에 대한 명확한 이해 없이는 안전한 해쉬함수를 설계하는 것이 어려울뿐더러 안전성을 보장할 수도 없기 때문에, 무엇보다 먼저 안전성이 증명된 설계 이론에 대한 연구가 요구된다. 또한 개발될 해쉬함수는 다양한 분야에서 사용되기 때문에 해쉬함수에 요구되는 성질들에 대한 명확한 규명 역시 절실한 상황이다.

대부분의 해쉬함수 결과들을 보면 단편적이고 지엽적인 연구 결과에 초점을 맞추고 있다. 또한 엄밀한 설계 이론, 분석 이론 위에 알고리즘을 설계하기 보다는 아직도 경험에 기초하여 설계하고 있는 현실이다. 2년 뒤에 선정될 국제 표준 해쉬함수는 기간이 말해주듯 다양한 설계 이론과 분석 이론을 기초로 다듬어져서 최종적으로 엄격한 공개 검증과정을 통하여 선정되어야 한다. 따라서 국제 표준 해쉬함수 개발은 단기간에 이루어질 수 없으며 설계 이론, 분석 기술, 안전성 증명 기술, 구현 논리 기술 등의 연구가 함께 병행되어야 하는 종합적인 연구 과제임을 알 수 있다.

대칭키 암호와 관련된 세계적인 학술대회인 FSE가 지난 2010년 2월 8~10일에 서울에서 개최되었다. 총 21편의 논문이 발표되었으며, 이 중 13편이 해쉬함수에 관한 논문이었는데, 논문 편수만 보더라도 해쉬함수가 세계적으로 큰 이슈가 되고 있음을 보여준다. 해쉬함수에 관련된 13편의 논문을 살펴보면, 기 제안된 해쉬함수 분석 논문이 7편, 해쉬함수에 대한 이론 논문 3편, 메시지 인증 코드 논문 3편이 발표되었다. 이처럼 해쉬함수의 연구 결과는 전반적으로 기존 해쉬함수 분석에 초점이 맞추어지는 경향이 있음을 알 수 있다. 반면 설계 이론의 바탕이 되는 신

규 구조에 대한 연구는 열악한 상황이다. 해쉬함수에 대한 신규 설계 논리에 대한 연구의 요구가 점점 커지고 있는 반면, 실제적으로는 이에 대한 연구 성과가 가시적으로 나타나고 있지 않다.

2. 경량 해쉬함수의 필요성

머지않은 미래에 유비쿼터스 컴퓨팅을 기반으로 일상생활의 사물들, 어플라이언스, 상품들, 기업의 생산, 물류, 판매 고객 관리 등의 비즈니스 프로세스를 구성하는 기기나 시스템들이 모두 지능화되고 네트워크로 연결됨으로써 매우 다양한 새로운 비즈니스를 출현시킬 것이다. 이러한 유비쿼터스 비즈니스는 단순한 상거래뿐만 아니라 일반적인 기업 경영, 공급망 관리, 고객 관계 관리, 자산 관리, 현장 인력 관리, 지식 관리, 유통 관리, 안전 관리 등 거의 모든 비즈니스 활동에 혁신적으로 적용될 수 있어 이와 관련된 기술과 상품이 미래 IT 시장을 주도할 것이다. 이 모든 유비쿼터스 컴퓨팅 혁명의 단초가 되는 핵심기술이 RFID(Radio Frequency IDentification)이다.

유비쿼터스 컴퓨팅의 핵심은 스마트한 상황 인식과 장소에 구애받지 않고 컴퓨터 네트워킹을 가능케 하는 무선 기술이다. 따라서 유비쿼터스 칩이란 상황 및 환경을 인식, 감지하고 무선을 통해 네트워크에 연결하는 무선 인식(RFID) 및 무선 센서(Wireless Sensor) 칩을 말한다.

RFID 태그는 그 용도와 기능, 크기 등에 따라 매우 다양하지만 가장 널리 사용될 것으로 예상되는 것은 저가 수동형 태그로 향후 바코드 시스템을 대체할 것으로 기대된다. 바코드의 경우 물품의 종류만을 기록하며 빛을 이용하여 인식하는 장치로 코드 체계만 표준화되어 있으면 매우 간단하게 구현 가능하며, 보안이나 프라이버시의 문제는 고려될 필요가 없다. 한편, RFID 태그는 각 물품 고유의 코드를 가지게 되어 물품의 종류, 생산 이력 등이 저장되고 활용된다. 또한, 인식 범위가 바코드 시스템보다 넓어 물류 관리에 편리하게 활용되고 있다. 하지만, 이러한 장점은 보안과 프라이버시 보호의 측면에서는 새로운 역기능을 유발한다. 개

인이 보유한 물품의 정보가 리더기를 통하여 유출될 수 있으며, 태그의 정보를 복사하여 위조될 우려가 있다. 따라서 유비쿼터스 환경에서는 정보화 역기능의 문제가 더욱 심각한 사회 문제로 대두될 것이며 사이버 공격으로 인한 피해는 규모와 확산 속도 면에서 그 파급 효과가 지금보다 훨씬 심각할 것으로 예상된다. 이에 따라 정보통신부를 중심으로 안전한 u-Korea 구현을 위한 중장기 정보보호 로드맵을 작성하여 예상되는 역기능에 대한 대비책을 강구하고 있다.

현재까지 발표된 대부분의 정보보호 기법들은 안전성을 높이기 위하여 복잡한 암호 연산을 사용하거나 통신량이 많이 소모되는 제안들이다. 따라서 학술적인 연구로만 가치 부여가 가능한 기술들이 주를 이루고 있다고 볼 수 있다. 한편, 해쉬함수 하나만 구현할 수 있으면 인증을 제공할 수 있을 것으로 기대되어 해쉬함수 기반 인증 기법에 대한 연구가 활발히 진행되었으나, 실제로 수동형 RFID 태그에 탑재 가능한 해쉬함수는 아직까지 발표된 바 없다. 따라서 현재까지 발표된 정보보호 방식은 RFID 시스템에 적용하기에는 실용적인 문제를 많이 지니고 있는 실정이다.

초경량 해쉬함수의 개발이 가능하면 해쉬함수 기반 정보보호 방식의 구현이 가능하여 활용될 수 있음에도 불구하고 해쉬함수 설계 기술의 진보는 매우 느리다. 최근 NIST의 SHA-3 프로젝트의 일환으로 해쉬 워크숍이 개최되었으며, 해쉬 컨퍼런스에서 해쉬함수의 정의, 설계 요구 조건, 압축함수의 설계 기법 등에 대하여 심도 있는 재검토가 필요하다는 공감대가 형성되었으며 장기적으로 용도에 따라 다양한 해쉬함수를 개발해야 한다는 쪽으로 논의가 진행되었다. 이 워크숍에서 구체적으로 언급되지는 않았으나 향후 RFID용 해쉬함수의 설계도 자연스럽게 전개될 것으로 예상된다. 따라서 이러한 연구 흐름에 대응하기 위하여 국내에서도 RFID를 비롯한 다양한 환경에 적합한 해쉬함수의 개발에 더욱 더 많은 노력이 기울여야 한다.

제 2 절 경량 해쉬함수 요구 사항

본 장에서는 경량 해쉬함수가 만족해야 할 요구 사항을 안전성과 효율성 측면에서 각각 살펴본다.

1. 안전성 요구 사항

암호학적 해쉬함수 H는 임의의 길이의 스트링을 입력으로 받아 고정된 길이의 스트링을 출력하는 압축 함수로, 해쉬함수를 사용하는 암호 시스템의 안전성뿐만 아니라 효율성을 향상시키는데 사용 목적이 있다. 해쉬함수가 암호 시스템에서 이용될 경우 암호 시스템의 안전성을 유지시키기 위해서는 다음의 세 가지 성질을 기본적으로 만족하여야 한다.

- **역상 저항성(Preimage resistance):** y가 주어졌을 때, $H(x)=y$인 x를 얻는 것이 계산상 어렵다.
- **제 2 역상 저항성(Second preimage resistance):** $H(x)=y$인 x, y가 주어졌을 때, $H(x')=y$인 다른 x'를 얻는 것이 계산상 어렵다.
- **충돌 저항성(Collision resistance):** $H(x)=H(x')$인 서로 다른 x, x'를 얻는 것이 계산상 어렵다.

세 번째 성질을 만족하는 해쉬함수 H는 두 번째 성질도 만족하므로, 해쉬함수 H가 지녀야할 성질은 다음과 같이 줄어든다.

- **역상 저항성(Preimage resistance):** y가 주어졌을 때, $H(x)=y$인 x를 얻는 것이 계산상 어렵다.
- **충돌 저항성(Collision resistance):** $H(x)=H(x')$인 서로 다른 x, x'를 얻는 것이 계산상 어렵다.

1990년에 MD4가 개발된 이후 비슷한 형태의 전용 해쉬 알고리즘인 MD5, RIPEMD, HAVAL, RIPEMD-128, 160, SHA-0, SHA-1, SHA-224, 256, 384, 512, PKC'98에서 제안된 해쉬함수, HAS-160이 개발되었으나, 해쉬함수의 안전성 분석 이론의 발전은 더딘 편이었다. 1996년 Dobbertin은 블록암호의 차분 공격과 대수적 방정식을 이용하여 MD4에 대한 2^{20}의 복잡도를 갖는 전체 라운드 충돌쌍 공격을 처음으로 소개하였으며[Dob98a], 1998년 2^{32}의 복잡도를 갖는 2-라운드 역상 공격을 소개하였다[Dob98b]. 1993년 Boer와 Bosselaers는 MD5의 최상위 비트의 확산 성질이 좋지 않음을 이용하여 2^{16}의 복잡도를 갖는 의사충돌쌍 공격을 소개하였다[BB94]. 여기서, 의사충돌쌍 공격은 서로 다른 두 개의 초기치에 동일한 메시지를 적용하여 충돌쌍을 찾는 공격을 의미한다. 1996년 Dobbertin은 선택 초기값에 서로 다른 메시지를 적용한 의사충돌쌍 공격을 소개하였다[Dob96]. 이 공격은 그 당시 하나의 개인 컴퓨터를 이용하여 약 10시간 만에 충돌쌍을 찾을 수 있는 매우 강력한 공격이었다. 하지만 나머지 해쉬함수 MD5, RIPEMD, RIPEMD-128, 160는 기존의 안전성 분석으로 분석이 되지 않아, 안전할 것으로 여겨져 왔다. 특히 HAS-160의 경우 기존의 분석 이론으로는 1 라운드조차도 분석을 할 수 없었다.

한편 1997, 1998년 Wang, Chabaud, Joux는 SHA-0에 대한 분석에 성공하였다[Jou04,Wan97,Wan98]. SHA-0는 다른 해쉬함수와는 달리 메시지 확장 과정이 고정된 메시지 순서를 사용하지 않고 LFSR과 같이 메시지를 확장해서 사용한다는 특징을 갖고 있다. 하지만 SHA-0에 사용된 분석 이론만으로는 SHA-1을 분석하는 것은 불가능하다. 따라서 SHA-1은 안전할 것으로 여겨져 왔으며, MD5와 SHA-1은 전 세계적으로 가장 많이 쓰이는 해쉬함수로 자리 잡아 왔다.

2004년 Crypto Rump session에서 중국의 산동대학의 Xiaoyun Wang 교수는 MD4, MD5, RIPEMD, HAVAL-128, SHA-0에 대한 분석 결과를 발표하였다. MD4, MD5, RIPEMD의 경우는 2005년 Eurocrypt에서 자세히 다루어졌으며, SHA-0와 SHA-1에 대한 분석 결과는 2005년 Crypto에서 발표하였다[WLF05,WY05,WYY05a,WYY05b,WYY05c, WYY05d]. Wang의 분석 방법이 주목을 받는 이유를 정리하면 다음과 같다.

○ MD 계열 해쉬함수를 분석하는 일반적인 분석기법을 처음으로 제시하였다.
○ MD 계열 해쉬함수의 안전성을 평가하는 하나의 평가 도구의 틀을 마련하였다.

Wang의 분석 이후 Wang의 분석 결과는 더욱 향상되고 강력해 졌으며, APOP, HMAC, NMAC, 인증서와 같이 단지 해쉬함수 자체에 대한 공격이 아닌 다양한 응용 환경에 대한 분석 결과도 제시되었다. 이는 안전하지 않은 해쉬함수가 암호 시스템에 사용될 경우, 해쉬함수의 취약점에 의하여 암호 시스템이 공격될 수 있는 단적인 예라 할 수 있다. 따라서 새로운 경량 해쉬함수를 개발하기 위해서는 역상 저항성, 충돌 저항성 등과 같은 해쉬함수의 기본적인 공격 기법뿐만 아니라, 기제안된 다양한 해쉬함수 공격 기법에 대해 충분한 안전성을 갖는 경량 해쉬함수를 설계해야 한다.

2. 효율성 요구 사항

저가격의 RFID 태그의 수동 태그에 내장되는 암호 모듈의 구현 환경을 가격(cost), 면적, 게이트 수, 연산 성능, 전력 소비 측면에서 분석한다 [Dob98a, JBC05,BR00,DC98].

가. 공정 기술 발달

RFID 칩을 제작하는 반도체 공정 기술은 계속 향상되고 있다. 칩 기술을 정의하는 파라미터는 feature size 혹은 minimum geometry이다. 이러한 파라미터는 가장 작은 선폭, 선 폭간의 최소 간격 등을 정의한다.

[표 2-2] 반도체 feature size 전망

년도	1995	1998	2000	2002	2003	2004	2006	2007	2009
Feature Size	.35	.25	.18	.13	.10	.09	.07	.065	.05

[표 2-1]은 인텔사의 공정 기술을 기준으로 한 feature size의 연도별 예측 값이다. feature size가 작아짐에 따라 동일한 칩 면적에 더 많은 게이트 수를 넣을 수 있어서 RFID 태그 내 암호 모듈에 할당되는 게이트 수에 영향을 주며 [표 2-2]와 같다.

[표 2-3] 반도체 feature size에 따른 mm^2당 게이트 수 및 제조 가격

Feature Size	.8	.5	.35	.25	.18
Gates/mm^2	1,500	4,000	10,000	38,000	60,000
Cost /mm^2	2.5	3	4	6	8

나. 가격

현재 수동(passive) 태그가 경제성을 인정받기 위해서는 5 cent 이하가 되어야 한다고 평가하고 있다. 5 cent 태그를 만들려면 IC 가격은 2 cent를 넘지 말아야 한다.

다. 면적과 게이트 수

RFID 태그가 소규모 물체에 내장되어야 하므로 태그의 크기는 가장 작은 물품에도 내장될 수 있어야 한다. 칩의 크기에 대한 정확한 표준안은 정의되어 있지 않지만, 태그 칩의 최대 크기는 $1mm^2$로 일반적으로 받아들여지고 있다. 현재 태그 전체 면적 $1mm^2$ 중에 RFID 암호용으로 할당 가능한 크기는 약

$0.16mm^2 \sim 0.25mm^2$로 평가되고 있다. 반도체 고정 기술에 따라 정해진 면적에 담길 수 있는 게이트 수는 가변적이 된다. 현재 $0.35\mu m$ 공정 기술을 고려할 때 태그에 내장 가능한 게이트 수는 약 10,000~20,000 게이트 정도로 판단되고 있다. 따라서 암호 모듈에 할당 가능한 크기는 약 2,000~5,000 게이트로 평가된다.

라. 전송되는 데이터의 크기와 암호 연산 성능

RFID 태그 내 암호 시스템의 성능은 다음과 같은 기준에 의해 결정된다.

- RFID 응용 시스템에서 요구하는 응답 시간
  - 1초에 50 ~ 100개의 태그 처리(50 ~ 100/sec)
- request-response 프로토콜에 정의된 response 시간
  - 13.56MHz RFID의 경우 : 320us 이내
- UHF대 RFID의 경우 주파수 호핑에 따른 응답 시간 제약
  - 900MHz대 RFID의 응답시간 : 400ms 이내

13.56MHz와 900MHz 밴드 RFID 태그에서 이용 가능한 전송률은 다음과 같다. 13.56MHz의 경우 50개의 태그를 처리하는데 필요한 전송률이 약 26$Kbps$이며, 900MHz의 경우 300개의 태그를 처리하는데 필요한 전송률은 약 128$Kbps$로 평가되고 있다. 태그와 리더간의 연결 시간이 제한되는데, 이 값이 1초라 가정할 경우 각 태그는 약 500 비트를 전송할 수 있다 [JBC05,DC98].

응용 시스템의 요구 사항에 따라 1초에 100개의 태그를 처리하는 경우, 각 태그의 암호 모듈에서 처리해야 하는 성능을 계산하면 다음과 같다. 이 경우 각 태그에 할당되는 연산 시간은 $1/100 = 10ms$가 된다. 단 태그 연산중에는 프로토콜 처리와 암호 연산으로 나뉘므로, 프로토콜 처리 시간이 약 1/3을 차지한다고 가정할 경우 가 태그 내 암호 모듈에 실제 할당되는 약 $6.7ms$(6,700us)가 된다. 이 경우 RFID 태그 내에서 100KHz 주파수를 사용할 경우 하나의 clock 주기(T)가 10us이므로 약 670개의 clock이 할당가능하다. 반면 RFID 태그에서 1MHz의

clock 주파수를 사용할 경우 clock 주기(T)는 $1u$이므로 6,700개의 clock이 할당 가능하다.

오스트리아 Graz 대학의 연구 결과에 따라 $13.56MHz$ RFID에서 response 시간은 약 $320us$로 정의되었다. 이 경우 $13.56MHz$의 캐리어 주파수에서 도출한 내부 태그 clock 주파수가 $100KHz$인 경우 허용 가능한 clock 사이클 수가 32 clock 이하로 제한되는 것으로 평가되고 있다. 이러한 문제를 해결하기 위해 Graz 대학의 2-way request-response 프로토콜의 경우 interleaved challenge-response protocol을 사용하여 $100KHz$ clock 주파수를 사용하는 수동 태그에서 1,000 clock의 할당 시간을 갖도록 암호 모듈을 설계하는 것이 가능하도록 개선하였다 [FDW04].

마지막으로 $900MHz$대 UHF RFID 태그의 경우 주파수 호핑에 따른 리더와 태그 간의 할당 시간이 $400ms$로 제한된다[REC04]. 이때 프로토콜 처리에 따른 오버헤드를 무시하고 $1MHz$ clock을 사용할 경우, 암호 모듈에 할당될 수 있는 clock 수는 400,000개가 된다. 이러한 clock 수는 1개의 태그 기준이므로 여러 개의 태그를 $400ms$ 내에 처리하는 기준으로 바꿀 경우 태그에 할당되는 clock 사이클 수는 감소될 수 있다. 위에서 기술한 3가지 조건은 서로 연관되므로 가장 작은 값을 clock 사이클 수도 상한으로 두고 암호 모듈을 설계해야 한다.

결론적으로 암호 모듈에 할당되는 clock 수 조건이 매우 작은 경우는 적절한 성능의 암호 하드웨어와 Graz 대학에서 제안한 Interleaved Request- Response protocol 구현이 바람직할 것으로 판단되다. 반면 암호 모듈에 할당되는 clock 수가 클 경우는 암호 하드웨어 구현 시 면적을 최소화하고 전력을 최적화하는 설계가 필요하다.

현재 캐리어 주파수 f_c가 $915MHz$ 혹은 $13.56MHz$인 RFID 시스템에서 태그 내 주파수의 최대 한계가 정의되어 있다. 클래스 0와 클래스 1의 경우 가능한 가장 느린 clock을 정의하고 있다. $f_c = 13.56MHz$로 동작하는 클래스 1 태그의 경우, 사용 가능한 clock 주파수는 $f_c / 32 = 423.75KHz$이하이다. 따라서 clock 주파수,

반응 시간 등을 고려한 암호 모듈 설계가 필요하다.

마. 전력 소비

RFID 태그의 경우 전원을 리더에서 유도하므로, 전력 소비를 최소화하는 것이 필수적이다. 현재 RFID 태그에서 소비 가능한 최대 전력량은 약 $150uW$ 이하인 것으로 평가되고 있다[Dob98a]. RFID 태그의 면적 상한이 $1mm^2$이고 암호 모듈이 차지하는 면적이 $0.25mm^2$인 점을 고려할 때, 암호 모듈에서 소비될 수 있는 전력양은 약 $35uW$ 이하인 것으로 판단된다. 일부 문헌의 경우, 전력 소비가 아닌 전류 소비를 기준으로 태그별로 $10uA$ 이하인 것으로 평가하는 경우가 있다[FDW04]. 이러한 저전력 소비 조건을 만족시키기 위해 저전력 회로 설계 연구가 필요하다.

[표 2-4] RFID 수동 태그의 암호학적 구현 환경

항목	구현 환경
가격	5 cent 이하
태그 전체 면적	$< 1mm^2$
암호 모듈 할당 면적	$< 0.25mm^2$
태그 전체 게이트 수	10,000 ~ 20,000 게이트
암호 모듈 할당 게이트 수	2,000 ~ 5,000 게이트
주파수 대역	$13.56MHz$, $900MHz$
전송률	$13.56MHz$: $26Kbps/50Tags$ $900MHz$: $128Kbps/200Tags$
시스템 측면의 응답 성능 (response performance)	low end : $50 \sim 100Tags/$sec $(10ms \sim 20ms/Tag)$ high end : $100 \sim 300Tags/$sec $(3.3ms \sim 10ms/Tag)$
$13.56MHz$ RFID 응답시간	$320us$

(암호 모듈 할당 clock 수)	(32 clocks@100KHz)
900MHz RFID 응답시간 (암호 모듈 할당 clock 수)	400ms(주파수 호핑 간격) (400,000 clocks@1MHz)
태그 전체 전력 소비 양	$<$ 150uW
암호 모듈의 전력 소비량	$<$ 34uW
암호 모듈의 전류 소비량	$<$ 10uA

바. RFID 태그 내 암호 모듈 구현 환경

[표 2-3]은 앞서 기술한 RFID 태그에 대한 암호 모듈 구현 환경을 정리한 결과이다. 따라서 RFID 환경에 적합하도록 경량 해쉬함수를 설계하려면 [표 2-3]의 결과를 만족해야만 한다. 이러한 결과는 공정 기술 발전에 따라 변동이 가능하겠지만, 수동 RFID 암호 모듈을 설계하는 참고 자료로 활용될 수 있을 것으로 판단된다.

제 3 절 경량 해쉬함수 개발안

1. 경량 해쉬함수 설계 논리

수동형 RFID 태그와 같이 리소스가 제약된 환경에 암호 기술을 적용하기 위해서는 2장에서 살펴본 바와 같이 태그에서 허용하는 구현 면적, 전력 소비량 등을 고려해야 한다. 이를 위해서는 암호 기술을 하드웨어로 구현하는데 따르는 구성 논리와 그들의 구현 면적, 전력 소비량 등에 대한 이해가 필요하다. 기본적으로 하드웨어는 데이터를 저장하지 않고 산술적인 연산 대부분을 수행하는 조합 논리(Combinational Logic)와 데이터를 저장하고 이어지는 clock에서 이전 결과 값을 이용하여 다시 결과를 계산하는 기능을 수행하는 비조합 논리(Non-combinational Logic)의 두 가지 종류로 구성이 된다.

가. 조합 논리(Combinational Logic)

조합 논리란 디지털 회로의 가장 기본이 되는 구성 요소로서, 회로의 출력이 현재의 입력 값으로부터만 영향을 받는 회로를 말하며 기본 게이트(Gate)에 해당하는 NAND, NOR 등이 조합되어 표현된다. 대부분의 수학적인 연산을 하는 부분은 모두 이 조합 논리 하드웨어로 구성된다. 여기에서 우리가 설계된 하드웨어의 구현 면적을 말할 때 흔히 '게이트 수' 혹은 '하드웨어 사이즈'라는 용어를 사용하게 되는데, 이때 '게이트 수'는 공정에 따라 약간의 차이가 있겠지만 기본적으로는 2개의 입력과 1개의 출력을 갖는 NAND 게이트의 수를 의미한다. 즉, NAND 게이트의 크기를 '1'로 기준 하였을 때 나머지 다른 종류 게이트의 크기를 비례 환산하여 표시한 것을 의미한다.

[표 2-4] 논리소자들과 NAND를 기준으로 한 각 논리소자의 비례 게이트 수

논리	AND	NAND	OR	NOR	XOR
게이트 수	1.67	1	1.67	1	3
논리	NXOR	INV	Full ADD	2-to-1 MUX	4-to-1 MUX
게이트 수	3	0.67	7.67	3	7.33

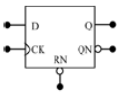
[표 2-4]은 삼성전자의 $0.13\mu m$ 라이브러리를 이용하여 디자인된 것을 참조하여 기본 NAND 게이트 대비 다양한 논리 게이트의 하드웨어 구현면적을 나타낸 것이다. 이는 제조 공정에 따라 값이 달라질 수 있다. [표 2-4]에서 보는 바와 같이 산술 덧셈기(Full Adder) 같은 경우나 조금 복잡한 Multiplexer의 경우는 많은 하드웨어 면적을 차지함을 알 수 있다. 따라서 하드웨어 구현 면적을 줄이기 위해서는 가능한 한 복잡한 계산식을 단순화해야 하며, 경우에 따라 여러 개의 결과 값 중 몇 개를 선택해야 하는 연산의 수를 줄여야 한다.

나. 비조합 논리(Non-combinational Logic)

비조합 논리란 순차 논리(Sequential Logic)라고도 불리는데, 이 회로는 출력이 현재의 입력뿐만 아니라 과거의 입력에도 영향을 받는 회로에 해당한다. 순차 논리는 clock 신호에 의해 입력을 샘플링 하여 출력을 내는 Flip-Flops와 clock 신호와 관계없이 계속해서 입력을 샘플링 하여 출력을 내는 Latches 등이 있다.

일반적으로 암호 알고리즘에서 비밀키나 혹은 평문, 암호문 같은 일련의 데이터를 저장하는 데에는 주로 Flip-Flops가 사용되는데, 통상 "레지스터(Register)"라고 부른다. 여러 가지 종류의 Flip-Flop들이 존재하지만 그 중에 가장 일반적으로 많이 사용되는 D Flip-Flop을 [표 2-5]에서 보여 준다.

[표 2-5] D Flip-Flops와 비례 게이트 수

논리	논리 심벌	게이트 수
D Flip-Flop with Reset		7.67

암호화 또는 복호화 연산을 수행하기 위해서 초기 입력 데이터를 저장할 레지스터, 매 라운드마다 라운드 함수를 수행하는데 필요한 레지스터, 라운드 함수를 수행하고 난 결과를 저장할 임시 레지스터 등이 존재한다. 물론 암호 알고리즘의 성격상 공유 가능한 레지스터가 존재할 수도 있지만 반드시 따로 구성되어야 하는 경우도 있다. [표 2-5]와 같이 1 비트를 저장하는 데에는 7.67 게이트가 필요하다. 그러므로 128 비트만을 저장한다고 가정하여도 약 1,000 게이트가 요구된다. 이는 조합 논리에 비해서는 매우 큰 값이다. 따라서 암호 알고리즘의 하드웨어 구현상 필요한 레지스터의 수가 얼마인가 하는 것이 전체 하드웨어 사이즈에 많은 영향을 준다는 것을 알 수 있다.

다. 저전력 하드웨어

하드웨어 구현에서는 목적하는 바에 따라 고성능 또는 저전력 구현으로 구분해서 접근할 수 있다. 절충되는 부분에 속하는 경우도 있으나, 일반적으로는 고성능 하드웨어 구현과 저전력 하드웨어 구현은 서로 Trade-off 관계에 있다. 이 두 가지 측면의 관점에서 추가적으로 개입되는 것이 구현 면적인데, 고성능 하드웨어 구현 혹은 저전력 하드웨어 구현에 구현 면적까지 작다면 최상의 하드웨어라 할 수 있다. RFID 태그에서는 제약된 구현 면적이 중요 요소로 대두되고 있으므로, 유사 라운드를 반복 적용하는 형태의 일반적인 암호 알고리즘에서의 구현 면적에 대해 알아보자. 속도 측면의 고성능 하드웨어를 구현하기 위해서는 기본적으로 라운드 당 1 clock 사이클을 소요하는 것으로 가정한다. 예를 들어, 128-비트 비밀키의 AES를 하드웨어로 구현하고 암호화를 수행하는데, 주어진 제조 공정에서 Critical Path의 연산시간이 $40ns$ 소요되었다고 하자. 1 라운드를 수행하는데, 1 clock이 소요되는 아키텍처에서는 최대 주파수가 $25MHz$ 이상이 될 수 없으므로 최대 Throughput이 $320Mbps$ 정도 된다. 그러나 1 clock에 여러 라운드를 수행하는 구조로 설계한다면, 그 만큼 Critical Path가 길어져 처리 clock 수는 현저히 줄어들지만 최대 주파수는 낮아지게 때문에 최종 Throughput은 개선되지 않는다. 각 구조에 따른 성능변화를 [표 2-6]에 나타내었다.

[표 2-6] 다양한 하드웨어 구조에 따른 성능 변화

clock 수/라운드	Critical Path (clock 주파수)	Throughput	하드웨어 사이즈
1 clock/4 R	$160ns(6.25MHz)$	$320Mbps$	$A0+4A1$
1 clock/2 R	$80ns(12.5MHz)$	$320Mbps$	$A0+2A1$
1 clock/1 R	$40ns(25MHz)$	$320Mbps$	$A0+A1$
2 clock/1 R	$20ns(50MHz)$	$320Mbps$	$A0+1/2\ A1$
4 clock/1 R	$10ns(100MHz)$	$320Mbps$	$A0+1/4\ A1$

[표 2-6]에서 $A0$는 라운드 당 clock 수에 영향을 받지 않는 하드웨어 면적을, $A1$은 라운드 당 clock 수에 영향을 받는 하드웨어 면적을 나타낸 것이다. 위의 표에서 알 수 있듯이 한 clock에 여러 라운드를 연속적으로 수행하는 구조는 $A1$의 크기가 라운드 수의 배수만큼 증가하게 된다. 반면, 한 라운드를 여러 clock으로 나누어 실행하는 구조는 $A1$의 크기가 라운드 수의 역수만큼 감소하게 되고, 또한 하드웨어를 분할하여 사용할 수 있기 때문에 전체적인 하드웨어 구현 면적을 줄일 수 있다.

저전력 하드웨어 구현에 대해서는 하드웨어의 소비 전력(Power)에 대한 이해가 필요하다. CMOS 회로에서 전력 소비는 Static Dissipation(P_s)과 Dynamic Dissipation(P_d)의 두 가지 성분의 합으로 정의할 수 있다. Static Dissipation은 CMOS 자체의 성질에 의해 소비되는 누출(leakage) 전류에 의한 것이며, Dynamic Dissipation은 Switching Transient Current와 Load Capacitance의 충·방전 전류의 합으로 표현할 수 있다. 위에 언급한 두 가지 종류의 전력 소비 중에서 큰 비중을 차지하는 것은 Dynamic Dissipation이다. P_d는 다음과 같은 수식으로 표현이 될 수 있다.

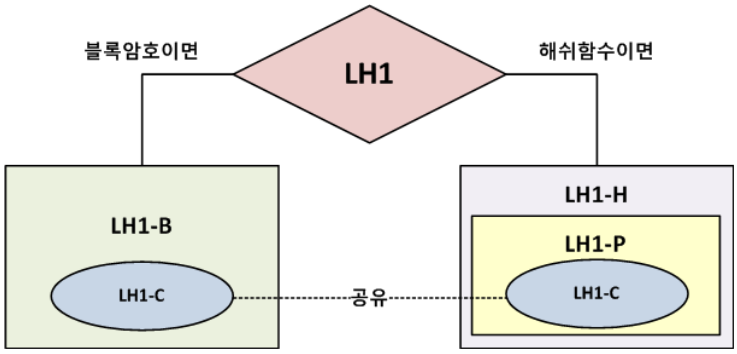
$$P_d = C_L \cdot V_{DD}^2 \cdot f_p$$

위의 수식에서 C_L은 회로가 구동해야 하는 Load Capacitance에 해당하고, V_{DD}는 회로에 걸리는 전압을 말하며 f_p는 회로가 Switching하는 Toggling Rate에 해당한다. 따라서 전체 Dynamic Dissipation의 양을 줄이기 위해서는 회로가 구동해야 하는 Load Capacitance를 줄이고 즉, 다시 말하면 회로의 수와 사이즈를 줄이고 전압을 낮추며 Switching이 최소가 되도록 설계하여야 한다.

위의 결과로 보아 저전력은 하드웨어 사이즈와 clock 주파수와 굉장히 밀접한 관계가 있음을 알 수 있다. 따라서 저전력으로 설계하기 위해서는 최대한 clock 주파수를 낮추고 꼭 필요한 회로만 추가하며 쓸데없는 회로의 switching을 없애는 등의 작은 면적을 차지하는 설계를 하는 것이 관건이다.

2. LH1 알고리즘 소개

LH1은 RFID/USN 등과 같은 자원이 제한된 하드웨어 구현 환경에 적합하도록 설계되었으며, 하나의 모듈을 이용하여 무결성을 위한 해쉬함수와 기밀성을 위한 블록암호를 모두 구성할 수 있다. 이는 제약된 구현 환경에서 두 가지 기능을 모두 제공할 수 있기 때문에 별도의 서로 다른 프리머티브가 구현될 필요가 없으므로 RFID/USN과 같은 유비쿼터스 환경에서 폭넓고 실용적으로 이용될 수 있을 것으로 기대된다. 기제안된 경량 해쉬함수와 경량 블록암호와 비교하였을 때, LH1은 구현 면적, 연산 성능 등의 하드웨어 구현 효율성이 매우 뛰어나도록 설계되었다. 해쉬함수와 블록암호를 모두 구성할 수 있는 LH1의 계층 구조는 [그림 2-1]과 같이 핵심 연산을 공유하면서 선택에 따라 해쉬함수로 사용될 것인지 블록암호로 사용될 것인지를 결정할 수 있다.



[그림 2-2] LH1의 계층 구조

LH1은 크게 해쉬함수 LH1-H와 블록암호 LH1-B로 구성된다. LH1-P는 해쉬함수에서 사용되는 permutation이며 핵심 단계 연산인 LH1-C를 블록암호 LH1-B와 공유하여 사용한다.

가. 비트열 - 워드 간의 변환

LH1의 내부 연산은 32-비트 정수간의 연산이 사용된다. 즉, 임의의 길이의 비트열을 32-비트 워드열로 변환하여 처리하고 그 결과를 32~224-비트 비트열로 변환하여 해쉬값으로 출력하게 된다. 따라서 비트열 - 워드열 간의 변환 규칙이 필요하며, LH1은 다음과 같은 변환 규칙(big-endian convention)을 따른다.

1) 32-비트 비트열 - 워드 간의 변환

32-비트 비트열을 4-바이트 문자열로 보고 첫 바이트가 워드의 최상위 바이트가 된다. 예를 들어 비트열

$$10101101\ 01101011\ 11001001\ 10101110 = ad6bc9ae$$

은 32-비트 워드로 $W = ad6bc9ae$ 가 된다. 이는 위의 4-바이트 문자열을 big-endian 컴퓨터에서 unsigned long의 형태로 형변환한 것과 같다.

2) 임의의 비트열 - 워드열 간의 변환

임의의 길이의 비트열을 32-비트 워드열로 변환하고자 할 때는 이 비트열을 바이트 열로 보고 첫 4 바이트를 첫 워드로, 두 번째 4 바이트를 두 번째 워드로 변환하는 과정을 반복한다. 예를 들면 비트열

$$10101101\ 01101011\ 11001001\ 10101110\ 00111111\ 01011001\ 01000110$$
$$= ad6bc9ae\ 3f5946$$

은 32-비트 워드열로는 $ad6bc9ae\ 3f594600$가 된다.

3) 임의의 워드열 - 비트열 간의 변환

32 비트 워드열을 비트열로 변환시킬 때는 1)과 2)의 역 과정을 따른다.

나. 핵심 단계 연산 LH1-C

LH1-C는 통합 암호 모듈의 핵심 연산으로 해쉬함수 LH1-H와 permutation LH1-P를 구성하며 다음과 같이 정의된다.

LH1-C($S=(S_0,\cdots,S_7)$, Y, Ctr_0, Ctr_1, Sel_0, Sel_1)

$S_7 = S_7 \oplus MUX(Y, 0^{29}\|Ctr_0, Sel_0)$; $S_2 = S_2 \oplus MUX(S_7, 0^{29}\|Ctr_0, Sel_1)$;

$S_6 = S_6 \oplus MUX(S_3, 0^{30}\|Ctr_1, Sel_1)$; $S_0 = S_0 \oplus G(S_2)$; $S_4 = S_4 \oplus G(S_6)$;

$S_2 = S_0^{\lll 11}$; $S_6 = S_4^{\lll 17}$;

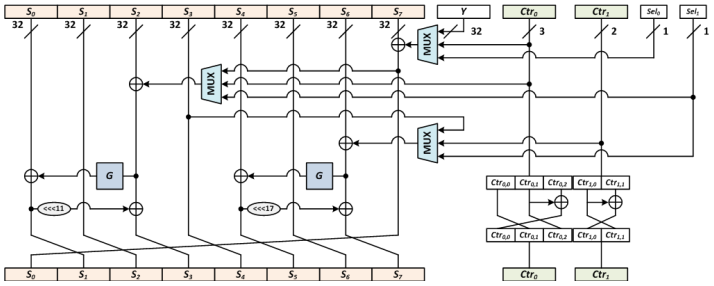
$T = S_7$; $S_7 = S_6$; $S_6 = S_5$; $S_5 = S_4$;

$S_4 = S_3$; $S_3 = S_2$; $S_2 = S_1$; $S_0 = T$;

$Ctr_0 = (Ctr_{0,1} \oplus Ctr_{0,2})\|Ctr_{0,0}\|Ctr_{0,1}$;

$Ctr_1 = (Ctr_{1,0} \oplus Ctr_{1,1})\|Ctr_{1,0}$;

LH1-C의 구조는 일반화된 Feistel 구조로써 내부적으로 MUX 함수를 사용하여 LH1을 해쉬함수 또는 블록암호로 동작하게끔 한다.



[그림 2-3] 핵심 단계 연산 LH1-C의 구조

1) MUX 함수

MUX 함수는 선택값(Sel_0, Sel_1)에 의해 2개의 입력값 중에 하나를 선택하는 함수이다. 이는 해쉬함수와 블록암호를 동시에 구성하는데 있어서 핵심이

되며 내부 상태를 갱신하는데 있어서 사용되는 값을 선택하는데 사용된다. 즉, 해쉬함수에서는 메시지가 사용될 것인지 상수가 사용될 것인지를 결정하며, 블록암호에서는 암호화를 할 것인지 키스케줄을 수행할 것인지를 결정한다.

$$MUX(X,Y,Sel) \mapsto \begin{cases} X & (\text{if } Sel = 0) \\ Y & (\text{if } Sel = 1) \end{cases}$$

2) G 함수

G 함수는 LH1-C의 내부함수로써 혼돈과 확산 효과를 주기위한 SL 함수와 DL 함수로 구성된다: $G(W) \mapsto DL(SL(W))$.

SL 함수는 혼돈 효과를 주기위한 대치함수로써, 32-비트 입력값 W을 8개의 4-비트 블록 $(w_0,\cdots,w_7)$로 나누어 각각에 대하여 S-박스 연산을 수행한다. 여기서 사용하는 4×4 S-박스는 수학적으로 다음과 같이 표기된다 ([표 2-7] 참조).

- $S(x_0,x_1,x_2,x_3) = (x'_0,x'_1,x'_2,x'_3)$.
  - $x'_0 = 1 \oplus x_0 \oplus x_1 \oplus x_2 \oplus x_3 \oplus x_0x_1 \oplus x_1x_2 \oplus x_0x_2x_3$.
  - $x'_1 = x_0 \oplus x_3 \oplus x_0x_2 \oplus x_1x_2 \oplus x_0x_1x_2 \oplus x_0x_1x_3$.
  - $x'_2 = x_2 \oplus x_0x_1 \oplus x_1x_3 \oplus x_2x_3 \oplus x_0x_2x_3$.
  - $x'_3 = x_0 \oplus x_1 \oplus x_2x_3$.

[표 2-7] S-박스

입력	0	1	2	3	4	5	6	7	8	9	a	b	c	d	e	f
출력	8	4	2	d	1	f	7	a	5	9	b	e	6	c	0	3

DL 함수는 32-비트 입력을 8개의 4 비트로 나누어 각각의 4 비트를 가지고 확산 효과를 주기 위하여 아래의 연산을 수행한다.

- $DL(w_0, w_1, w_2, w_3, w_4, w_5, w_6, w_7) = (w'_0, w'_1, w'_2, w'_3, w'_4, w'_5, w'_6, w'_7)$.

- $w'_0 = w_1 \oplus w_2 \oplus w_3 \oplus w_4 \oplus w_7$ - $w'_1 = w_0 \oplus w_2 \oplus w_3 \oplus w_4 \oplus w_5$.
- $w'_2 = w_0 \oplus w_1 \oplus w_3 \oplus w_5 \oplus w_6$ - $w'_3 = w_0 \oplus w_1 \oplus w_2 \oplus w_6 \oplus w_7$.
- $w'_4 = w_1 \oplus w_2 \oplus w_3 \oplus w_4 \oplus w_5$ - $w'_5 = w_0 \oplus w_2 \oplus w_4 \oplus w_5 \oplus w_6 \oplus w_7$.
- $w'_6 = w_0 \oplus w_1 \oplus w_3 \oplus w_4 \oplus w_6 \oplus w_7$ - $w'_7 = w_0 \oplus w_1 \oplus w_2 \oplus w_4 \oplus w_5 \oplus w_7$.

이는 Camellia의 P-함수[AIKOO]와 동일하며 아래와 같이 행렬 곱으로 표현할 수 있다.

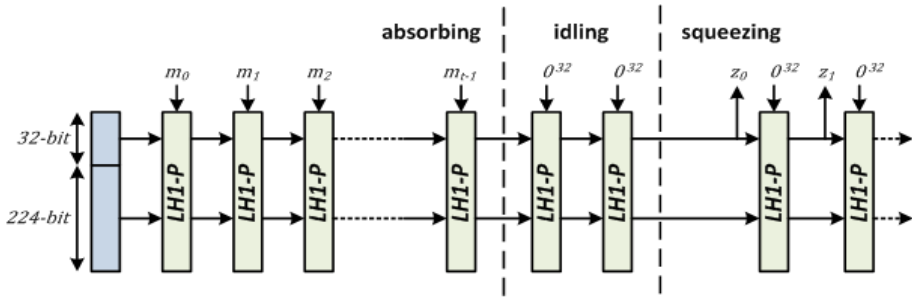
$$\begin{pmatrix} w'_0 \\ w'_1 \\ w'_2 \\ w'_3 \\ w'_4 \\ w'_5 \\ w'_6 \\ w'_7 \end{pmatrix} = \begin{pmatrix} 0\,1\,1\,1\,1\,0\,0\,1 \\ 1\,0\,1\,1\,1\,1\,0\,0 \\ 1\,1\,0\,1\,0\,1\,1\,0 \\ 1\,1\,1\,0\,0\,0\,1\,1 \\ 0\,1\,1\,1\,1\,1\,1\,0 \\ 1\,0\,1\,1\,0\,1\,1\,1 \\ 1\,1\,0\,1\,1\,0\,1\,1 \\ 1\,1\,1\,0\,1\,1\,0\,1 \end{pmatrix} \begin{pmatrix} w_0 \\ w_1 \\ w_2 \\ w_3 \\ w_4 \\ w_5 \\ w_6 \\ w_7 \end{pmatrix}$$

다. 해쉬함수 LH1-H

LH1-H는 absorbing 과정과 squeezing 과정 사이에 idling 과정이 추가된 변형된 Sponge 구조를 가진다. LH1-H는 임의의 길이를 가지는 입력 메시지를 32-비트 블록 단위로 나누어 처리하여 32~224 비트의 출력을 생성한다. 256-비트 내부 상태값을 처리하는 단계 함수는 모두 8 라운드로 구성되며, 해쉬값을 계산하는 연쇄변수는 8개이다. 또한 각 라운드에 적용될 메시지는 32 비트이다.

해쉬함수 LH1-H는 임의의 길이의 메시지 M이 입력으로 들어오면 그 M을 덧붙이기 과정을 통해 32 비트의 배수로 만든 후, 32-비트 블록 $m_i (0 \le i < t)$로 나눈다. [그림 2-3]과 같이 absorbing 과정에서는 각 블록 m_i를 압축하여 처리한다. 변형된 sponge 구조에 추가된 idling 과정에서는 빈 메시지를 입력으로 한 2번의 공회전을 수행한다. 해쉬값 출력을 수행하는 squeezing 과정에서는 해쉬값의 길이에 따라 추가적으로 단계 함수를 수행하여 $z_0, z_1, z_2, z_3, z_4, z_5, z_6$를 비트열로 변환

한 것이 해쉬값이 된다. 즉, 해쉬값이 32 비트인 경우 z_0를 해쉬값으로 출력하고, 해쉬값이 64 비트인 경우 $z_0||z_1$를 해쉬값으로 출력한다. 같은 논리를 이용하여 해쉬값이 224 비트인 경우 $z_0||z_1||z_2||z_3||z_4||z_5||z_6$을 해쉬값으로 출력한다.



[그림 2-4] 해쉬함수 LH1-H의 전체구조

1) 연쇄 변수 상태값 초기화

LH1-H의 연쇄 변수의 초기화에 사용되는 값은 16-진수 표기로 다음과 같다.

$H_0 = c24b8b70$, $H_1 = c76c51a3$, $H_2 = d192e819$, $H_3 = d6990624$,

$H_4 = f40e3585$, $H_5 = 106aa070$, $H_6 = 19a4c116$, $H_7 = 1e376c08$.

2) 메시지 덧붙이기

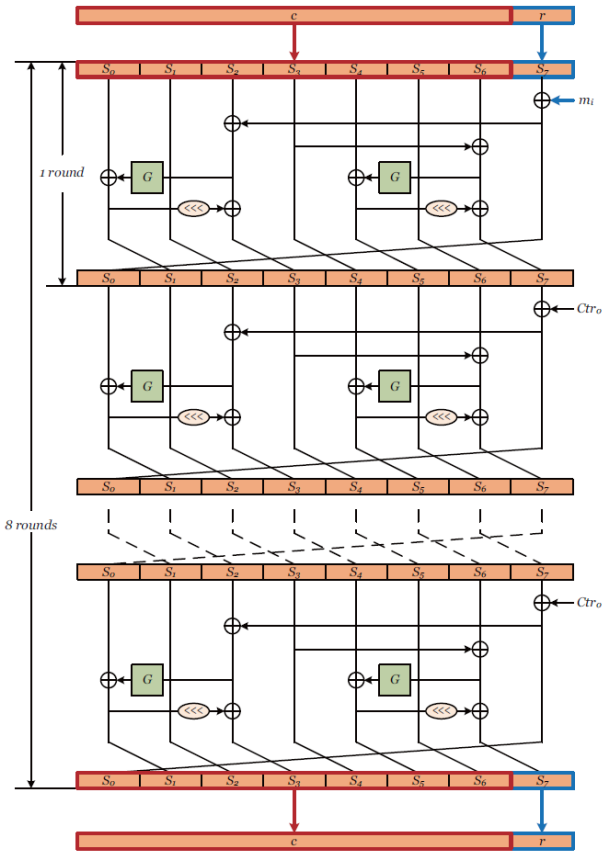
입력메시지는 32-비트 단위로 처리된다. 마지막 메시지 블록은 블록의 길이가 32비트가 되도록 $10\cdots0$을 채운다. 예를 들어 다음의 비트열이 주어졌다고 하자: 10100010 00111101 11100101 01101011 11001001 10001010 10011101. 이 비트열의 길이는 56이므로 1 다음에 7개의 0을 덧붙여 64 비트로 만든다. 따라서 그 결과 64 비트의 32-비트 워드열은 $a239e56b\,c98a9d80$로 주어진다.

3) Permutation LH1-P

LH1-P는 absorbing, idling, squeezing 과정에서 모두 사용된다. Absorbing

과정에서는 32-비트 블록 m_i를 처리하는 압축함수로 동작하며 idling 과정에서는 빈 메시지를 입력으로 한 2번의 공회전을 수행한다. Squeezing 과정에서는 해쉬값의 길이에 따라 추가적으로 단계 함수를 수행하여 해쉬값을 출력하는 역할을 수행한다.

LH1-P(S, m)
$Ctr_0 = 001_2$; $Ctr_1 = 01_2$;
LH1-C(S, m, Ctr_0, Ctr_1, 0, 0); // 메시지 처리
For $i = 1$ to 7
LH1-C(S, m, Ctr_0, Ctr_1, 1, 0); // 상수 사용



[그림 2-5] Permutation LH1-P의 구조

4) 해쉬값 출력

출력하고자 하는 해쉬값의 길이에 따라 추가적으로 압축함수를 수행하여 $z_0, z_1, z_2, z_3, z_4, z_5, z_6$를 워드열 - 비트열 변환 규칙에 의해 문자열로 변환된 후 해쉬값으로 출력된다. 즉 각 변수 z_i가 $z_i = z_{i0}z_{i1}z_{i2}z_{i3}$ (z_{ij}는 8-비트 비트열)와 같이 주어질 때 해쉬값은 다음과 같이 비트열이 된다.

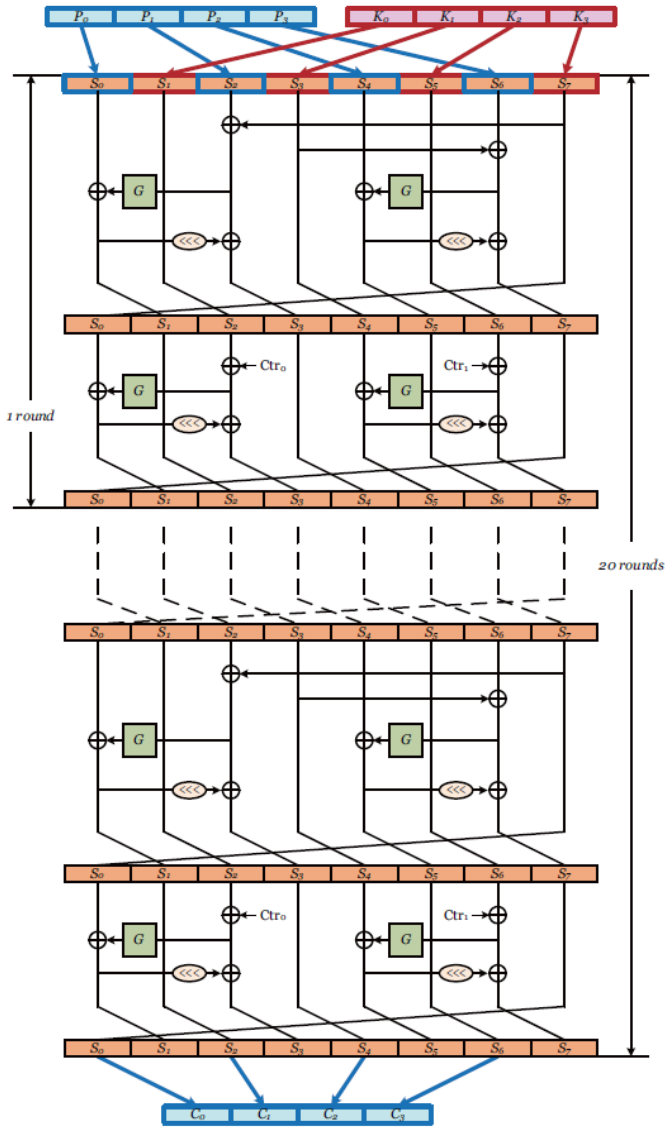
$z_{00}z_{01}z_{02}z_{03}z_{10}z_{11}z_{12}z_{13}z_{20}z_{21}z_{22}z_{23}z_{30}z_{31}z_{32}z_{33}z_{40}z_{41}z_{42}z_{43}z_{50}z_{51}z_{52}z_{53}z_{60}z_{61}z_{62}z_{63}$.

라. 블록암호 LH1-B

블록암호 LH1-B는 128-비트 평문 P와 128-비트 비밀키 K를 입력받아 128-비트 암호문 C를 출력한다. 128-비트 평문 $P=(P_0,P_1,P_2,P_3)$와 비밀키 $K=(K_0,K_1,K_2,K_3)$는 4개의 32-비트 단위로 나누어 LH1-B의 8개의 32-비트 내부 상태 중 평문은 (S_0,S_2,S_4,S_6)에, 비밀키는 (S_1,S_3,S_5,S_7)에 입력된다. 전체 암호화 과정은 총 20 라운드로 구성되며 각 라운드마다 LH1-C 함수가 2번씩 사용되는데 첫 번째는 암호화이며 두 번째는 키스케줄에 해당한다. 해쉬함수와 달리 전체 구조에서 idling 과정과 squeezing 과정은 생략되며, 20 라운드로 구성된 암호화 과정이 끝나면 홀수 번째의 내부 상태 값이 다음과 같이 암호문이 된다:

$$C=(C_0,C_1,C_2,C_3)=(S_0,S_2,S_4,S_6).$$

LH1-B($P=(P_0,P_1,P_2,P_3)$, $K=(K_0,K_1,K_2,K_3)$, $C=(C_0,C_1,C_2,C_3)$)
- $S_0=P_0$; $S_1=K_0$; $S_2=P_1$; $S_3=K_1$;
- $S_4=P_2$; $S_5=K_2$; $S_6=P_3$; $S_7=K_3$;
- $Ctr_0=0001_2$; $Ctr_1=01_2$;
- For $i=1$ to 20
  - LH1-C(S, 0, Ctr_0, Ctr_1, 0, 1); // 암호화
  - LH1-C(S, 0, Ctr_0, Ctr_1, 0, 0); // 키스케줄
- $C_0=S_0$; $C_1=S_2$; $C_2=S_4$; $C_3=S_6$;

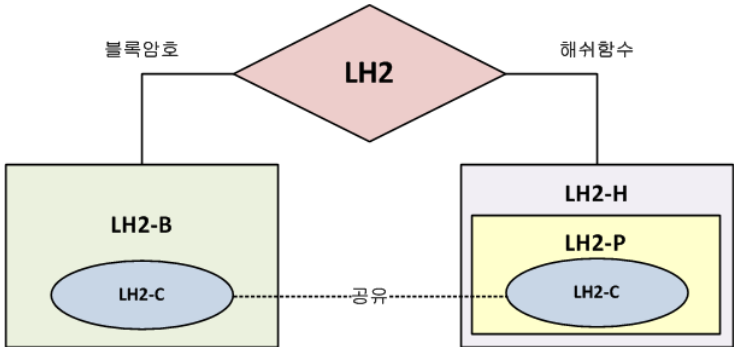


[그림 2-6] 블록암호 LH1-B의 구조

3. LH2 알고리즘 소개

LH2는 [그림 2-6]과 같이 LH1과 유사하게 하나의 통합된 암호 모듈을 이용하여 블록암호와 해쉬함수를 구성하도록 설계되었다. LH2는 크게 해쉬함수 LH2-H와 블

록암호 LH2-B로 구성된다. LH2-P는 해쉬함수에서 사용되는 permutation이며 핵심 단계 연산인 LH2-C를 블록암호 LH2-B와 공유하여 사용한다. LH1과 LH2의 차이점은 [표 2-8]와 같다.



[그림 2-7] LH2의 계층 구조

[표 2-9] LH1과 LH2 비교

	LH1	LH2
전체구조	변형된 Sponge 구조	Sponge 구조
연산	S-박스, Rotation, XOR	ARX(Addition, Rotation, XOR)

가. 비트열 - 워드 간의 변환

LH2의 내부 연산은 LH1과 동일한 32-비트 정수간의 연산이 사용되며 비트열과 워드 간의 변환 규칙 또한 일치한다.

나. 핵심 단계 연산 LH2-C

LH2-C는 통합 암호 모듈의 핵심 연산으로 해쉬함수 LH2-H와 permutation LH2-P를 구성하며 다음과 같이 정의된다.

LH2-C($S=(S_0,\cdots,S_7),\ Y,\ Z,\ Sel$)

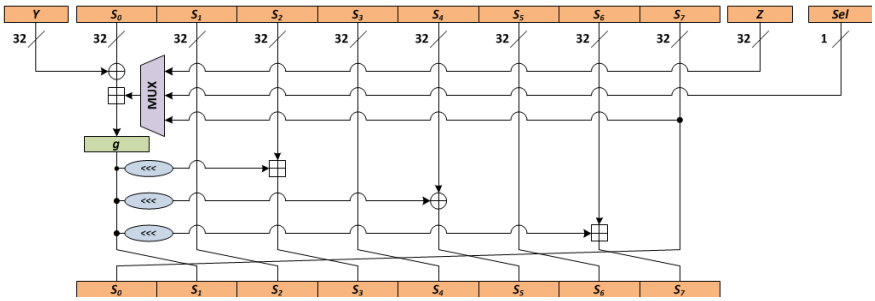
$T_1 = g((S_0 \oplus Y) + MUX(Z, S_7, Sel));$

$T_2 = S_7;$

$S_7 = S_6 + T_1^{\lll 29};\ S_6 = S_5;\ S_5 = S_4 \oplus T_1^{\lll 7};\ S_4 = S_3;$

$S_3 = S_2 + T_1^{\lll 13};\ S_2 = S_1;\ S_1 = T_1;\ S_0 = T_2;$

LH2-C의 구조는 일반화된 Feistel 구조로써 내부적으로 MUX 함수를 사용하여 LH2를 해쉬함수 또는 블록암호로 동작하게끔 한다.



[그림 2-8] 핵심 단계 연산 LH2-C의 구조

1) MUX 함수

LH2-C에서 사용되는 MUX 함수는 LH1-C에서의 MUX 함수와 동일하다.

$$MUX(X, Y, Sel) \mapsto \begin{cases} X & (\text{if } Sel = 0) \\ Y & (\text{if } Sel = 1) \end{cases}$$

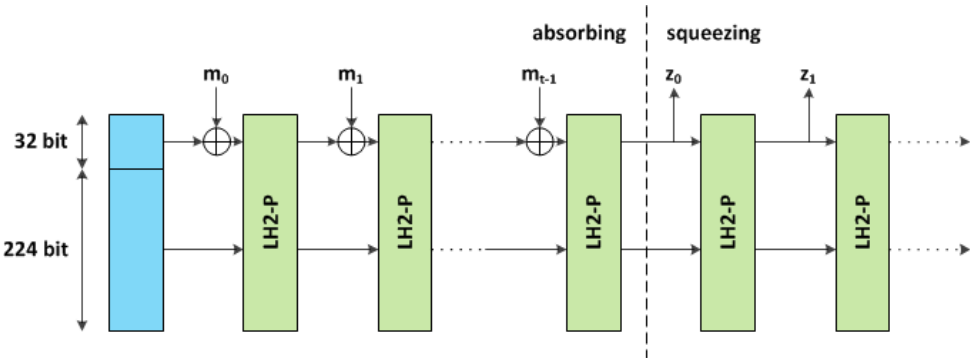
2) g 함수

g 함수는 32-비트 워드를 입력 받아 32-비트 워드를 출력하는 함수로서, 다음과 같이 정의된다.

$$g(x) = x \oplus x^{\lll 7} \oplus x^{\lll 17} \oplus x^{\lll 13} \oplus x^{29}.$$

다. 해쉬함수 LH2-H

LH2-H는 [그림 2-8]과 같은 Sponge 구조를 가지며, 임의의 길이를 가지는 입력 메시지를 32-비트 블록 단위로 처리하여 32~224 비트의 출력을 생성한다. 256-비트 내부 상태값을 처리하는 단계 함수는 모두 5 라운드로 구성되며, 해쉬값을 계산하는 연쇄 변수는 8개이다. 또한 각 라운드에 적용될 메시지는 32 비트이다.



[그림 2-9] 해쉬함수 LH2-H의 전체구조

해쉬함수 LH2-H는 임의의 길이의 메시지 M이 입력으로 들어오면 그 M을 덧붙이기 과정을 통해 32 비트의 배수로 만든 후, 32-비트 블록 $m_i(0 \le i < t)$로 나눈다. 각 블록 m_i를 [그림 2-8]과 같은 과정을 통해 압축하여 최종 블록 처리 후, 해쉬값의 길이에 따라 추가적으로 단계 함수를 수행하여 $z_0, z_1, z_2, z_3, z_4, z_5, z_6$를 비트열로 변환한 것이 해쉬값이 된다. 즉, 해쉬값이 32 비트인 경우 z_0를 해쉬값으로 출력하고, 해쉬값이 64 비트인 경우 $z_0||z_1$를 해쉬값으로 출력한다. 같은 논리를 이용하여 해쉬값이 224 비트인 경우 $z_0||z_1||z_2||z_3||z_4||z_5||z_6$을 해쉬값으로 출력한다.

1) 연쇄 변수 상태값 초기화

LH2-H의 연쇄 변수의 초기화에 사용되는 값은 16-진수 표기로 다음과 같다.

$H_0 = 6a09e667$, $H_1 = bb67ae85$, $H_2 = 3c6ef372$, $H_3 = a54ff53a$,

$H_4 = 510e527f$, $H_5 = 9b05688c$, $H_6 = 1f83d9ab$, $H_7 = 5be0cd19$

2) 메시지 덧붙이기

LH2-H의 메시지 덧붙이기 과정은 LH1-H의 메시지 덧붙이기 과정과 동일하다.

3) Permutation LH2-P

LH2-P는 absorbing, squeezing 과정에서 모두 사용된다. Absorbing 과정에서는 32-비트 블록 m_i를 처리하는 압축함수로 동작하며 squeezing 과정에서는 해쉬값의 길이에 따라 추가적으로 단계 함수를 수행하여 해쉬값을 출력하는 역할을 수행한다. 여기서, 라운드 상수 $Ctr_i (0 \le i \le 4)$는 서로 다른 고정된 상수를 의미한다.

```
LH2-P(S)
    For i = 0 to 4
        LH1-C(S, Ctr_i, 0, 1);
```

4) 해쉬값 출력

모든 32-비트 메시지 블록을 처리한 후, 해쉬값의 길이에 따라 추가적으로 압축함수를 수행하여 $z_0, z_1, z_2, z_3, z_4, z_5, z_6$를 워드열 - 비트열 변환 규칙에 의해 문자열로 변환된 후 해쉬값으로 출력된다. 즉 각 변수 z_i가 $z_i = z_{i0}z_{i1}z_{i2}z_{i3}$ (z_{ij}는 8-비트 비트열)와 같이 주어질 때 해쉬값은 다음과 같이 비트열이 된다.

$z_{00}z_{01}z_{02}z_{03}z_{10}z_{11}z_{12}z_{13}z_{20}z_{21}z_{22}z_{23}z_{30}z_{31}z_{32}z_{33}z_{40}z_{41}z_{42}z_{43}z_{50}z_{51}z_{52}z_{53}z_{60}z_{61}z_{62}z_{63}$.

라. 블록암호 LH2-B

블록암호 LH2-B는 128-비트 평문 P와 128-비트 비밀키 K를 입력받아 128-비트 암호문 C를 출력한다. 128-비트 평문 $P=(P_0,P_1,P_2,P_3)$와 비밀키 $K=(K_0,K_1,K_2,K_3)$는 4개의 32-비트 단위로 나누어 LH2-B의 8개의 32-비트 내부 상태 중 평문은 (S_0,S_2,S_4,S_6)에, 비밀키는 (S_1,S_3,S_5,S_7)에 입력된다. 전체 암호화 과정은 총 20 라운드로 구성되며 각 라운드마다 LH2-C 함수가 2번씩 사용되는데 첫 번째는 암호화이며 두 번째는 키스케줄에 해당한다. 20 라운드로 구성된 암호화 과정이 끝나면 홀수 번째의 내부 상태값이 암호문이 된다: $C=(C_0,C_1,C_2,C_3)=(S_0,S_2,S_4,S_6)$. 여기서, 라운드 상수 $Ctr_i(0 \le i \le 19)$는 서로 다른 고정된 상수를 의미한다.

LH2-B($P=(P_0,P_1,P_2,P_3)$, $K=(K_0,K_1,K_2,K_3)$, $C=(C_0,C_1,C_2,C_3)$)
  $S_0=P_0$; $S_1=K_0$; $S_2=P_1$; $S_3=K_1$;
  $S_4=P_2$; $S_5=K_2$; $S_6=P_3$; $S_7=K_3$;
  For $i=0$ to 19
    LH2-C(S, 0, Ctr_i, 1); // 암호화
    LH2-C(S, 0, Ctr_0, Ctr_1, 0); // 키스케줄
  $C_0=S_0$; $C_1=S_2$; $C_2=S_4$; $C_3=S_6$;

4. LH3 알고리즘 소개

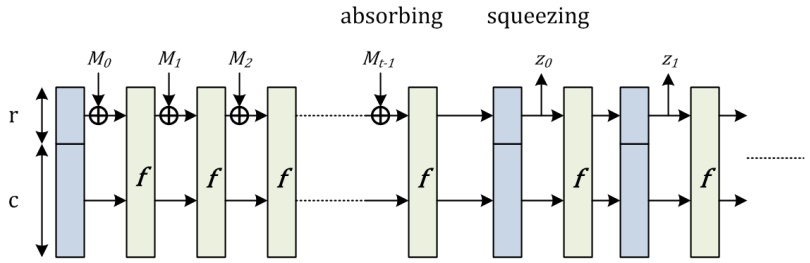
전 세계적으로 유비쿼터스 환경 등에 적합하도록 설계된 경량 암호들 중에서, LFSR 기반 스트림 암호에 기반을 둔 설계 사상을 갖는 알고리즘이 꽤 많은 비중을 차지한다. 대표적인 예가 스트림 암호 Trivium[Can06], 블록암호 KATAN[CDK09], 해쉬함수 Quark[AHM11]를 들 수 있다. 이 연구 흐름에 맞춰, LH3와 다음 절에서 소개될 LH4는 LFSR 기반 스트림 암호의 설계 사상을 따른다. 즉, LH3의 단계 함수

는 1-비트 메시지를 처리하며, 1-비트 해쉬값을 출력한다.

가. 비트열 - 워드 간의 변환

LH3의 내부 연산은 1-비트 단위 연산이 사용된다. 따라서 LH3는 비트열 - 워드 간의 변환이 필요 없다.

나. 전체구조



[그림 2-10] LH3의 전체구조

LH3는 임의의 길이를 가지는 입력 메시지를 1-비트 단위로 처리하여 32~224 비트의 출력을 한다. 256-비트 내부 상태값을 처리하는 단계 함수는 모두 256 라운드로 구성되며, 각각의 라운드마다 4개의 내부 함수가 수행된다. 또한 각 라운드에 적용될 메시지는 1비트이다. 즉, $c+r=256$를 만족하며 $r=1$을 만족한다.

LH3는 임의의 길이의 메시지 M이 입력으로 들어오면, 1-비트 $M_i(0 \le i < t)$씩 처리한다. 각 M_i를 [그림 2-9]와 같은 과정을 통해 압축하여 처리 후, 해쉬값의 길이에 따라 추가적으로 단계 함수를 수행하여 $z_0,\cdots,z_{223}$이 해쉬값이 된다. 즉, 해쉬값이 32 비트인 경우 $z_0,\cdots,z_{31}$을 해쉬값으로 출력하고, 해쉬값이 64 비트인 경우 $z_0,\cdots,z_{63}$를 해쉬값으로 출력한다. 같은 논리를 이용하여 해쉬값이 224 비트인 경우 $z_0,\cdots,z_{223}$을 해쉬값으로 출력한다.

1) 내부 상태값 초기화

LH3의 내부 상태값의 초기화 과정은 다음과 같다. 먼저 256-비트 내부 상태값을 0으로 고정한 후, 단계 함수 f를 1번 수행한다. 이 과정을 수행한 후의 내부 상태값이 초기 상태값이다.

2) 메시지 덧붙이기

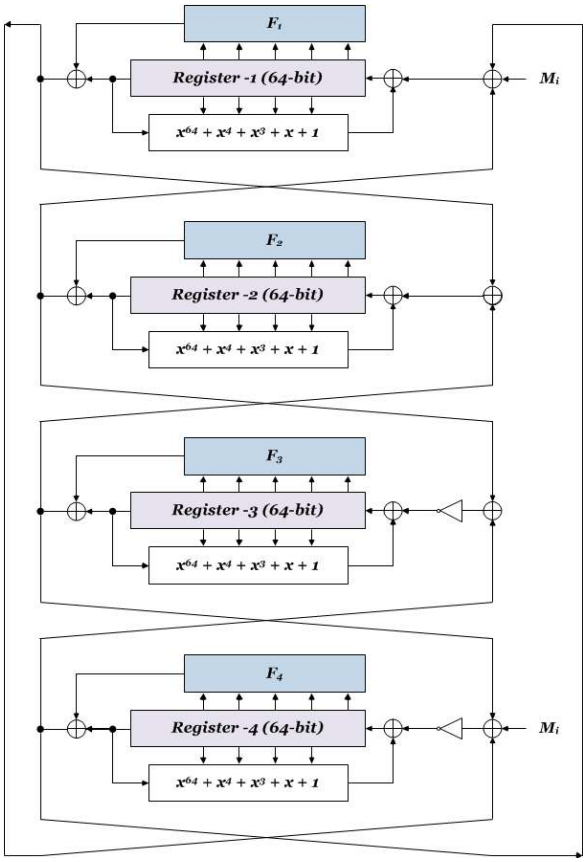
LH3에서는 1-비트 단위로 메시지를 처리하기 때문에, 메시지 덧붙이기 과정이 필요 없다.

3) 단계 함수

단계 함수 f는 [그림 2-10]의 과정을 256번 반복 수행한다. 이 함수는 4개의 내부 함수로 구성되며, 각각의 내부 함수는 64-비트 LFSR과 2차 비선형 함수로 구성된다. 비선형 함수 F_1, F_2, F_3, F_4는 다음과 같이 정의된다. 이때, 64-비트 레지스터는 $(x_{63}, x_{62}, \cdots, x_1, x_0)$으로 표기된다.

- $F_1(x_1, x_{15}, x_{27}, x_{39}, x_{51}) = x_{51} \oplus x_{39}x_{27} \oplus x_{15}x_1$
- $F_2(x_2, x_{11}, x_{24}, x_{37}, x_{50}) = x_{50} \oplus x_{37}x_{24} \oplus x_{11}x_2$
- $F_3(x_3, x_{17}, x_{29}, x_{33}, x_{53}) = x_{53} \oplus x_{33}x_{29} \oplus x_{17}x_3$
- $F_4(x_4, x_{14}, x_{27}, x_{30}, x_{46}) = x_{46} \oplus x_{30}x_{27} \oplus x_{14}x_4$

각각의 내부 함수의 출력값은 연접해 있는 내부 함수에 영향을 준다. 즉, 첫 번째 내부 함수의 출력값은 두 번째, 네 번째 내부 함수에 영향을 준다 ([그림 2-10] 참조).



[그림 2-11] 해쉬함수 LH3의 단계 함수

4) 해쉬값 출력

모든 메시지를 처리한 후, 해쉬값의 길이에 따라 추가적으로 단계 함수를 수행하여 Register-3의 최상위 비트 x_{63}을 1-비트 해쉬값으로 출력한다.

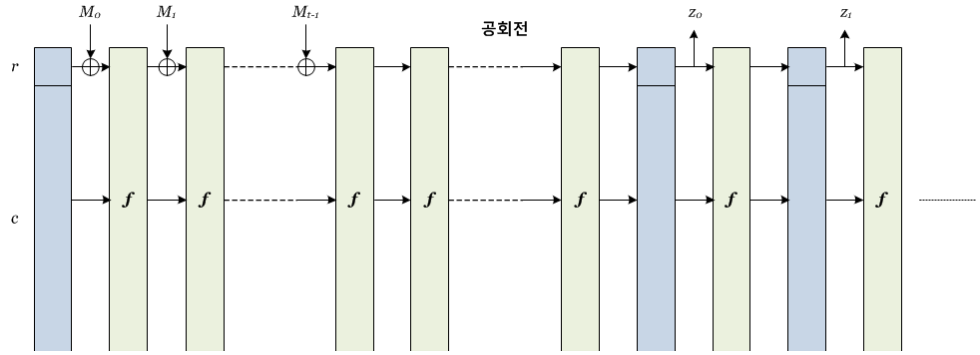
5. LH4 알고리즘 소개

LH4는 LH3과 유사하게 LFSR 기반 스트림 암호의 설계 사상을 따른다. 하지만, LH4는 1 라운드로 구성된 단계 함수를 사용하고 공회전 단계가 포함된다. 또한 단계 함수는 3개의 내부 함수로 구성된다.

가. 비트 열 - 워드 간의 변환

LH4의 내부 연산은 1-비트 단위의 연산이 사용된다. 따라서 LH4는 비트열 - 워드 간의 변환이 필요 없다.

나. 전체구조



[그림 2-12] LH4의 전체구조

LH4에는 공회전 구간이 포함되어 있다. 따라서 LH4의 전체구조는 [그림 2-11]과 같은 변형된 Sponge 구조이다. 이 구조는 공회전 구간을 제외하면 LH3의 전체구조와 동일하다. 즉, LH4는 임의의 길이를 가지는 입력 메시지를 1 비트 단위로 처리하여 32~224 비트의 출력을 한다. 256-비트 내부 상태값을 처리하는 단계 함수는 모두 1 라운드로 구성되며, 각각의 라운드마다 3개의 내부 함수가 수행된다. 또한 각 라운드에 적용될 메시지는 1 비트이다. 즉, $c+r=256$를 만족하며 $r=1$을 만족한다.

LH4는 임의의 길이의 메시지 M이 입력으로 들어오면, 1-비트 $M_i(0 \le i < t)$씩 처리한다. 각 M_i를 [그림 2-11]과 같은 과정을 통해 압축하여 처리 후, 공회전 단계를 수행한다. 공회전 단계에서는 1-비트 메시지를 입력 받지 않고 단계 함수가 384번 반복 수행된다. 공회전 단계를 수행한 후, 해쉬값의 길이에 따라 추가적으로 단계 함수를 수행하여 $z_0,\cdots,z_{223}$이 해쉬값이 된다. 즉, 해쉬값이 32 비트인

경우 $z_0,\cdots,z_{31}$을 해쉬값으로 출력하고, 해쉬값이 64 비트인 경우 $z_0,\cdots,z_{63}$를 해쉬값으로 출력한다. 같은 논리를 이용하여 해쉬값이 224 비트인 경우 $z_0,\cdots,z_{223}$을 해쉬값으로 출력한다.

1) 내부 상태값 초기화

LH4의 초기화 과정에서는, 256-비트 내부 상태값이 0으로 고정된 상태에서 해쉬값의 길이가 메시지로 입력된 후, 384번의 공회전 단계를 수행한다. 이 단계를 수행한 후의 내부 상태값이 초기 상태값이다.

2) 메시지 덧붙이기

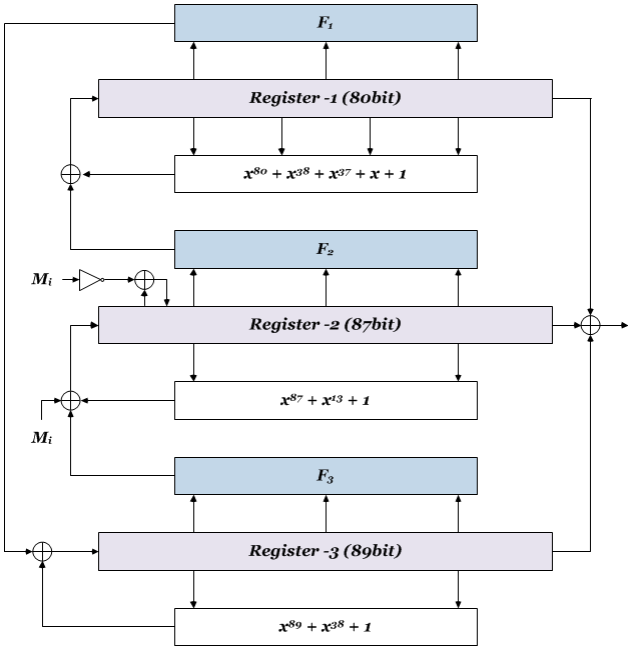
LH3에서는 1-비트 단위로 메시지를 처리하기 때문에, 메시지 덧붙이기 과정이 필요 없다.

3) 단계 함수

단계 함수 f는 [그림 2-12]와 같다. LH3의 압축 함수와 달리, 이 함수는 1번만 수행된다. 스트림 암호의 관점에서 보면, 이 과정은 1 clock 과정과 같다. 총 3개의 내부 함수로 구성되며, 각각의 내부 함수는 80/87/89-비트 LFSR과 3차 비선형 함수로 구성된다. 비선형 함수 F_1,F_2,F_3는 다음과 같이 정의된다. 이때, 레지스터의 인덱스는 LH3과 동일하다.

- $F_1(x_{20},x_{40},x_{60}) = x_{40} \oplus x_{60} \oplus x_{20}x_{40} \oplus x_{20}x_{60} \oplus x_{20}x_{40}x_{60}$
- $F_2(x_{20},x_{40},x_{60}) = 1 \oplus x_{20} \oplus x_{40} \oplus x_{60} \oplus x_{20}x_{40} \oplus x_{20}x_{40}x_{60}$
- $F_3(x_{20},x_{40},x_{60}) = 1 \oplus x_{20} \oplus x_{40} \oplus x_{20}x_{60} \oplus x_{40}x_{60} \oplus x_{20}x_{40}x_{60}$

Register-2에서 $\widetilde{M}_i$가 XOR되는 부분은 x_{43}이다.



[그림 2-13] 해쉬함수 LH4의 단계 함수

4) 해쉬값 출력

모든 메시지를 처리한 후, 384번의 공회전 단계가 수행된다. 공회전 단계가 수행된 후의 각각의 LFSR 출력 비트들은 XOR 연산을 거쳐 첫 번째 해쉬값 비트로 출력된다. 이후, f를 1번 동작시킨 후 각각의 LFSR 출력 비트들을 XOR하여 해쉬값으로 출력시킨다.

제 3 장 경량 그룹 서명 프로토콜 개발

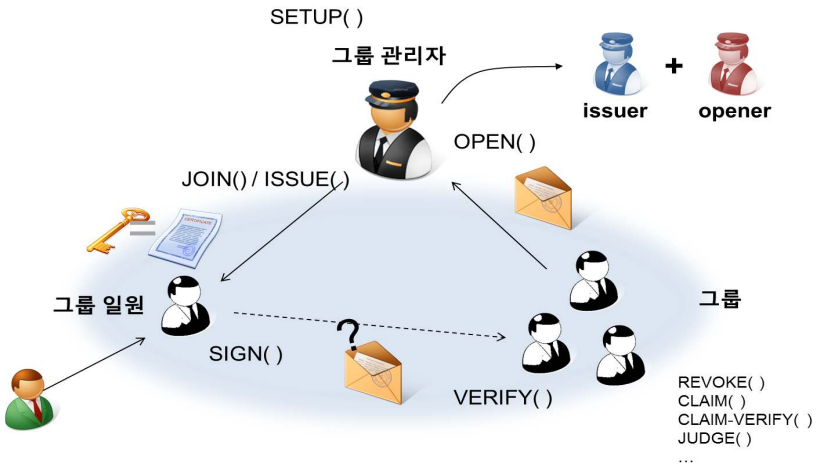
제 3 장 경량 그룹 서명 프로토콜 개발

제 1 절 연구 개발 필요성

본 고에서는 서명자의 익명성을 제공하는 그룹 서명 프로토콜을 소개하고 최근 연구 동향을 살핌으로써 경량 그룹 서명 프로토콜의 필요성에 대해 살펴본다.

1. 그룹 서명 프로토콜 소개

1991년 D. Chaum과 E. van Heyst는 그룹 서명을 소개하였다[CH91]. 그룹 서명은 서명 생성자의 신원이 직접적으로 드러내지 않고 소속 그룹의 구성원만 보임으로써 서명의 정당성을 증명하여 서명자의 익명성을 제공하는 서명 기법이다. 서명의 생성을 위해 신원과 관련된 정보가 직접 사용되지 않기 때문에 서명 데이터를 통하여 서명 생성자의 신원을 직접적으로 알 수는 없고 생성된 정보에서 신원과 관련된 정보를 찾아낼 수 없도록 설계되기 때문에 서명 생성자의 신원이 보호된다. 서명 검증은 그룹의 공개키에 해당하는 정보를 사용하여 수행되기 때문에 서명을 검증하는 주체는 주어진 서명을 생성한 주체가 특정 그룹의 구성원이라는 것을 확인함으로써 서명의 정당성을 검증할 수 있다. 이때 무조건적인 익명성이 주어지면 적법한 그룹 구성원에 의해 이루어지는 부정한 서명 생성에 대한 제재가 어렵게 된다. 이를 방지하기 위해 그룹 매니저에게 그룹의 어떠한 서명이라도 실제 서명자를 공개할 수 있는 능력을 제공하여 부정한 서명 생성 행위를 관리할 수 있는 능력이 제공된다. 서명자의 익명성이 가장 중요한 그룹 서명 프로토콜을 설계함에 있어 이와 같은 그룹 서명의 특성은 매우 유용하며 그룹 서명 프로토콜의 설계에 있어 매우 중요하게 사용된다. 따라서 본 고에서는 그룹 서명을 위주로 논의를 진행한다.



[그림 3-1] 그룹 서명 기법의 정의

그룹 서명 기법의 구성을 알고리즘을 기반으로 간략히 살펴보자. 이는 그림 [그림 3-1]에서 볼 수 있다. 그룹 서명에는 그룹을 관리하는 그룹 매니저와 다른 여러 그룹의 일원이 참여한다. 우선 그룹 매니저는 그룹 마스터키와 그룹 공개키 등을 생성한다. 그 후, JOIN()이나 ISSUE()를 통하여 그룹의 일원은 그룹에 가입할 수 있으며, 그룹에 가입한 그룹의 일원은 SIGN()을 통해 익명 서명을 생성할 수 있다. 그룹의 공개키를 소지한 사람은 누구든지 생성된 익명 서명을 VERIFY()를 통해 확인할 수 있다. 이 과정에서 서명을 확인한 당사자는 해당 서명이 그룹의 한 일원이 생성하였다는 사실을 알 수 있지만, 누가 생성하였는지는 알 수 없다. 예외적인 상황에서 그룹 매니저는 OPEN()을 통하여 익명 서명을 생성한 서명자를 밝혀낼 수 있다. 추가적으로, 그룹 서명에서는 악의적인 행동을 한 그룹의 일원은 그룹 매니저와 그룹 전체가 수행하는 REVOKE() 과정을 통하여 더 이상 그룹의 일원이 아니게 조정될 수 있다. 또한 어떠한 그룹의 일원이 주어진 익명 서명에 대해서 자신이 생성한 서명임을 증명하기 위해서 CLAIM()을 수행할 수 있다. 다른 사용자들은 CLAIM-VERIFY()를 통하여 CLAIM()으로부터 출력된 증명이 올바른

것인지 확인할 수도 있다. 또한 경우에 따라서 OPEN()의 결과가 신뢰할 수 있는지 확인하기 위해 JUDGE()를 수행할 수 있다. 그룹 매니저는 수행하는 역할에 따라 두 개로 나뉠 수 있다. JOIN()이나 ISSUE()를 수행하여 그룹의 일원에게 그룹 멤버십 증명서(또는 그룹 서명키)를 발급해주는 발행자(Issuer)와 필요에 따라 서명이 갖는 서명자의 익명성을 철폐하기 위해서 OPEN()을 수행하는 공개자(Opener)가 그것이다.

가. 그룹 서명 기법의 안전성 요구 사항

2003년 M. Bellare 등이 [BMW03]에서 그룹 서명 기법의 안전성에 대해 이론적으로 정형화된 보안 요구 사항을 제시하기 전에는 그룹 서명 기법은 다양한 보안 요구 사항을 만족하도록 설계되었다. 그룹 서명 기법은 개념적으로는 다음과 같은 다양한 보안요건을 만족하도록 설계된다.

- **정확성(Correctness)**: 정상적인 서명은 언제나 유효하고, 정상적인 서명으로부터 정상적으로 서명자의 아이디를 꺼낼 수 있다.
- **위조 불가성(Unforgeability)**: 그룹 멤버만이 유효한 서명을 생성할 수 있다.
- **익명성(Anonymity)**: 유효한 서명이 주어졌을 때, 그룹 매니저를 제외하고 서명자를 알아내는 것은 매우 힘들다.
- **연결 불가성(Unlinkability)**: 두 개의 서명이 주어졌을 때, 두 서명이 동일한 서명자가 생성한 서명인지 알아내는 것은 매우 힘들다
- **무죄 증명성(Exculpability)**: 그룹의 일원이 다른 그룹 멤버인 것처럼 서명을 생성하는 것은 매우 힘들다. 후에 그룹 매니저를 포함하는 것과 대비하기 위하여, 그룹 매니저를 제외하는 경우를 무죄 증명성, 그룹 매니저를 포함하는 경우를 강한 무죄 증명성(Strong Exculpability)로 구별하여 부르기도 한다.

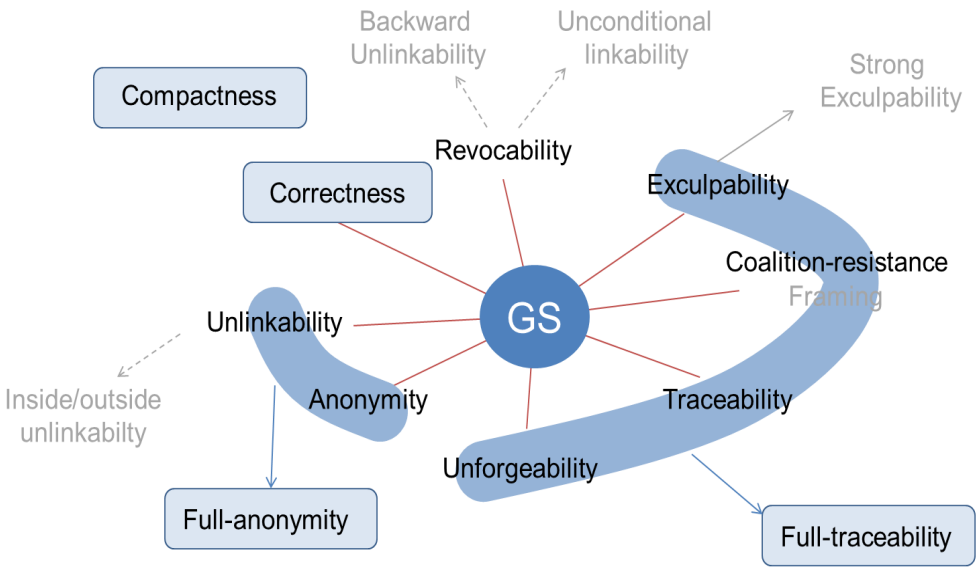
○ **추적 가능성(Traceability)**: 그룹 매니저는 언제나 유효한 서명으로부터 서명자의 아이디를 추출할 수 있다.

○ **공모 저항성(Coalition-resistance, Non-Framing)**: 다수의 그룹 멤버가 협력하여 유효한 서명을 생성할 수 없다. 즉, 다수의 그룹 멤버들이 자신의 키를 조합하여 그룹의 다른 일원으로 판명되도록 유효한 서명을 생성할 수 있다.

○ **철폐성(Revocability)**: 제외된 그룹 멤버가 서명한 서명은 유효하지 않아야 한다.

○ **무조건적인 연결성(Uncoditional Linkability)**: 몇몇 그룹 서명은 후방 연결 불가성(Backward Unlinkability)가 지원된다. 후방 연결 불가능성은 제외된 그룹의 일원이 생성한 서명이라도, 제외 이전에 생성된 서명이라면 유효하게 남겨져 있어야 한다는 것이다.

이와 같은 보안 요구 사항은 [그림 3-2]와 같은 관계를 가지고 있다.



[그림 3-2] 그룹 서명과 요구 사항들 사이의 연관 관계

상기 기술된 보연 요건들은 2003년 M. Bellare, D. Micciancio, 그리고 B. Warinschi에 의해 제안된 안전성 모델에 의해 간소화되었다[BMW03]. 이전에 다수의 기준들에 의해 분석되어야 했던 안전성을 축약된 소수의 기준에 대한 안전성으로 모두 검증 가능하다는 이론적인 개선은 이루었으나 정적인 그룹 서명에 대한 안전성 모델이라는 한계를 가진다. 이후 M. Bellare와 H. Shi, G. Zhang이 동적인 그룹 서명에 대한 안전성 모델을 제시하였으며 동적인 그룹 서명 기법의 안전성에 대해서는 현재까지 가장 잘 정형화된 안전성 기준으로 평가받고 있다 [BSZ05].

[BSZ05]에서 제안된 모델은 [BMW03]에서 제안된 모델에서 일부 변형된 것으로 [BMW03]에 기술된 보안 요구사항을 먼저 살피고 차이점을 확인함으로써 [BSZ05]에 제안된 안전성 기준에 대한 설명을 대신한다. [BMW03]에서 제안된 안전성 모델은 다음과 같은 기준을 안전성 요건으로 제시하고 있다.

- **완전 익명성(Full-anonymity):** 그룹 매니저의 비밀키를 가지고 있지 않은 공격자는 서명으로부터 서명자의 아이디를 알아내는 것이 힘들다. 공격자에게 공개(Open) 오라클로의 접근을 허용한다. 기존의 익명성(Anonymity)는 공개 오라클로의 접근을 허용하지 않았다. 기존의 연결 불가성(Unlinkability)과 익명성(Anonymity)은 이 완전 익명성으로 대체될 수 있다.
- **완전 추적성(Full-traceability):** 어떠한 그룹 멤버의 협력 집단으로 열리지 않는 서명을 생성할 수 없거나, 협력 그룹의 한 일원으로 추적이 되지 않는 서명을 생성할 수 없다. 기존의 위조 불가성(Unforgeability)과 추적성(Traceabiltiy), 공모 저항성(Coalition-resistance), 무죄 증명성(Exculapbility)는 이 완전 추적성으로 대체될 수 있다.

[BSZ05]의 모델에서는 동적으로 구성되는 발행자와 공개자가 존재할 때, 요구되는 보안 요구 사항을 추가로 정의하였다. 발행자와 여러 공개자가 존재할 수 있기 때문에, 발행자나 공개자를 기본적으로 신뢰할 수 없다는 가정 아래에서의 안전성을 고려한다. 특히 공개자는 OPEN()에서 잘못된 결과를 출력할 수 있기 때문

에, 이를 JUDGE()를 통해 검증하는 과정이 필수로 제공되어야 한다는 제약 사항이 추가되었다. 이 점이 [BMW03]에서 제시된 안전성 모델과의 가장 큰 차이점이다. [BSW05]에서 제안된 안전성 모델은 다음과 같은 기준을 안전성 요건으로 제시하고 있다.

- **익명성(Anonymity):** [BMW03]의 완전 익명성(Full-anonymity)와 동일하다.
- **추적 가능성(Traceability):** 공격자는 정직한 공개자가 서명의 생성자가 누구인지 모르겠다고 선언하거나, 정직한 공객자가 서명의 생성자가 누구인지 알 수 있지만, 정상적인 선언(Claim)에 대한 증명의 생성이 불가능한 서명을 생성할 수 없다.
- **무죄 증명성(Non-Frameability):** 공격자는 정직한 사용자가 정말로 해당 서명을 생성하지 않는 한, JUDGE()의 결과를 통과하는 증명을 생성할 수 없다.

나. 그룹 서명 기법의 구성

그룹 서명 기법 $GS=(GKg,GSig,GVf,Open)$은 다음과 같은 다항식 시간(Polynomial time) 알고리즘으로 구성된다.

- $(gpk,gmsk,gsk)=GKg(1^k,1^n)$ (그룹 키 생성 알고리즘): 입력으로 보안 파라메터 k와 그룹의 크기 n을 받고 그룹 공개키 gpk, 그룹 마스터키 $gmsk$, 비밀 서명키 gsk를 반환한다. gsk는 특히나 사용자의 수만큼의 크기를 갖는 벡터로 사용자 i의 비밀 서명키는 $gsk[i]$로 표시한다.
- $\sigma=GSig(gsk[i],m)$ (그룹 서명 알고리즘): 입력으로 그룹 공개키 gpk와 비밀 서명키 $gsk[i]$와 서명할 메시지 m을 받고 m의 서명 값 σ를 반환한다.
- $1/0=GVf(gpk,m,\sigma)$ (그룹 서명 확인 알고리즘): 입력으로 그룹 공개키 gpk와 메시지 m, 서명 값 σ를 받는다. 서명 값이 메시지 의 서명일 경우 '참(1)'을, 그렇지 않은 경우 거짓(0)을 반환한다.

○ $i/0 = Open(gmsk,m,\sigma)$ (공개 알고리즘): 입력으로 그룹 마스터키 $gmsk$와 메시지 m, 서명 값 σ를 받는다. 서명 값 σ가 메시지 m의 올바른 서명일 경우 식별자 i를 반환하고 그렇지 않다면 거짓(0)을 반환한다.

상기 알고리즘은 [BMW03]의 모델을 기준으로 기술되어 있으나 [BSZ05]의 모델과 상이하지 않아 동일하게 기술하였다. [BSZ05]의 모델에서는 공개자를 신뢰할 수도 그렇지 않을 수도 있기 때문에, 공개의 경과에 대해 추가적으로 판단의 과정이 있어 $Open$의 출력이 달라지고 $Judge$가 필요하다.

○ $(i,\tau)/0 = Open(ok,m,\sigma)$ (공개 알고리즘): 등록 테이블 reg을 가지고 있는 공개자는, 입력으로 공개 비밀키 ok와 메시지 m, 서명 값 σ를 받는다. 서명 값 σ가 메시지 m의 올바른 서명일 경우 식별자 i와 증명할 수 있는 τ를 반환하고 그렇지 않다면 거짓(0)을 반환한다.

○ $1/0 = Judge(gpk,upk[j],j,\tau,m,\sigma)$ (판단 알고리즘): 입력으로 그룹 공개키 gpk와 사용자 식별자 j와 연결된 익명 사용자 공개키 $upk[j]$와 공개 알고리즘의 출력 (i,τ)과 메시지와 서명 m,σ를 받아 공개 결과가 올바르면 참(1)을, 그렇지 않으면 거짓(0)을 반환한다.

상기 기술된 함수들은 그룹 서명 기법에서 KeyGEN(), JOIN()/ISSUE(), SIGN(), VERIFY(), OPEN()와 같은 멤버 함수를 구현함에 사용되고 각 멤버 함수들은 목적에 맞는 기능을 수행한다. 각 함수는 다음과 같은 기능을 수행 한다.

○ KeyGEN()은 그룹에서 사용할 파라미터나 키 정보를 만드는 과정이다.

○ JOIN()/ISSUE()에서는 발행자가 그룹 멤버에서 멤버십 증명서(Membership Certificate) 또는 그룹 서명키(Group Signing Key)를 발급해주는 과정이다. 일반적으로 발행자와 그룹 멤버는 상호작용하거나(Interactive)나 상호작용하지 않거나(Non- interactive)하게 그룹 멤버가 선택한 비밀에 대하여 영지

식 증명(Zero-Knowlege Proof)을 수행하며, 사용자가 선택한 비밀은 발행자가 가지고 있는 발행키(Issue Key)를 이용하여 서명된다. 최종적으로 그룹 멤버는 이러한 서명을 자신의 멤버십 증명서로 가지게 된다. 발행자는 이 과정에서 얻은 트랜잭션과 그룹 멤버의 실명을 자신의 데이터베이스에 저장하게 된다.

- SIGN()은 멤버십 증명서를 소지한 그룹 멤버의 주어진 메시지에 대한 NIZK(Non-interactive Zero-Knowledge) 증명으로 대체된다. 익명성의 철폐를 위하여, 이 과정에서 멤버십 증명서의 아이디가 ElGamal 암호화 등을 통하여 암호화되어 포함된다.
- VERIFY()는 SIGN() 과정에서 생성된 그룹 서명에 대한 NIZK 증명으로 완전히 대체될 수 있다.
- OPEN()은 그룹 서명에 포함되어 있는 멤버십 증명서의 아이디를 복호화하는 것으로 이루어질 수 있다.

이렇게 그룹 서명은 두 번의 NIZK 형태의 서명 기법과 한 번의 암호화 기법이 혼합되어 있다고 볼 수 있다. 주의해야 할 점은 각각의 기법이 그룹 서명이 갖는 보안성에 직접적으로 영향을 미치고 있다는 것이다. JOIN()/ISSUE()에서 사용되었던 서명 기법의 안전성은 추적 가능성을 성취하는데 가장 중요하며 SIGN()에 사용되는 NIZK 기반의 서명의 안전성은 무죄 입증성을 보이는데 가장 중요하다. SIGN()에 사용되는 암호화 기법의 안전성은 익명성을 보장하는데 가장 중요하다. 그룹 서명에서의 NIZK의 사용은 그룹에 포함된 사용자의 수에 비례하여 서명 길이가 증가하였던 과거 형태의 그룹 서명에서 탈피하여 새로운 형태의 그룹 서명을 만드는데 크게 기여하였다.

2. 그룹 서명 프로토콜 연구 동향

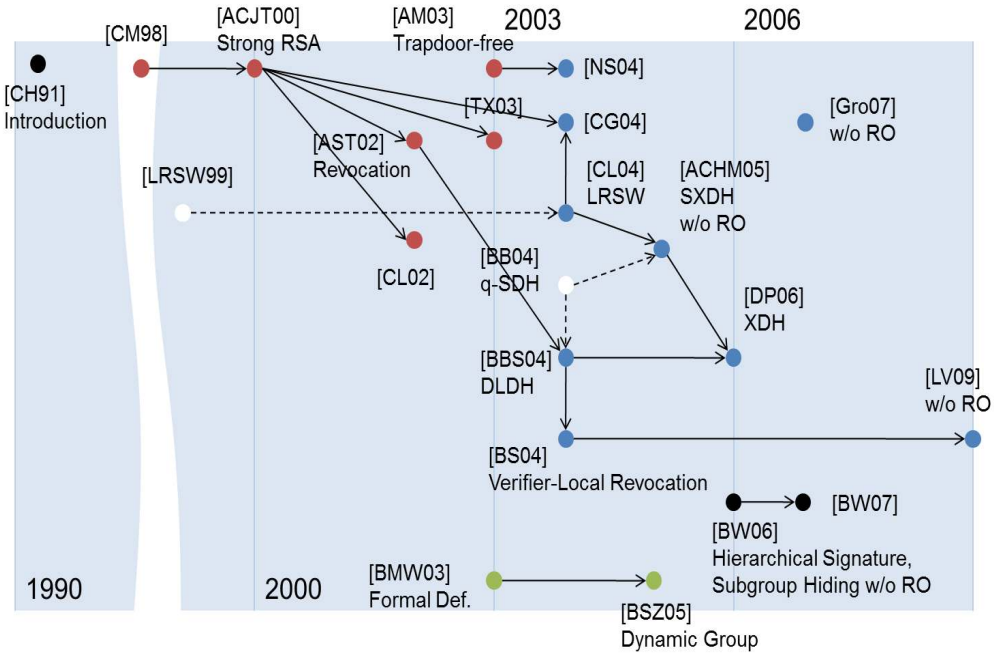
1991년 Chaum와 Heyst이 그룹 서명(Group Signature) 기법을 최초 제안 하였다

[CH91]. 이후 Camenisch와 Stadler는 CRYPTO ' 97에서 NIZKP를 이용하여 최초의 고정된 길이의 그룹 서명 기법을 개발하였다 [CS97]. 이전의 그룹 서명 기법은 그룹의 공개키와 서명의 길이가 그룹 멤버의 수에 비례하여 증가하여 서명을 확인하는데 걸리는 시간이 그룹 멤버의 수에 비례하여 증가하는 문제가 있었다. 또한 ASIACRYPT ' 98에서 이를 더 효율적으로 개선한 기법이 소개하였다[CM98]. 그러나 이러한 기법들은 그룹 멤버가 그룹 매니저에게 자신의 비밀키를 전송해야만 하는 문제가 있었다. 2000년 Ateniese 등은 CRYPTO ' 00에서 Strong RSA 가정을 기반으로 효율적이면서 동시에 Adaptive한 공격자에게도 안전한 그룹 기법을 제안하였다[ACJ00]. 이 기법은 이후에 수많은 그룹 서명 기법에 영향을 미치게 된다. CRYPTO ' 04에서 Camenisch와 Lysanskaya가 LRSW 가정[LRS99]과 겹선형 접합(bilinear pairing)을 기반으로 그룹 서명과 익명 신용장 시스템에 적합한 서명 기법을 제안하였으며[CL04] 이를 바탕으로 SCN ' 04에서 더 효율적인 그룹 서명 기법을 제안하였다[CG05]. 한편, 유사한 시기에 Boneh와 Boyen는 q-SDH 가정을 기반으로 EUROCRYPT ' 04에서 짧은 서명 기법을 제안하였으며[BB04], 이를 바탕으로 CRYPTO ' 04에서 겹선형 쌍함수를 이용하는 기존의 그룹 서명과 비교하여 더 짧은 서명 길이를 갖는 그룹 서명 기법을 제안하였다[BBH04]. Delerablee와 Pointcheval은 eXternal DH 가정을 이용하여 VIETCRYPT ' 06에서 BBS04보다 더 짧은 서명 길이를 갖으면서 IND-CCA 안전성이 증명되는 그룹 서명 기법을 제안하였다[DP06].

FC' 02에서 [ACJ00]가 가진 Revocation 문제를 지적하며, Atenise 등은 보다 효율적인 Revocation이 가능한 기법을 제안하였다[AST03]. 기존의 기법은 전체 그룹 멤버의 비밀키와 그룹 공개키 등을 모두 갱신 하는 방식으로 이루어졌으나 이 기법에서는 CRL을 기반으로 하여 모든 서명에 대하여 Revoke된 그룹 멤버가 아님을 확인하는 방식을 사용하였다. 그러나 여전히 CRL의 크기가 커지면 그룹을 재갱신하는 것이 효율적일 수 있다. 또한 Verifier-local Revocation의 기능을 가진 그룹 서명 기법이 ACM CCS ' 04에서 발표되었다[BS04].

2003년, Bellare등은 그룹 서명 기법 최초로 정확성, 간결성, 완전 익명성, 완전 추적 가능성을 내용으로 하는 그룹 서명 기법을 위한 보안 모델을 제시하였다

[BMW03]. 2005년에서는 동적 그룹(Dynamic Group)을 위하여 정확성, 익명성, 추적 가능성, 날조 불가능성을 내용으로 하는 보안 모델을 새롭게 제시하였다[BSZ05]. 표준 모델(Standard Model)에서의 IND-CCA 보안 관련하여, 2005년에 Ateniese 등은 Strong LRSW, Extended DH, Strong Symmetric eXternal DH 가정을 이용하여 표준 모델에서 IND-CCA 안전성이 증명되는 최초의 그룹 서명 기법을 제안하였다[ACH05]. Boyen과 Waters는 EUROCRYPT ' 06에서 기존의 기법과 전혀 다른 방식(Hierarchical Signature와 Subgroup Hiding)으로 만들어 표준 모델에서 CPA 익명 안전성이 증명되는 그룹 서명 기법을 제안하였다[BW06]. 이후, Boneh등의 서명기법[BBH04]과 Waters의 서명을 이용한 익명 계층(Anonymous Hierarchical) 서명을 고안하고 이에 Sahai의 영지식증명을 결합하여 일정한 그룹 서명 길이로 바꾼 기법을 PKC ' 07에서 제안하였다[BW07]. Groth는 ASIACRYPT ' 07에서 [BSZ05] 모델을 만족하면서 표준 모델에서 증명 가능한 그룹 서명 기법을 제안하였다[Gro07].



[그림 3-3] 시간의 경과에 따른 그룹 서명 기법의 발전

그룹 서명은 크게 두 번의 전환기를 맞이한다. 2000년을 전후로 그룹 서명 기법에 NIZK의 사용함으로 그룹 서명 길이를 그룹 멤버의 수와 관계없이 구성하여 서명의 길이를 크게 단축시켰다. 또 2004년을 전후로 이중 선형 쌍함수(Bilinear Pairing)의 사용으로 서명의 길이를 다시 대폭 줄이는 것이 가능했던 것이다. 최근에는 점차 랜덤 오라클 모델이 아닌 표준 모델에서 구성되는 그룹 서명 기법들이 많이 선보이고 있다.

[표 3-1]은 2000년 이후 현재까지 나온 주요 그룹 서명 기법 간의 성능을 간략히 비교하여 보여준다. 표에서 Tracing DB는 공개자가 서명을 공개하기 위해서는 반드시 멤버십 증명서의 아이디에 관련된 데이터베이스가 있어야 함을 의미한다. w/o RO은 해당 기법이 표준 모델에서 안전성이 증명됨을 의미한다.

[표 3-1] 주요 그룹 서명 간 비교

(OTS = One Time Signature, NIWI = Non-interactive Witeness Indistinguishable, P = Pairing, E = Exponentation, w/o RO = without Random Oracle)

	ACJ00	AST02	ACH05	BBS04	BS04	DP06	Gro07	BW07
가정	S-RSA, DDH		SXDH, S-LRSW, EDH, S-SXDH	q-SDH, DLDH		DDH, XDH, q-SDH	-	Subgroup Decision, HSDH
ENC	ElGamal Enc		-	Linear Enc		Double ElGamal Enc	Selective-Tag Weakly CCA-Secure Enc	-
SIG	NIZK						NIWI + NIZK + OTS	BB04 + IBE
서명 길이	8	10	8	9	7	9	6	6
성능	10E / 11E	12E / 11E+|CRL|E	|CL04|+ |BB04|	12E+1P / 12E+1P	8E+1P / 6E+ (3+2|RL|)P	11E+1P / 11E+1P	-	(2|m|+12)E / |m|E+6P
안전성 모델	Traceability, Coalition-resistance		Not BMW03	CPA-anonymity	Selfless-anonymity	BSZ05	BSZ05	CPA-anonymity

	ACJ00	AST02	ACH05	BBS04	BS04	DP06	Gro07	BW07
비고	Constant size	CRL Revocation	Tracing DB, CCA, w/o RO		VLR		w/o RO	Tracing DB, w/o RO

3. Lattice 기반 그룹 서명 프로토콜 소개

가. Lattice 등장 배경

Lattice는 18세기 Gauss, Minkowski 등에 의해 수학적 이론을 중심으로 연구되어 왔고, 최근에는 컴퓨터 응용 분야에 널리 사용되고 있다. 특히 최근에 많은 관심을 받고 있는 양자 컴퓨팅은 기존의 암호 시스템 설계의 사용되었던 인수분해 문제(factoring problem)와 이산대수 문제(discrete logarithm problem)를 쉽게 풀 수 있기 때문에[Sho94] 대체 암호 시스템에 대한 연구가 요구된다. Lattice 기반의 암호 시스템은 양자 컴퓨팅 환경에서도 쉽게 풀 수 없기 때문에 이 분야에 대한 연구가 활발히 진행되고 있다. 뿐만 아니라 Lattice 기반의 암호 시스템은 기존의 암호 시스템보다 효율적이다. 기존의 암호 시스템의 연산 복잡도는 $O(n^3)$인데 반해 Lattice 기반의 암호 시스템의 연산 복잡도는 $O(n^2)$으로 효율적이다.

나. Lattice의 정의

Lattice는 m 차원 공간에서 일정한 규칙을 가지고 분포되어 있는 점들의 집합을 의미한다. 일반적으로 Lattice는 m 차원 벡터 공간 R^m에서 주어진 일차 독립인 n개의 벡터 $b_1, b_2, ..., b_n$에 대하여 다음과 같이 정수 계수의 일차 결합이다.

$$\Lambda(b_1, b_2, ..., b_n) = \left\{ \sum_{i=1}^{n} x_i b_i \mid x_i \in \mathbb{Z} \right\}$$

여기서 n개의 일차 독립인 벡터들의 모임 $b_1, b_2, ..., b_n$을 기저(basis)라고 한다.

벡터들 $b_1, b_2, ..., b_n$을 열 벡터로 갖는 $m \times n$행렬 B로 기저를 표현하기도 하며, 이를 이용해 Lattice를 정의하면 다음과 같다.

$$\Lambda(b_1, b_2, ..., b_n) = \Lambda(B) = \{Bx \mid x \in \mathbb{Z}^n\}.$$

주어진 Lattice Λ에 대해서 Dual Lattice를 정의할 수 있다. Dual Lattice Λ^*는 Lattice Λ의 어떠한 벡터와 내적을 하더라도 정수인 벡터들의 집합이다.

$$\Lambda^* = \{y \in span(\Lambda) \mid \forall x \in \Lambda, < x, y > \in \mathbb{Z}\}$$

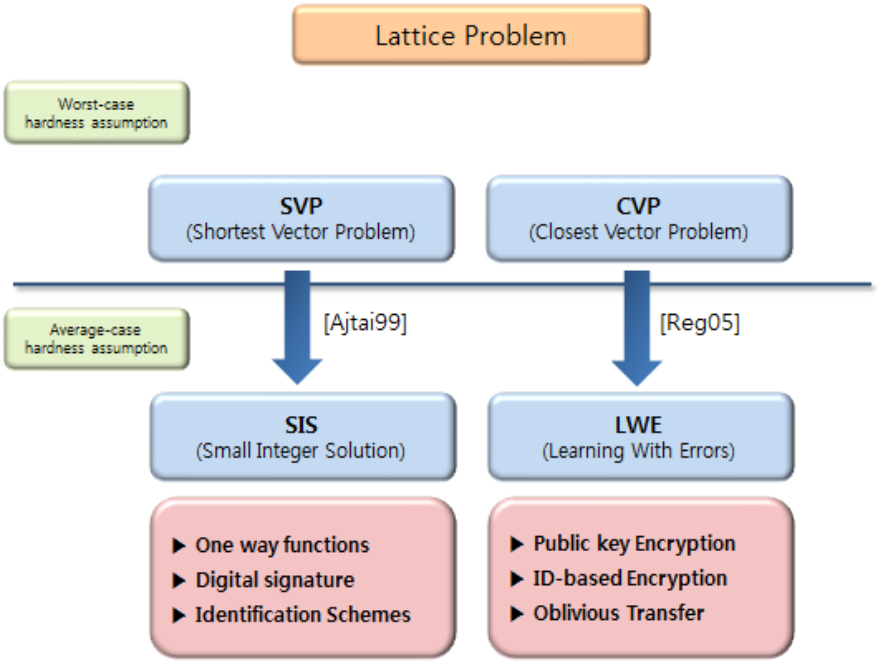
Lattice Λ의 기저 $B = (b_1, b_2, ..., b_n) \in R^{m \times n}$에 대하여 다음의 두 조건을 만족하면 $D = (d_1, d_2, ..., d_n) \in R^{m \times n}$을 Dual 기저(dual basis)라고 한다.

- $span(D) = span(B)$
- $B^T D = I$

주어진 행렬 $A \in Z_q^{n \times m}$에 대해서 $Ae = 0 \bmod q$를 만족하는 벡터들의 집합은 Lattice를 이루게 되는데, 이를 q와 관련된 Lattice(q-ary Lattice)라 하고 $\Lambda^{\perp}(A) = \{e \in \mathbb{Z}^m \mid Ae = 0 \bmod q\}$로 나타낸다. 이와 비슷한 형태로 법 q상에서 특정한 벡터 y가 되는 벡터들의 모임도 Lattice의 coset 형태를 이루게 되는데 이러한 coset은 $\Lambda_y^{\perp}(A) = \{e \in \mathbb{Z}^m \mid Ae = y \bmod q\}$로 표기한다.

다. Lattice에서의 hardness assumption

Lattice 기반 암호 시스템에서는 크게 'average-case hardness assumption'과 'worst-case hardness assumption'의 어려운 문제를 기반으로 설계 된다.



[그림 3-4] average-case 와 worst-case hardness assumption

'average-case hardness assumption'은 주어진 문제에 대해, 어떤 다항 시간 알고리즘이라 할지라도 대부분의 instance를 해결할 수 없다는 가정이다. 기존의 RSA 암호 시스템, ElGamal 암호 시스템과 같이 인수분해 문제와 이산대수 문제 기반의 암호 시스템은 이러한 문제들이 average-case hardness assumption을 만족한다는 사실을 기반으로 설계되었다.

'worst-case hardness assumption'은 주어진 문제에 대해 모든 instance를 풀 수 있는 다항 시간 알고리즘은 존재하지 않는다는 가정이다. 다시 말해, 어떠한 다항 시간 알고리즘으로도 해결되지 않는 instance가 하나만 존재해도 우리가 그것으로 안전한 암호 시스템을 설계할 수 있음을 의미한다.

1) Lattice에서의 worst-case hard problem

Lattice 기반의 전자서명이나 암호화 기법을 설계하기 위해 필요한

worst-case의 어려운 문제(hard problem)로 가장 짧은 벡터를 찾는 “SVP(shortest vector problem)”와 임의의 주어진 점에 대해서 해당 점과 가장 가까운 위치에 있는 격자점(Lattice point)을 찾는 “CVP(closest vector problem)”이 있다. SVP와 CVP는 입력 값과 출력 값에 따라 검색(search), 최적화(optimization), 결정(decision) 버전으로 나눌 수 있다.

- SVP : SVP의 버전을 나누기 위해 사용되는 값으로 Lattice에 속한 격자점들의 길이를 비교할 때 격자점 중에서 가장 짧은 길이를 λ_1(successive minima)으로 표시한다.
  - SVP Search(SVP_S) : 주어진 Lattice Λ의 기저 $B=(b_1,b_2,...,b_n)$에 대하여 길이가 가장 짧은 격자점 Bx를 찾는 문제
  - SVP Optimization(SVP_O) : 주어진 Lattice Λ의 기저 $B=(b_1,b_2,...,b_n)$에 대하여 λ_1을 구하는 문제
  - SVP Decision(SVP_D) : 주어진 Lattice Λ의 기저 $B=(b_1,b_2,...,b_n)$와 실수 r에 대하여 $r \geq \lambda_1$인지를 결정하는 문제
- CVP : CVP의 버전을 나누기 위해 사용되는 값으로 어떤 임의의 점 t와 가장 가까이 있는 Lattice Λ의 격자점까지의 거리를 $dist(t,\Lambda)$로 표시한다.
  - CVP Search(CVP_S) : 주어진 Lattice Λ의 기저 $B=(b_1,b_2,...,b_n)$와 임의의 점 t에 대하여, 점 t와 가장 가까운 Lattice Λ의 한 점 Bx를 구하는 문제
  - CVP Optimization(CVP_O) : 주어진 Lattice Λ의 기저 $B=(b_1,b_2,...,b_n)$와 임의의 점 t에 대하여, $dist(t,\Lambda)$를 구하는 문제
  - CVP Decision(CVP_D) : 주어진 Lattice Λ의 기저 $B=(b_1,b_2,...,b_n)$와 임의의 점 t, 실수 r에 대하여, $r \geq dist(t,\Lambda)$인지를 결정하는 문제

2) Lattice에서의 average-case hard problem

'SIS(small integer solution)' 문제는 [Ajt99]에서 처음으로 소개된 것으로 'ISIS(inhomogeneous small integer solution)'의 특별한 형태이다. ISIS 문제는 주어진 instance (q,A,u,β)에 대하여 $Ae = u \bmod q$과 $\| e \| \leq \beta$을 만족하는 벡터 $e(\neq 0) \in \mathbb{Z}^m$을 찾는 것이다. 여기서 $A \in \mathbb{Z}_q^{m \times n}$와 $u \in \mathbb{Z}_q^n$는 해당 집합에서 임의로 선택하고, $\beta \in \mathbb{R}$, $q(\geq 2) \in \mathbb{Z}$에서 선택하여 instance를 구성한다. SIS 문제는 ISIS 문제에서 벡터 $u = 0$으로 고정한 것으로, $Ae = 0 \bmod q$과 $\| e \| \leq \beta$을 만족하는 벡터 $e(\neq 0) \in \mathbb{Z}^m$를 찾는 것이다. 실수 β가 짧을수록 이것보다 짧은 벡터 e를 찾는 것은 어렵다[14].

'LWE(learning with errors)' 문제는 [Reg05]에서 처음 소개된 것으로 서로 다른 분포에서 추출한 값들이 구별 불가능하다는 점에 초점을 맞춘다. 일반적으로, LWE 문제는 두 개의 오라클을 사용하여 표현한다. $\mathbb{Z}_q$ ($q \geq 2$) 상의 어떤 확률 분포 χ와, 벡터 $s \in \mathbb{Z}_q^n$에 대하여, $\mathbb{Z}_q^n$ 상에서 임의로 추출한 벡터 a와 분포 χ에서 독립적으로 추출한 x를 이용하여 $(a, a^T s + x)$형태의 값을 집합 $\mathbb{Z}_q^n \times \mathbb{Z}_q$상에서 추출하는 분포를 $A_{s,\chi}$라고 하자. LWE 문제의 목표는 한 오라클은 분포 $A_{s,\chi}$에서 값을 추출하고, 다른 오라클은 $\mathbb{Z}_q^n \times \mathbb{Z}_q$ 에서 임의로 값을 추출할 때, 이 두 분포 사이에서 추출한 값을 구별하기 어렵다는 것이다. 만약 LWE 문제가 어렵다면 분포 $A_{s,\chi}$에서 추출한 값은 임의의 값으로 간주하고 사용할 수 있다[GPV08].

[Ajt99]과 [Reg05]에서는 average-case에서 정의된 SIS 문제와 LWE 문제를 사용하여 worst-case에서 정의된 문제들을 풀 수 있음을 보였다. 즉, SIS 문제와 LWE 문제의 어려움에 기반하여 설계된 서명 기법이나 암호화 기법은 결과적으로 worst-case hardness assumption에 기반을 두었다고 말할 수 있으며, 결과적으로 이렇게 설계된 서명 기법이나 암호화 기법은 양자 컴퓨팅 환경에서도 안전하다고 말할 수 있다.

라. Lattice 기반의 전자 서명 프로토콜의 연구 동향

본 소절에서는 기존의 Lattice 기반 서명 기법들의 동향을 소개한다. 먼저 서명 기법을 정리하고, 링 서명 기법과 그룹 서명 기법을 살펴본다.

1) Lattice 기반 서명 기법

Lattice를 기반으로 한 서명 기법은 Goldreich 등에 의해서 1997년에 최초로 제안(GGH 서명 기법)되었으며[GGH97], 2003년에는 NTRU 서명 기법이 제안되었다[HGP03]. GGH 서명 기법과 NTRU 서명 기법은 우선 메시지를 공간속의 하나의 점으로 표현한 후, 해시함수를 이용해서 공간속 임의의 한 점으로 이동시킨다. 그리고 Lattice의 짧은 기저를 이용해서 가까운 Lattice에 속하는 점으로 이동시킴으로써 메시지에 대한 서명을 생성한다. 두 가지 기법 모두 가장 가까운 벡터를 근사적으로 찾는 문제(approximate closet vector problem)가 어렵다는 것에 기반을 두고 있다. 하지만 이 두 기법의 초기 모델은 각각의 서명에서 서명키의 정보가 노출되기 때문에 안전하지 않다는 것이 Gentry와 Szydlo에 의해서 증명되었다[GKV10,Ruc10]. 그리고 2006년에는 GGH 서명 기법의 서명에서 서명키인 짧은 기저의 평행사변형 형태가 드러나기 때문에 서명키가 복구 가능함이 Nguyen과 Regev에 의해 증명되었다[NR06].

Micciancio와 Vadhan은 처음으로 랜덤 오라클을 이용해서 공식적으로 안전성이 증명된 효율적인 서명 기법을 제안했다[MV03]. 이후에 다양한 Lattice 기반의 어려운 문제를 활용해서 통계적인 영 지식 증명 시스템(statistical zero-knowledge proof system)이 제안되었고, 이를 이용한 효율적인 Lattice 기반 인증(identification) 기법이 제안되었다. 또한 이 인증 기법과 Fiat-Shamir의 기법을 이용해서 랜덤 오라클 기반의 안전한 서명 기법이 제안되었다.

최근 들어 Lattice 기반의 암호시스템이 활발히 연구되면서 많은 서명 기법들이 제안되었다. Gentry 등의 기법[GPV08], Cash 등의 기법[CHK10], Rückert

의 기법[Ruc10], 그리고 Boyen의 기법[Boy10]이 제안되었다. Gentry, Peikert, Vaikuntanathan은 GGH 서명 기법과 NTRU 서명 기법을 변형시킨 새로운 서명 기법을 제안했다[GPV08]. 랜덤 오라클 모델에서 최악의 경우에도 어려운 문제(worst-case hardness assumption)를 기반으로 하는 서명 기법이다. Cash, Hofheinz, Kiltz, Peikert 등에 의해 제안된 서명 기법은 SIS 문제에 기반하고 있으며 표준모델에서 증명되었다[CHK10]. 고정된 선택 메시지 공격에 대해서 위조 불가능성을 만족 한다. 메시지의 길이가 늘어남에 따라 검증키 및 서명의 크기가 늘어난다는 단점이 있다. Rückert는 선택 메시지 공격에 안전한 기법을 제안하였다[Ruc10]. 이 서명 기법은 고정된 선택 메시지 공격에 대해 강하지만 [CHK10]에서처럼 카멜레온 해시 함수를 사용해서 선택 메시지 공격에 대한 안전성을 제공할 수 있다. SIS 문제에 기반하고 있으며, 표준 모델에서 증명되었다. [CHK10]에서처럼 메시지 길이에 의존해서 검증키의 크기와 서명의 크기가 늘어난다는 단점이 있다. Boyen은 특별한 형태의 프리이미지 샘플링 함수를 이용한 짧은 서명 기법을 제안하였다[Boy10]. 이 서명 기법은 SIS 문제를 기반으로 표준 모델에서 선택 암호문 공격에 대한 안전성을 제공한다. 앞의 두 서명 기법에서 발생하는 메시지 길이에 의존해서 서명의 길이가 늘어나는 문제를 해결했지만, 검증키의 길이가 메시지 길이에 의존하는 문제는 해결하지 못했다.

최근에 제안된 Lattice 기반의 서명 기법들의 랜덤 오라클 사용 유무, 검증키/서명키의 크기, 서명의 크기, 안전성을 비교하면 다음과 같다.

[표 3-2] Lattice 기반 서명 기법들의 비교

논문	안전성 모델	검증키 크기	서명키 크기	서명 크기	강한 위조 불가능성
[GVP08]	랜덤오라클	nm	m^2	m	X
[CHK10]	표준	$2lnm$	m^2	lm	X
[Ruc10]	표준	$2lnm+m$	m^2	lm	O
[Boy10]	표준	$(2+l)nm$	m^2	$2m$	X

여기서 l은 메시지의 비트수를 나타내고, n,m은 각각 행렬 $A \in Z_q^{n \times m}$에서의 n,m을 나타낸다. [표 3-2]에서 보여주듯이 [GVP08]과 [Boy10]을 제외하고는 서명 크기에 l이 포함되어 있는데, 이는 서명의 크기가 메시지 크기에 의존한다는 것을 나타낸다. 하지만 [Boy10]에서는 이러한 문제를 해결해서 메시지 크기와 관계없이 서명의 크기가 일정하다. 또한 [GVP08]을 제외하고는 검증키 크기에 l값이 포함되어 있는데, 이것 역시 검증키의 크기가 메시지 크기에 의존함을 의미한다. 비록 [GVP08]는 검증키의 길이와 서명의 크기가 메시지 길이 l와 관계없이 일정하지만, 표준 모델이 아닌 랜덤 오라클 모델에서 증명되었기 때문에 동일 조건으로 표준 모델에서 증명 가능한 서명 기법에 관한 연구가 필요하다.

2) Lattice 기반 링 서명 기법

Lattice 기반의 링 서명 기법은 아직 많이 제안되지 않았다. 현재 제안된 기법은 Brakerski와 Kalai가 제안한 서명 기법[BK10]과 Wang이 제안한 서명 기법[Wan10] 뿐이다. Barkerski 등에 의해 처음으로 Lattice 기반의 링 서명 기법이 제안되었다[BK10]. 제안된 기법은 ISIS(inhomogenous short integer solution) 문제에 기반하고 있으며, 새롭게 정의한 링 트랩도어 함수(ring trapdoor function)를 이용해서 링 서명 기법을 설계하였다. 고정된 선택 메시

지 공격에 대해 이전에 선택했던 메시지의 위조 불가능성(a-priori message ring unforgeable under static chosen message attack)을 만족한다. Barkerski 등의 연구에 이어서 Wang은 Lattice 기반 링 서명 기법을 제안했다[Wan10]. 이 기법은 Barkerski 등의 링 트랩도어 함수와 Boyen의 기법[Boy10]을 이용해서 링 서명 기법으로 확장했으며, [BK10]에서 제안된 최초의 Lattice 기반 링 서명보다 서명의 길이가 더 짧다. 링 서명의 위조 불가능성은 SIS 문제를 기반으로 증명 가능하다.

3) Lattice 기반 그룹 서명 기법

링 서명과 마찬가지로 Lattice 기반의 그룹 서명도 아직 연구 초기 단계이다. 현재까지의 그룹 서명 기법은 GorDon, Katz, Vaikuntanathan이 제안한 기법이 유일하다[GKV10]. 이 그룹 서명 기법은 랜덤 오라클 모델에서 안전성이 증명되었으며, 주어진 Lattice의 직교하는 Lattice와 트랩도어 추출 알고리즘을 이용해서 설계되었다. 그리고 임의의 Lattice와 벡터가 주어졌을 때 가장 가까운 Lattice 상의 벡터를 드러내지 않으면서 Lattice와의 거리를 증명할 수 있는 기법, 다시 말해, 주어진 기저 B, 벡터 z, 임의의 길이 t에 대해서 효과적인 NIWI(non-interactive witness-indistinguishable) 증명 값을 생성하는 기법을 제안하고 이를 그룹 서명 설계에 이용했다.

4. 새로운 그룹 서명 프로토콜 연구의 필요성

최근 CRYPTO, EUROCRYPT, ASIACRYPT, PKC 등의 저명 학술대회에서 주목할 만한 점은 익명성이 보장되는 그룹 서명의 설계에 관한 것이었으며 특히 경량 그룹 서명의 설계에 대한 관심이 높았다. 경량화에 더불어 양자 컴퓨팅의 발전과 함께 안전성 측면에서 Lattice 관련 암호 기법의 설계에 대한 관심이 높았다. 이처럼 경량화가 필요한 환경에서의 익명성 보장형 그룹 서명 프로토콜의 개발과 Lattice 기반 암호 시스템의 개발은 전성기를 맞고 있다고 평가된다.

가. 유비쿼터스 시대에 따른 경량 암호 시스템의 필요성

유비쿼터스란 라틴어에서 유래하여 ‘곳곳에 널려 있다.’, ‘언제, 어디서나 동시에 존재한다.’ 는 의미로, 사용자가 컴퓨터나 네트워크를 의식하지 않고 장소에 상관없이 자유롭게 네트워크에 접속할 수 있는 환경을 말한다. 즉, 컴퓨터에 기능을 추가하는 개념이 아닌 시계, 의류, 자동차 등 특정 기기나 사물에 컴퓨터를 장착하여 서로 간의 통신이 가능하게 하는 패러다임이다. 하지만, 이러한 환경에서의 프라이버시 문제는 매우 치명적이다. 자신이 의식하지 못한 상태에서 네트워크에 접속할 수 있기에 언제 어디서든 자신의 민감한 정보가 노출될 수 있다.

유비쿼터스 환경 및 그에 따른 프라이버시 문제로 인해 암호 시스템의 역할은 한층 더 중요해졌고, 보다 더 가벼워지도록 요구된다. 일반적으로 암호 시스템이 사용되던 환경은 서버나 사용자의 컴퓨터로 연산능력이 좋은 곳이었다. 하지만 특정 기기 및 사물이 컴퓨터처럼 인식되는 유비쿼터스 환경에서는 암호 시스템이 구현되는 환경이 지금보다 훨씬 제한적이다. 따라서 보다 더 경량화된 암호 시스템이 필요한 실정이다.

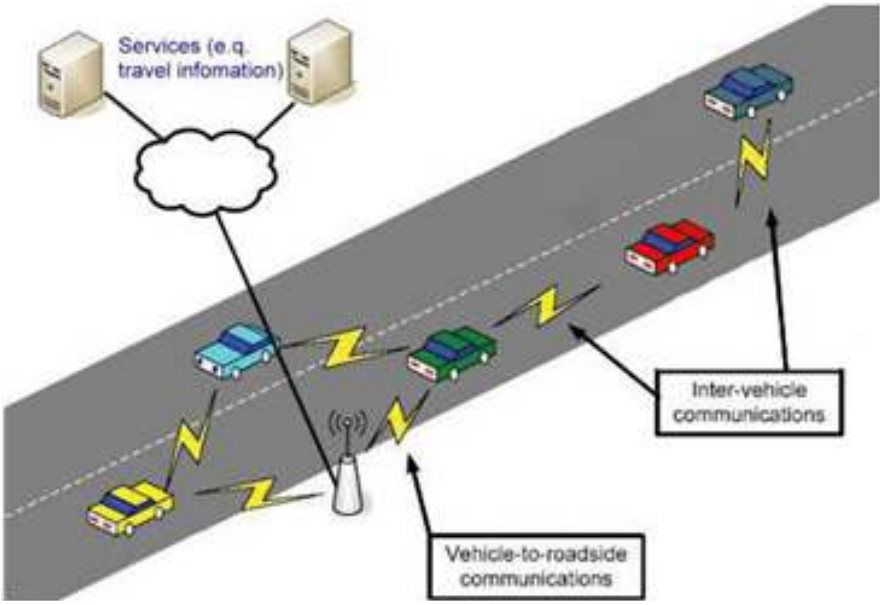
최근 설계되는 많은 암호 시스템은 pairing 연산을 사용한다. 이는 서로 독립적인 비밀 정보를 하나로 묶어주는 매우 유용한 성질을 갖기 때문에, ID 기반 암호화·전자 서명·키 교환 등 암호 시스템 전반에 걸쳐 매우 빈번하게 사용되고 있다. pairing 기반의 암호기법은 통신량의 절감이라는 효과를 제공함으로써 통신비용의 경량화를 제공한다. 따라서 무선 통신이 빈번하게 사용되고 대부분의 전자기기들이 통신을 수행하는 유비쿼터스 통신 환경에서는 이와 같이 통신 효율성을 제공하는 기법들의 개발이 매우 유용할 것이다. 반면, 연산이 비효율적이라는 단점 때문에 학계에서는 pairing의 사용 횟수를 줄인 것만으로도 중요한 연구 결과로 여겨질 정도로 pairing 연산의 비효율성을 경계하고 있다. 따라서 이러한 연산 비용상의 비효율성을 극복하기 위한 기술적인 진보가 요구되는 현실이다.

[그림 3-5] 유비쿼터스 환경

연산 효율성이 높은 암화 프리미티브의 연구는 성능이 제한된 기기를 기반으로 통신이 수행되는 유비쿼터스 환경에서의 통신을 위해 중요하다. Lattice 기반 암호 시스템의 연산은 pairing을 포함한 기존의 대수적인 연산들에 비해 매우 효율적이다. Lattice에서의 연산은 벡터 간의 덧/뺄셈, 내적 등이 대부분이기 때문에 지수 및 모듈러 연산이 주를 이루는 pairing 연산에 비해 효율적이다. 1절에서 언급했듯이 실제로도 $O(n^3)$인 pairing 기반 암호 시스템의 연산 복잡도와, $O(n^2)$인 Lattice 기반 암호 시스템의 복잡도가 이 사실을 구체적으로 뒷받침한다. 따라서 연산 능력이 제한된 모바일, 스마트카드, RFID 등에서 유용하게 사용할 수 있다.

나. VANET 환경에서 적용 가능한 그룹 서명의 필요성

VANET(vehicular ad-hoc network)은 MANET(mobile ad-hoc network)의 한 형태로 차량 간 통신 또는 차량과 노변 장치 간 통신을 제공한다.



[그림 3-6] VANET 환경

[그림 3-6]처럼 차량은 외부 인터넷과의 통신(V2R: vehicle-to-roadside unit)을 통해 실시간 교통 정보, 디지털 지도, 음악 등 상업적인 서비스를 받을 수 있다. 또는, 차량이 이동 단말기의 구실을 함으로써 차량 간 통신(V2V :vehicle-to-vehicle)을 통해 충돌 방지, 사고 경보와 같은 운전자 안전 정보 서비스를 받을 수 있다. VANET 환경은 개인의 위치 정보가 그대로 노출될 수 있는 치명적인 단점이 존재한다. 따라서 이러한 문제를 해결하기 위해 VANET 환경에서의 익명 인증은 반드시 고려되어야 한다.

그룹 서명은 익명 인증을 위한 대표적인 기본 원소이지만 다음과 같은 이유로 현재의 그룹 서명을 VANET 환경에 바로 적용하기에는 문제가 있다. VANET 환경에서 자동차는 보통 1초에 3-10번 정도의 서명 값을 발생시킨다. 만일 도시 한가운데에 있다고 가정할 때, 주변에 100대의 차량만으로 1초에 1,000회 이상의 서명 확인이 필요하다. 하지만, 현재 pairing 기반의 그룹 서명은 1초에 100회 정도의 서명 확인만 가능하다고 판단된다. 이러한 VANET 환경의 특수성을 고려해서 지금보다 경량화된 효율적인 그룹 서명이 요구된다.

특히, Lattice 기반의 암호 시스템은 앞에서 언급한 높은 계산 효율성으로 위와 같은 문제를 해결하기에 매우 적합하다. Lattice 기반의 그룹 서명은 위와 같은 환경에서 충분한 속도를 보여줄 것으로 기대된다.

다. 위치 기반 서비스에 적용 가능한 그룹 서명의 필요성

위치 기반 서비스는 위치 정보의 접속, 제공 또는 위치 정보에 의해 작용하는 모든 응용 소프트웨어 서비스로써 이동식 사용자가 그들의 지리학적 위치, 소재, 또는 알려진 존재에 대한 서비스를 받는 것으로 정의된다. 모바일을 활용한 위치 기반 서비스가 가장 대표적이며, 위치 찾기 서비스(친구 찾기, 보안 서비스), 위치 기반 정보 검색(상거래, 광고, 엔터테인먼트 서비스), 위치 기반 공공 서비스(긴급구조 서비스, 대국민 공공 서비스) 등 사회 각 분야에 걸쳐 널리 사용된다.

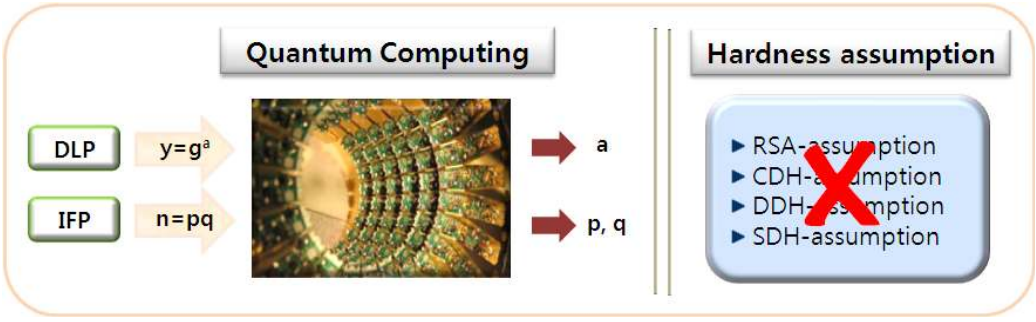
[그림 3-7] 위치 기반 서비스 예시

위치 기반 서비스는 프라이버시 문제와 밀접한 관련이 있다. 서비스에 사용될 개인의 위치 정보는 매우 민감한 정보이기 때문에 반드시 보호되어야 한다. 따라서 위치 기반 서비스에서도 익명 인증이 반드시 필요하다. 그룹 서명은 이러한 익명 인증을 수행함과 동시에 사용자의 익명성 철회도 때때로 가능하기 때문에 여러

가지 서비스를 제공함에 있어 굉장히 유용하다. 하지만, 모바일이라는 제한적인 연산 환경에서 pairing 기반의 그룹 서명은 서명 생성 과정에서만 수 초가 소요될 것으로 예상되며, 따라서 적합하지 않음을 알 수 있다. 따라서 이러한 서비스에도 경량화된 그룹 서명이 절실하게 요구되며, Lattice 기반의 그룹 서명은 서명 생성 및 확인 과정의 연산 효율성 덕분에 이와 같은 환경에 매우 적합하다.

라. 양자 컴퓨팅에 안전한 암호 시스템의 필요성

현재까지 대부분의 암호 시스템은 고전적 수학난제인 인수분해 및 이산대수 문제의 어려움에 기반하고 있다. 즉, 이러한 문제들이 어렵다는 가정 하에서 설계된 기법들이 안전하다고 여긴다. 하지만 90년대 후반 Shor가 양자 알고리즘을 이용해 이러한 수학적 난제를 해결할 수 있음을 보였고[24], 양자 컴퓨터의 개발이 점차 실현되면서 기존의 암호 시스템은 더 이상 안전성을 보장할 수 없게 되었다.

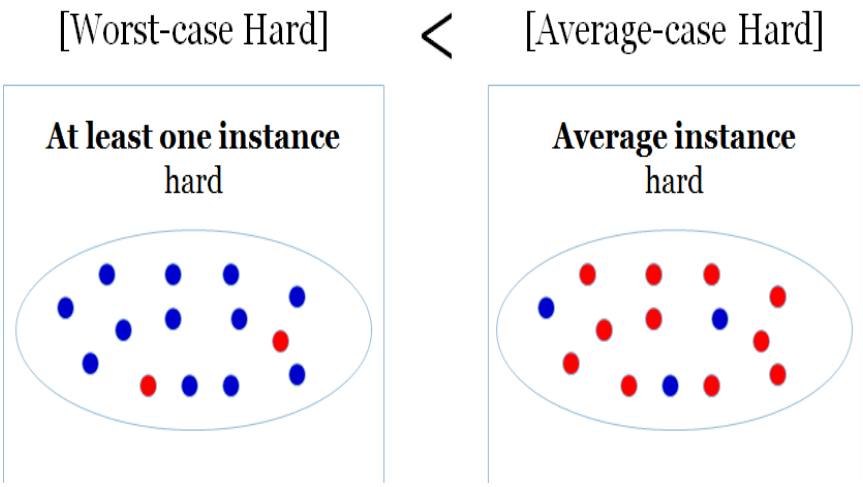


[그림 3-8] 양자 컴퓨팅을 통한 기존의 수학적 난제 해결

양자 알고리즘에 의한 공격은 단순히 암호 시스템의 키 길이만을 증가시켜 막을 수 없다. 이 공격은 기존의 난제들과는 전혀 다른 새로운 난제 기반의 암호 시스템을 필요로 한다. 양자 알고리즘에 안전한 Lattice 기반 암호 시스템은 가장 유력한 차세대 암호 시스템으로 여겨지고 있으며, 미래 지향적 측면에서 이 분야에 대한 연구는 매우 타당하다고 할 수 있다.

마. WORST-CASE 문제 기반의 암호 시스템

기존의 암호 시스템은 대부분 average-case 문제에 기반을 두고 설계된 것에 비해, Lattice 기반의 암호 시스템은 일반적으로 worst-case hard problem에 안전성 기반을 두고 설계된다.



[그림 3-9] Worst-case VS Average-case

[그림 3-9]에서는 worst-case hard와 average-case hard를 비교한다. worst-case는 적어도 하나의 풀기 어려운(hard) 경우만 존재하면 안전성이 보장되는데, average-case는 풀기 어려운 경우들이 가능한 모든 경우들 중 평균적으로 분포해야 안전성이 보장되므로 average-case hard가 더 강한 가정임을 보여주고 있다. 다시 말해, 'worst-case hardness assumption'에 기반을 둔 암호 시스템은 ' average-case hardness assumption'에 비해 좀 더 약한 가정을 기반으로 설계되었다고 할 수 있고, 따라서 worst-case 기반의 암호 시스템을 설계하는 것이 더 현실적인 안전성을 보장받을 수 있음을 의미한다.

바. 차세대 암호 시스템 원천 기술 확보를 통한 국제 경쟁력 향상

Lattice 기반 암호 시스템은 현재 연구가 가장 활발한 분야 중 하나이지만, 아직 전 세계적으로 연구 초기 단계이다. Lattice의 특성상 기초 수학 및 암호 시스템에 대한 전반적 이해가 매우 중요하기 때문에 연구 개발 초기부터 집중하여 장기적 관점에서의 투자가 필요하다. 이러한 초기 집중 투자는 세계와의 격차를 줄일 수 있음은 물론이고, 가장 중요한 원천 기술을 확보할 수 있게 한다. 원천 기술 확보는 수입된 암호 기술 대신 독자 개발한 국산 암호 기술을 사용을 통해 수입 대체 효과를 얻을 수 있을 뿐만 아니라, 특허 기술 등의 선점으로 국가 경쟁력을 향상시킬 수 있다. 또한, 개발된 기술을 국내 업체에 이전함으로써, 국내 업체의 경쟁력 강화 및 지적재산권에 대한 간접 수익 기대가 가능하다.

제 2 절 경량 그룹 서명 프로토콜 개발안

1. 그룹 서명 설계 논리

그룹 서명 기법(GSig)은 일반적으로 서명 기법(Sig), 암호화 기법(Enc)의 결합 형태이며, 추가적으로 NIZK(non-interactive zero-knowledge) 증명 또는 SoK(signature of knowledge) 증명 중의 하나를 포함하여 구성된다. 즉, 다음과 같다.

○ GSig = "Sig + Enc + NIZK or Sig + Enc + SoK"

그룹 서명 기법에서 서명 기법은 그룹 구성원의 서명을 위해 사용되고, 암호화 기법은 그룹 매니저가 그룹 서명이 누구에 의해 생성된 것인지를 추적하는데 사용된다. 그리고 NIZK 증명 또는 SoK 증명은 그룹 서명이 정당한 그룹원에 의해 서명

되었지만 그 중의 누가 서명했는지를 모르게 하는데 사용된다.

가. NIZK 증명과 SoK 증명

ZKIP(zero-knowledge interactive proof)은 P(prover)가 자신이 갖고 있는 비밀 값 x를 노출시키지 않고 V(verifier)에게 x를 알고 있음을 증명하고, V는 P가 x에 대한 정보를 알고 있음을 확인하는 기법이다. ZKIP을 증명하려면 completeness, soundness, zero-knowledgeness 세 가지 성질을 만족해야 한다. completeness는 P가 x를 알면 높은 확률로 ZKIP을 통과해야 한다는 것이고, soundness는 P가 ZKIP을 통과했다면 V는 P를 신뢰해야 한다는 성질이다. zero-knowledgeness는 ZKIP을 진행하는 도중에 제 3자(verifier 포함)에게 x에 대한 어떤 정보도 노출되지 않아야 하는 성질이다. 대부분의 그룹 서명은 ZKIP의 non-interactive 버전인 NIZK 증명을 사용하며, 때로는 NIZK 증명보다 약화된 버전인 NIWI 증명을 사용하기도 한다.

SoK(signature of knowledge) 증명은 랜덤 오라클 모델에서만 증명이 가능하다. 예를 들어 $y = g^x$라 할 때, $c = H(m \| y \| g \| g^s y^c)$를 만족하는 (c, s)쌍은 비밀 x와 임의의 r로 ($c = H(m \| y \| g \| g^r)$, $s = r - cx$)를 계산할 수 있으므로 (c, s)쌍을 x에 대한 이산 대수 SoK 증명이라고 한다.

이러한 증명 기법들은 서명 기법이나 암호화 기법에 비해 비교적 많은 연산을 요구하며, 결과적으로 그룹 서명을 효율적으로 설계하는 것에 많은 어려움을 가져다준다.

나. 기존 그룹 서명들의 구조 비교

본 항에서 우리는 lattice에 기반하는 그룹 서명 기법들이 제안되기 이전에 설계된 그룹 서명들의 구조를 비교한다. 다음의 [표 3-3]은 랜덤 오라클 모델에서 증명된 그룹 서명 기법들에 대한 구조 비교이며, [표 3-4]는 표준 모델에서 증명

된 그룹 서명 기법들에 대한 구조 비교이다.

[표 3-3] 랜덤 오라클 모델에서 증명된 그룹 서명 기법

Paper	SIG1	SIG2	Anonymity	Traceability
CS97	SIG	ENC + NIZK	ENC + NIZK	ENC
	RSA_SIG	SoK	SoK	info at join
ACJ00	StrongRSA_SIG	EIG_ENC + SoK	ENC + SoK	ENC
BBS04	BB_SIG	LIN_ENC _ SoK	ENC + SoK	ENC
CL04	CL_SIG	CS_ENC + SoK	ENC + SoK	ENC

[표 3-4] 표준 모델에서 증명된 그룹 서명 기법

Paper	SIG1	SIG2	Anonymity	Traceability	Property
BMW03	SIG	SIG + ENC + NIZK	ENC + NIZK	ENC	
ACHDM05	CL+_SIG	BB+_SIG	randomization	info at join	O(1), ANO_CCA
BW06	Wat_SIG	Wat_SIG + BGN_ENC + GOS_NIZK	ENC + NIZK	ENC	O(logN), ANO_CPA
BW07	BB+_SIG	Wat_SIG + BGN_ENC + GOS'_NIZK	ENC + NIZK	ENC	O(1), ANO_CPA

2. 기본 Lattice 관련 알고리즘

암호 기법을 설계함에 있어 사용되는 대다수의 암호 기법은 일반적으로 널리 알려져 있으므로 연산 알고리즘에 대한 설명은 배제하고 본 절에서는 현재까지 많이 사용되지 않는 Lattice와 관련된 다양한 서명 기법들에서 자주 사용되는 기본

적인 알고리즘에 대해 간략히 소개한다.

- $SampleD(B,s,c)$: $SampleD$는 randomized nearest-plane 알고리즘으로, Lattice Λ 의 기저 $B \in Z^{n \times m}$를 입력 받아, Lattice Λ 상의 이산 가우시안 분포 $D_{\Lambda,s,c}$에서 c에 가까운 격자점(Lattice point)을 샘플링한다. 여기서 s는 가우시안 파라미터(gaussian parameter)이다.

- $SampleDom(1^n)$: 치역이 랜덤한 분포를 갖도록 하는 정의역을 샘플링하는 알고리즘으로, 함수 $f_a : D_n \rightarrow R_n$의 R_n에서 $f_a(x)$가 랜덤한 분포를 갖도록 D_n의 원소 x를 샘플링한다.

- $SamplePre(t,y)$: 트랩도어를 이용하여 함수의 역상을 샘플링하는 알고리즘으로, f_a의 트랩도어 정보 t와 $y \in R_n$에 대해, $SampleDom(1^n)$을 이용하여 $f_a(x) = y$을 만족하는 x를 샘플링한다.

- $TrapGen(1^n)$: 함수와 그 함수의 트랩도어를 생성하는 알고리즘으로, 함수 $f_a : D_n \rightarrow R_n$가 효율적으로 계산되게 하는 a와 f_a의 트랩도어 정보 t를 순서쌍 (a,t)으로 출력한다.

- $GenBasis(A_1)$: Lattice의 짧은 기저를 얻기 위해 사용하는 알고리즘으로, 사용자가 선택한 행렬 $A_1 \in Z_q^{n \times m_1}$가 주어질 때, $Z_q^{n \times m}$상에서 $A = [A_1|A_2]$를 균등하게 만드는 $A_2 \in Z_p^{n \times (m - m_1)}$와 Lattice $\Lambda^{\perp}(A)$의 짧은 기저 $S \in Z^{m \times m}$를 출력한다.

- $ExtBasis(S,A')$: A를 확장시킨 A'으로 만든 Lattice $\Lambda^{\perp}(A')$의 기저 S'

를 생성하는 알고리즘으로, $\Lambda(A)=Z_q^n$인 $A\in Z_q^{n\times m}$와 Lattice $\Lambda^{\perp}(A)$의 기저 $S\in Z^{m\times m}$에 대해 임의로 $\overline{A}\in Z_q^{n\times \overline{m}}$를 선택했을 때, $A'=A\|\overline{A}$에 대한 Lattice $\Lambda^{\perp}(A')$의 기저 S'을 출력한다.

○ $RandBasis(S,s)$: 주어진 기저와 길이가 같고 동일한 Lattice를 생성하는 새로운 기저를 생성하는 알고리즘으로, Lattice Λ의 기저 $S\in Z^{m\times m}$와 파라미터 $s\geq\|\tilde{S}\|\cdot\omega(\sqrt{\log n})$로, $\Lambda(S)=\Lambda(S')$이고 $\|S\|=\|S'\|$인 새로운 기저 S'을 출력한다.

○ $TrapSamp(1^n,1^m,1^q)$: 랜덤한 $A\in Z_q^{n\times m}$와 트랩도어 $T\in Z^{m\times m}$를 출력하는 알고리즘으로, A는 $Z_q^{n\times m}$ 상에서 랜덤한 분포를 가지고, Lattice $\Lambda^{\perp}(A)$의 기저들을 T의 열 벡터로 택한다. 따라서 $AT=0(\mathrm{mod}q)$를 만족하게 되고, T의 각 열 벡터들은 길이가 $O(n\log q)$이다.

○ $OrthoSamp(1^n,1^m,1^q,B)$: 직교하는 Lattice와 그에 대응되는 트랩도어를 출력하는 알고리즘으로, 계수(rank)가 $\min(n,m)$인 $B\in Z_q^{n\times m}$(full-rank B)를 입력 받아, $AB^T=0(\mathrm{mod}q)$이고 분포가 랜덤한 $A\in Z_q^{n\times m}$와 $AT=0(\mathrm{mod}q)$인 $T\in Z^{m\times m}$를 출력한다. T의 각 행 벡터들은 길이가 $O(n\log q)$이다.

○ $GPVInvert(A,T,s,u)$: $Ae=u(\mathrm{mod}q)$인 짧은 벡터 $e\in Z^m$를 계산하는 알고리즘으로, $At=u(\mathrm{mod}q)$를 만족하는 임의의 $t\in Z^m$을 계산하고, 이산 가우시안 분포 $D_{\Lambda^{\perp},s,-t}$로 부터 v를 샘플링해서 $e=t+v$를 출력한다.

3. 기반의 경량 링 서명 기법 개발안

지금까지 제안된 Lattice 기반의 링 서명 기법[BK10, Szy03]은 모두 약한 위조 불가능성만을 만족하며, 강한 위조 불가능성에 대한 안전성 모델은 정립되지 않았다. 여기서, 약한 위조 불가능성은 알려진 평문-서명 쌍 (m,σ)을 사용하여 동일한 평문 m에 대한 새로운 서명 σ'을 위조하는 것에 대한 안전성은 보장하지 않지만, 강한 위조 불가능성은 이러한 위조에 대해서도 안전성을 보장한다. 우리는 본 절에서 강한 위조 불가능성에 대한 안전성 모델을 최초로 정립한다. 강한 위조 불가능성 모델은 우리가 아는 한 지금까지 링 서명 기법에서 정립된 위조 불가능성 모델 중에서 가장 강력한 모델이다.

가. 배경지식

본 절에서 n은 시큐리티 파라미터로 사용되며, 모든 알고리즘(공격자도 포함)에 단항으로 구성된 시큐리티 파라미터 n이 내포되었다고 가정한다. 정수 $q \geq 1$로 모듈러한 정수들의 집합은 Z_q로 나타낸다. 어떤 문자열 x에 대해 $|x|$ 는 x의 길이를 나타내고, 어떤 집합 K에 대해 $|K|$ 는 K의 원소의 개수를 나타낸다. 열벡터(column vector)는 소문자로 표시하고(예를 들면, x), 행렬(matrix)은 대문자로 표시한다(예를 들면, X). 행렬 X는 순서를 가지는 열 벡터의 집합 $\{x_j\}$이며, $X \| X'$는 집합 X와 X'의 순서를 가지는 접합을 나타낸다.

1) 링 서명의 안전성

링 서명의 알고리즘 구성 및 기본적인 정의는 위의 1장에서 언급되어 있으므로 생략한다. 본 절에서는 링 서명 기법을 논할 때 추가적으로 필요한 안전성에 관해 기술한다.

기본적으로 링 서명은 익명성과 위조 불가능성에 대해 안전해야 하며, 본 절에서 제안하는 링 서명 기법은 특별히 강한 위조 불가능성에 대해서도 안전

하다. 익명성 모델과 강한 위조 불가능성 모델은 다음과 같다.

가) 익명성 모델

익명성 모델은 Adam Bender 등이 정립한 anonymity against full key exposure 모델을 그대로 따른다[BKM06]. 여기서 anonymity against full key exposure 모델은 Adam Bender 등이 정립한 익명성 모델 중에서 가장 강력한 모델이다.

링 서명 기법 $RS=\{R.Gen,R.Sign,R.Vrfy\}$에 대한 익명성은 챌린저와 공격자 A 사이의 다음과 같은 게임으로 정의된다.

- $Setup$: 챌린저는 $R. \geq n$ 알고리즘을 l번 수행하여 서명키/검증키 쌍 $(sk_1,vk_1),\cdots,(sk_l,vk_l)$을 얻는다. 그리고 챌린저는 각각의 키 쌍을 생성할 때 사용했던 랜덤 코인(random coin) ω_i을 기록한다. 여기서 l은 게임 파라미터이다. 그리고 챌린저는 공격자 A에게 검증키 집합 $\{vk_1,\cdots,vk_l\}$을 보낸다.
- $Signing\ Queries$: 공격자 A는 $Signing(\bullet,\bullet,\bullet)$ 오라클이 주어지며, 입력의 형태는 (i,r,m)이다. 여기서 r은 검증키들의 집합, i는 $vk_i \in r$를 만족하는 인덱스이며, m는 서명하고자 하는 메시지이다. 그리고 r에 포함된 다른 검증키들이 $Setup$ 단계에서 챌린저에게 받은 검증키 집합 $\{vk_1,\cdots,vk_l\}$에 포함될 필요는 없다. 챌린저는 $Signing(i,r,m)$ 오라클에 대한 서명 σ를 공격자 A에게 보낸다.
- $Corruption\ Queries$: 공격자 A는 $Crpt(\bullet)$ 오라클이 주어지며, 입력의 형태는 (i)이다. 여기서 $i \in \{1,\cdots,l\}$이다. 챌린저는 $Crpt(i)$ 오라클에 대한 랜덤 코인 ω_i를 공격자 A에게 보낸다.
- $Challenge$: 공격자 A는 챌린저에게 챌린지 (i_0,i_1,r,m)를 보낸다. 여기서 i_0와 i_1은 $vk_{i_0},vk_{i_1} \in r$를 만족하는 인덱스이고, r에 포함된 다른 검증

키들이 $Setup$ 단계에서 챌린저에게 받은 검증키 집합 $\{vk_1,\cdots,vk_l\}$에 포함될 필요는 없다. 챌린저는 $\{0,1\}$ 중에서 랜덤하게 하나의 비트 b를 선택하고, $\sigma_b \leftarrow R.Sign(sk_{i_b}, r, m)$를 계산하여 공격자 A에게 보낸다.

- $Output$: 공격자 A는 최종적으로 b에 대한 b'을 추측하고, 만약 $b = b'$이면 공격자 A는 게임에서 승리한다. 여기서 우리는 $Crpt$ 오라클에 질의한 집합 C에 대해 $|\{i_0,i_1\} \cap C| = 2$ 를 허락한다.

우리는 위의 게임에서 공격자 A의 1/2보다 큰 이점(advantage)을 $Adv_{RS,A}^{anonymity}$라고 정의한다.

나) 강한 위조 불가능성 모델

강한 위조 불가능성 모델은 Adam Bender 등이 정립한 unforgeability w.r.t. insider corruption 모델의 구조를 그대로 따르면서 $Signing$ 오라클에 질의한 것을 위조할 수 없다는 제약사항을 제거하였다[BKM06]. 여기서 unforgeability w.r.t. insider corruption 모델은 Adam Bender 등이 정립한 위조 불가능성 모델 중에서 가장 강력한 모델이다.

링 서명 기법 $RS = \{R.\geq n, R.Sign, R.Vrfy\}$에 대한 강한 위조 불가능성은 챌린저와 공격자 A 사이의 다음과 같은 게임으로 정의된다.

- $Setup$: 챌린저는 $R.\geq n$ 알고리즘을 l번 수행하여 서명키/검증키 쌍 $(sk_1,vk_1),\cdots,(sk_l,vk_l)$을 얻는다. 여기서 l은 게임 파라미터이다. 그리고 챌린저는 공격자 A에게 검증키 집합 $\{vk_1,\cdots,vk_l\}$을 보낸다. 챌린저는 커럽된 사용자(corrupted user)의 집합 C를 $C \leftarrow \varnothing$ 으로 초기화한다.
- $Signing\ Queries$: 공격자 A는 $Signing(\bullet,\bullet,\bullet)$ 오라클이 주어지며, 입력의 형태는 (i,r,m)이다. 여기서 r은 검증키들의 집합, i는 $vk_i \in r$를 만족하는 인덱스이며, m는 서명하고자 하는 메시지이다. 그리고 r에 포함

된 다른 검증키들이 $Setup$ 단계에서 챌린저에게 받은 검증키 집합 $\{vk_1,\cdots,vk_l\}$에 포함될 필요는 없다. 챌린저는 $Signing(i,r,m)$ 오라클에 대한 서명 σ를 공격자 A에게 보낸다.

- $Corruption\ Queries$: 공격자 A는 $Crpt(\bullet)$ 오라클이 주어지며, 입력의 형태는 (i)이다. 여기서 $i\in\{1,\cdots,l\}$이다. 챌린저는 $Crpt(i)$ 오라클에 대한 서명키 sk_i를 공격자 A에게 보내고 커럽된 사용자의 집합 C에 추가한다.
- $Output$: 공격자 A는 최종적으로 튜플 (r^*,m^*,σ^*)을 출력하고, 만약 $R.Vrfy(r^*,m^*,\sigma^*)$이고, $(r^*,m^*,\sigma^*)\notin\{(r_1,m_1,\sigma_1),\cdots,(r_{q_s},m_{q_s},\sigma_{q_s})\}$이며, $r^*\subseteq\{vk_1,\cdots,vk_l\}\setminus C$이면 공격자 A는 게임에서 승리한다. 여기서 q_s는 공격자 A가 $Signing(\bullet,\bullet,\bullet)$ 오라클을 사용하는 횟수이다.

우리는 위의 게임에서 공격자 A가 게임에서 승리할 확률을 $Adv_{RS,A}^{su-full}$라고 정의한다.

나. 경량 링 서명 기법

제안하는 강한 위조 불가능성을 만족하는 링 서명은 2장에서 설명한 $SampleD$, $\geq nBasis$, $ExtBasis$ 알고리즘을 사용하여 구성된다. 먼저 신뢰할 수 있는 $setup$ authority는 다음과 같은 $Global\ Setup$ 알고리즘을 수행하여 제안하는 기법에서 사용할 추가적인 파라미터들을 생성한다.

○ $Global\ Setup$: 신뢰할 수 있는 링 서명 $setup$ authority는 다음의 파라미터들을 생성한다.

- 차원 $m=O(n\lg q)$
- 바운드(bound) $\tilde{L}=O(\sqrt{n\lg q})$

- 해시된 메시지의 길이 $|\mu|$, 이것은 링 서명의 차원이 $m' = m \cdot \max(|r|,|\mu|)$ 임을 말한다. 여기서 $|r|$ 은 링 r에 포함된 참여자들의 수를 의미한다. 제안하는 기법에서는 충돌성-저항(collision-resistant) 해시 함수 $h(\cdot,\cdot): \{0,1\}^* \times \{0,1\}^* \rightarrow \{0,1\}^{|\mu|}$ 를 사용한다.
- 가우시안 파라미터 $s = \tilde{L} \cdot \omega(\sqrt{\log m'})$
- 공개 파라미터 $params = \left\{B_1^{(0)}, B_1^{(1)}, \cdots, B_{|\mu|}^{(0)}, B_{|\mu|}^{(1)}, y\right\}$, 여기서 $B_j^{(b)} \in Z_q^{n \times m}$는 균등하게 랜덤하고(uniformly random) 서로 독립적인(independent) $2|\mu|$ 개의 $n \times m$ 행렬이고, $y \in Z_q^n$는 균등하게 랜덤한 $n \times 1$ 열 벡터이다.

각각의 사용자는 *Global Setup* 알고리즘을 통해 생성한 공개 파라미터들을 사용하여 링 서명 기법 $RS = \{R.Gen, R.Sign, R.Vrfy\}$를 다음과 같이 구성한다.

○ $R.Gen$: i번째 사용자는 $GenBasis(1^m, 1^n, q)$ 알고리즘을 두 번씩 수행하여 $A_i^{(0)} \in Z_q^{n \times m}$, $A_i^{(1)} \in Z_q^{n \times m}$과 $S_i^{(0)} \in Z_q^{m \times m}$, $S_i^{(1)} \in Z_q^{m \times m}$을 얻는다. 여기서 $S_i^{(0)}$는 $\Lambda^{\perp}\left(A_i^{(0)}\right)$의 짧은 기저($\| \widetilde{S_i^{(0)}} \| \leq \tilde{L}$)이고, $S_i^{(1)}$는 $\Lambda^{\perp}\left(A_i^{(1)}\right)$의 짧은 기저($\| \widetilde{S_i^{(1)}} \| \leq \tilde{L}$)이다. 결과적으로 i번째 사용자의 서명키는 $sk_i = \left\{S_i^{(0)}, S_i^{(1)}\right\}$이고, 검증키는 $vk_i = \left\{A_i^{(0)}, A_i^{(1)}\right\}$이다.

○ $R.Sign(sk_i, r, m)$: $R.Sign$ 알고리즘의 입력으로 서명키 $sk_i = \left\{S_i^{(0)}, S_i^{(1)}\right\}$, 링 $r = \left\{vk_1, \cdots, vk_{|r|}\right\}$, 메시지 $m \in \{0,1\}^*$를 받는다. 여기서 $i \in \{1, \cdots, |r|\}$ 이다. 랜덤한 값 $\gamma \in \{0,1\}^*$을 선택하고, $\mu = h(m, \gamma) = \mu_1 \| \cdots \| \mu_{|\mu|}$ 을 계산한다. 그리고 $|\mu|$ 와 $|r|$ 의 길이의 차이에 따라 다음과 같은 세 가지 경우를 고려하여 행렬 A를 계산한다.

- $|\mu| = |r|$ 인 경우, $A = A_1^{(\mu_1)} \| \cdots \| A_{|\mu|}^{(\mu_{|\mu|})} \in Z_q^{n \times m'}$ 이다.

- $|\mu| > |r|$ 이면, $A = A_1^{(\mu_1)} \| \cdots \| A_{|r|}^{(\mu_{|r|})} \| B_1^{(\mu_{|r+1|})} \| \cdots \| B_{|\mu|-|r|}^{(\mu_{|\mu|})} \in Z_q^{n \times m'}$ 이다.
- $|\mu| < |r|$ 인 경우, $A = A_1^{(\mu_1)} \| \cdots \| A_{|r|}^{(\mu_{|r| \bmod |\mu|+1})} \in Z_q^{n \times m'}$ 이다. 여기서 $j \in \{1, \cdots, |\mu|\}$ 는 임의의 값이다. 쉽게 말하면, μ의 마지막 값 $\mu_{|r|}$ 까지 링 r의 검증키 값들을 순차적으로 반복하여 A를 구성한다. 위와 같이 구성된 A를 사용하여 다음과 같이 v를 계산한다.

$$v \leftarrow SampleD\left(ExtBasis\left(S_i^{(\mu_i)}, A\right), y, s\right)$$

결과적으로 메시지 m과 링 r에 대한 서명은 $\sigma = (v, \gamma)$이다.

○ $R.Vrfy(r, m, \sigma)$: $R.Vrfy$ 알고리즘의 입력으로 링 r, 메시지 m, 서명 $\sigma = (v, \gamma)$를 받고, $\mu = h(m, \gamma)$를 계산한다. 그리고 위의 $R.Sign$ 알고리즘에서 행렬 A를 계산하는 방법과 동일하게 행렬 A를 계산한다. 만약 $\| v \| \leq s\sqrt{m'}$이고 $Av = y$이면 1을 출력하고, 그렇지 않으면 0을 출력한다. 제안하는 링 서명 기법 $RS = \{R. \geq n, R.Sign, R.Vrfy\}$의 정확성은 다음과 같다. 링 r의 검증키 중에 서명키를 아는 사람만 $ExtBasis$ 알고리즘을 통해 행렬 A의 짧은 기저를 계산할 수 있고, 짧은 기저를 아는 사람만 $SampleD$ 알고리즘을 통해 $\| v \| \leq s\sqrt{m'}$를 만족하는 v를 추출할 수 있다. 이렇게 구한 v는 가우시안 분포 $D_{\Lambda_y^{\perp}(A), s}$를 따르며, 즉 $y \equiv Av \bmod q$이다.

4. Lattice 기반의 경량 그룹 서명 기법 개발안

[GKV10]은 현재까지 제안된 유일한 Lattice 기반의 그룹 서명 기법이다. 하지만, 서명의 길이 및 공개키가 그룹원의 수에 비례해서 늘어나는 단점이 있다. 따라서 우리는 이러한 단점을 보완하기 위해, 서명의 길이 및 공개키의 길이가 그룹원의 수에 상관없이 상수 길이를 갖는 효율적인 그룹 서명을 제안한다. 제안한 기법은 [GKV10]에서와 같이 랜덤 오라클 모델에서 Full-익명성과 Full-추적가능성의

안전성을 만족한다.

가. 배경 지식

1) NIZK 증명

[MV03]에서는 임의의 Lattice 상에서 주어진 벡터에 가장 가까운 격자점을 숨기면서, 둘 사이의 거리를 증명하는 기법이 제안되었다. 주어진 Lattice의 기저 $B \in Z_q^{n \times m}$, 임의의 벡터 $z \in Z^m$, 임의의 길이 $t \in R$에 대해 다음과 같은 언어 L이 정의된다.

$$L = \{(B,z,t) | \exists s : \|z - B^T s\| \leq t\}$$

[MV03]에서는 주어진 (B,z,t)에 대해 효율적으로 ZKIP 값 π(L에 대해)을 생성하는 기법이 제안되었다. V(verifier)는 생성된 증명 값 π을 확인하여 s값을 모르는 채로 $(B,z,t) \in L$인지 확인 수 있다.

이러한 ZKIP은 [CDS94]에서 제안된 기법을 사용해 s값을 모르는 사용자가 매우 희박한 확률로 정당한 π을 생성하게 할 수 있다. 또한, [FS87]에서 제안된 기법을 사용해서 interactive한 증명 방법을 non-interactive하게 바꿀 수 있다. 따라서 P(prover)는 주어진 (B,z,t)에 대해 증명 값 π(L에 대해)을 한 번 생성해서 보내는 것만으로, 받는 사람에게 Lattice $\Lambda(B)$와 벡터 z 간의 거리가 t 이하인 것과 비밀 값 s를 알고 있음을 증명할 수 있다. 앞으로 (B,z,t)에 대한 π 값의 생성은 다음과 같이 표현한다: $\pi = proof\,of\,(B,z,t)$.

2) 짧은 열 벡터를 갖는 특별한 행렬의 분포

본 절에서는 특별한 분포를 갖는 행렬을 소개한다. 이 행렬은 각각의 열이 짧은 벡터로 이루어져 있기 때문에, 그룹 서명을 설계할 때 매우 유용하다. 생성하는 방법은 다음과 같다.

① $s = \sqrt{n \log q}\, w(\sqrt{\log m})$에 대해 다음과 같이 서로 독립인 m개의 벡터를

생성한다: $r_i \in Z^m \leftarrow D_{Z^m, s}\ (1 \le i \le m)$.

② 만일 $R = [r_1, r_2, ..., r_m] \in Z^{m \times m}$이 역행렬이 존재하지 않는다면 ①을 반복한다.

이렇게 생성된 행렬 R의 분포를 $D_{m \times m}(s)$으로 정의한다. 위의 생성 과정 때문에 $|R| \le s\sqrt{m}$을 만족하고, 역행렬 R^{-1}이 존재한다.

3) 그룹 서명의 안전성

가) Full-익명성 모델

그룹 서명은 서명자의 익명성을 완전히 보장해야 한다. 이를 증명하기 위해서는 다음과 같은 공격 모델에서 안전해야 한다.

① 공격자 A에게 그룹 검증키 VK와 그룹 구성원들의 모든 서명키로 구성된 N차원 벡터 $\vec{gsk}$를 제공한다.
② 공격자 A에게 서명자를 추적할 수 있는 오라클을 제공한다.
③ 공격자는 자유롭게 모든 사람의 그룹 서명을 생성하거나 주어진 그룹 서명의 서명자를 추적할 수 있다.
④ 공격자는 자신이 공격할 그룹 멤버 2명의 아이디를 선택한다.
⑤ 외부에서 두 아이디 중 하나로 만든 그룹 서명을 공격자에게 준다.
⑥ 공격자는 받은 그룹 서명에 대해 오라클을 사용하지 않고 서명자를 추측해 내놓는다.

위와 같은 공격모델에서 공격자 A가 맞출 확률이 거의 1/2에 근사하면, 이 모델에 사용된 그룹 서명은 Full-익명성을 만족한다.

나) Full-추적가능성 모델

정당한 그룹 서명은 그룹 매니저가 서명한 그룹 구성원을 정확히 추적할 수 있어야 한다. 이를 증명하기 위해서는 다음과 같은 공격 모델에서 안전해야 한다.

① 공격자 A에게 그룹 검증키 VK와 추적키 TK, 그리고 공격자가 선택한 그룹 구성원들 중 일부의 서명키 $gsk[i]$를 제공한다. 이는 공격자가 마음대로 서명자를 추적하게 하고, 자신이 선택한 그룹 구성원의 아이디로 그룹 서명을 할 수 있게 한다.

② 공격자 A에게 임의의 아이디로 그룹 서명을 받을 수 있는 오라클을 제공한다.

③ 공격자 A는 알고 있는 $gsk[i]$로 자유롭게 그룹 서명을 생성하거나 $gsk[i]$를 모르는 구성원에 대해서도 오라클을 통해 그룹 서명을 생성할 수 있다.

④ 공격자 A는 메시지와 그것의 그룹 서명을 내놓는다. 공격자가 내놓은 그룹 서명이 다음을 만족하면 공격자는 위의 공격에서 이긴다.

- TK로 서명자를 추적할 수 없는 서명인 경우
- 서명자를 찾았으나, 공격자가 그 구성원의 서명키를 모르면서 동시에 해당 그룹 서명을 오라클에 묻지 않은 경우

위와 같은 공격모델에서 공격자 A가 맞출 확률이 거의 0에 근접할 때, 우리는 그룹 서명이 Full-추적가능성을 만족한다고 말한다.

나. 경량 그룹 서명 기법

본 절에서는 [FS87]의 단점을 보완한 효율적인 그룹 서명 기법을 제안한다. 이 기법은 서명의 길이 및 공개키의 길이가 그룹 구성원의 수에 영향 받지 않고 상수

길이를 갖도록 설계되었다. 그룹 서명에 사용될 변수를 다음과 같이 정의한다.

- $n = \lambda$
- $m = n^{1+C}, (C > 0)$
- $L = \sqrt{m}\, w(\sqrt{\log m})$
- $s_0 = \sqrt{m}\, w(\sqrt{\log m})^2$
- $s_1 = \sqrt{2}\, m\, w(\sqrt{\log m})^3$

○ $G. \geq n(1^{\lambda}, 1^{N})$: 먼저 해시 함수 $H: \{0,1\}^{*} \rightarrow Z_q^n$를 임의로 선택한다. 그룹 관리자는 임의로 생성한 행렬 D_1로 $GenBasis(D_1)$을 수행하여 임의의 행렬 D와 Lattice $\Lambda^{\perp}(D)$의 짧은 기저 T_D를 생성한다.

$$(D, T_D) \leftarrow GenBasis(D_1),\ D \in \mathbb{Z}_q^{n \times m},\ T_D \in \mathbb{Z}^{m \times m},\ |T_D| \leq L$$

그 후에 $OrthoSamp(1^n, 1^m, 1^q, D)$을 이용하여 $AD^T = 0$인 행렬 A와 Lattice $\Lambda^{\perp}(A)$의 짧은 기저 T_A를 생성한다.

$$(A, T_A) \leftarrow OrthoSamp(1^n, 1^m, 1^q, D),\ A \in \mathbb{Z}_q^{n \times m},\ T_A \in \mathbb{Z}^{m \times m},\ |T_A| \leq L$$

마지막으로 N명의 그룹 구성원들에게 서명키를 발급하기 위해 짧은 열벡터로 구성된 행렬 R_i을 다음과 같이 선택한다.

$$R_i \leftarrow D_{m \times m}(s_0),\ |R| \leq s_0 \sqrt{m},\ 1 \leq i \leq N.$$

행렬 U_i와 Lattice $\Lambda^{\perp}(U_i)$의 짧은 기저 T_{U_i}를 다음과 같이 생성한다.

$$U_i = A \| AR_i,\ U_i \in \mathbb{Z}^{n \times 2m},$$

$T_{U_i} \leftarrow RandBasis(Extbasis(T_A, U_i), s_0),\ T_{U_i} \in Z^{2m \times 2m},\ |T_{U_i}| = s_0 \sqrt{2m}$.

위의 과정을 통해 생성된 키는 다음과 같다.

- 검증키 : $VK = (A, D, H)$
- i번째 그룹 구성원 서명키 : $gsk[i] = (U_i, R_i, T_{U_i}),\ 1 \le i \le N$
- 추적키: $TK = (T_D)$

○ $G.Sign(gsk[i], M)$: 사용자 i는 메시지 M에 대한 그룹 서명을 생성하기 위해, 난수 $r \leftarrow \{0,1\}^*$로 $U_i x = H(M \| r)$이고 길이가 짧은 벡터 $x \in Z^{2m}$를 다음과 같이 생성한다.

$$x \in Z^{2m} \leftarrow SampleD(T_{U_i}, H(M \| r), s_1),\ |x| \le s_1 \sqrt{2m}\,.$$

벡터 x는 $x_1 \| x_2\ (x_i \in \mathbb{Z}^m)$ 로 나누어 생각할 수 있고, 이러한 벡터 x로 다음과 같이 그룹 서명을 생성한다.

$$s_1, s_2 \leftarrow Z^n,\ \sigma_1 = D^T s_1 + x_1\ ,\ \sigma_2 = D^T s_2 + R_i x_2\,,$$

$$\pi_1 = proof\ of\ (D, \sigma_1, s_1 \sqrt{2m})\ ,\ \pi_2 = proof\ of\ (D, \sigma_2, \sqrt{2}\, m s_0 s_1)\ .$$

생성된 벡터 x가 서명에 바로 사용되면 익명성을 보장할 수 없기 때문에 x_1을 암호화해서 σ_1으로 하고, x_2와 R_i을 암호화해서 σ_2로 한다. σ_1, σ_2에 정당한 서명 값이 더해져 있음을 증명하기 위해, 증명 값 π_1, π_2을 함께 생성한다. 이는 σ_1, σ_2에 Lattice $\Lambda(D^T)$와 짧은 벡터가 더해진 형태라는 것을 증명한다. 최종적인 그룹 서명 σ은 다음과 같다: $\sigma = (M, r, \sigma_1, \sigma_2, \pi_1, \pi_2)$.

○ $G.Vrfy(VK,M,\sigma)$: 서명 확인자는 먼저 증명 값 π_1,π_2을 통해 σ_1,σ_2이 Lattice $\Lambda(D^T)$ 상의 벡터와 짧은 벡터가 더해져서 생성되었음을 검증한다. 그리고 다음의 과정을 통해 해당 그룹 서명을 검증한다.

$$A\sigma_1 + A\sigma_2 = H(M\|r)$$

정당하게 서명 값이 생성되었다면 다음과 같은 이유로 검증될 수 있다.

$$A\sigma_1 + A\sigma_2 = Ax_1 + AR_ix_2 = [A\|AR_i]x = U_ix = H(M\|r).$$

○ $G.Open(TK,M,\sigma)$: 그룹 관리자의 추적키 T_D를 사용하여 서명자를 추적한다. R_ix_2와 x_1와 x_2를 다음과 같이 복호화한다.

$$(T_D)^T\sigma_1 = (T_D)^T(D^Ts_1 + x_1) = (T_D)^Tx_1 \rightarrow (T_D^T)^{-1}(T_D)^Tx_1 = a,$$
$$(T_D)^T\sigma_2 = (T_D)^T(D^Ts_2 + R_ix_2) = (T_D)^TR_ix_2 \rightarrow (T_D^T)^{-1}(T_D)^TR_ix_2 = R_ix_2 = b\ .$$

그룹 관리자는 각 그룹 구성원에게 발급한 R_i 값을 알고 있기 때문에 $(R_i)^{-1}$를 계산하여 a,b에 곱한다. 만약 다른 구성원의 값 $(R_j)^{-1}(i \neq j)$을 곱하면 $(R_j)^{-1}R_ix_2$와 $(R_j)^{-1}x_1$이 되어 길이가 긴 벡터가 된다. 하지만 정확히 i번째 구성원의 서명키에 대한 $(R_i)^{-1}$를 곱하면 $(R_i)^{-1}R_ix_2 = x_2$와 $(R_i)^{-1}x_1$이 되어 둘 중 하나는 길이가 매우 짧은 벡터가 된다. 이를 통해 관리자는 서명자 i를 찾을 수 있다.

5. 연결불능성을 제공하는 인증 프로토콜 개발안

모바일 사용자의 프라이버시를 보호하기 위해 Lee 등이 제안한 프로토콜은 익

명성이라는 명칭으로도 알려진 사용자 신원정보 프라이버시를 제공하도록 설계되었다 [LCH09]. Lee 등의 프로토콜은 설계된 의도에 맞게 사용자 신원정보 프라이버시는 제공하지만 연결불능성 (unlinkability)을 제공할 수 없음을 보이고 이로 인해 모바일 사용자의 프라이버시가 보호받을 수 없음을 보인다. 본 절에서는 연결불능성을 제공하는 위임기반 인증 프로토콜을 기술한다. 이는 Lee 등에 의해 제안된 위임기반 인증 프로토콜에 연결불능성을 추가로 제공하도록 구성한 것으로 안전하면서 효율적으로 사용자의 프라이버시를 보호할 수 있도록 로밍 서비스를 제공함에 사용될 수 있다.

가. 배경 지식

안전한 로밍 서비스를 위해서는 다양한 특성들이 제공되어야 한다. 이러한 다양한 보안 특성들을 제공하기 위해 DES나 AES와 같은 대칭키 암호 또는 RSA나 ECC와 같은 공개키 암호와 같은 암호화 알고리즘들이 사용된다. 기존에 제안된 기법들 중에서 대부분의 프로토콜은 대칭키를 기반으로 설계되어 있다. 이는 모바일 장비의 연산 및 전력이 유산 장비들에 비하여 매우 제한되어 있기 때문이다. 그러나 개별 암호화 알고리즘의 특성에 의해 대칭키 기반의 기법들은 부인봉쇄라는 특성을 제공할 수 없다. 따라서 부인봉쇄가 반드시 필요한 응용 환경에서 로밍 서비스를 제공하기 위해서는 공개키 기반의 프로토콜이 사용되어야 한다. 로밍 서비스 이용 내역에 따라 요금이 부과되는 형태의 서비스의 경우에는 서비스 수혜자인 모바일 사용자가 로밍 서비스를 제공받지 않았다는 주장을 하지 못하도록 하기 위해 부인봉쇄 특성이 반드시 필요하다.

공개키 암호를 기반으로 로밍 서비스를 제공하는 경우 암호 알고리즘의 기본이 되는 수학 연산들의 비용이 매우 큰 문제로 작용한다. 실제로 공개키 암호에서 수행하는 기본 수학을 계산하는 것은 대칭키 암호에 비해 수천 배의 비용이 사용된다. 그러나 근래에 이르러 모바일 장비들이 기존에 비해 높은 컴퓨팅 능력을 지니게 되어 모바일 장비에서도 공개키 암호를 사용하는 것이 가능해지고 있으며 향후

더욱 개선될 것이다. 따라서 연산량에 의한 제약은 크지 않은 것으로 판단된다. 오히려 큰 문제로 고려되는 것이 PKI (public key infrastructure)의 필요성이다. PKI를 사용해야 하는 공개키 암호화를 고려하는 경우 인증서의 관리에 관련된 통신 및 연산 비용이 매우 크다. 따라서 PKI와 연관된 비용을 최소화 하는 것이 매우 중요하다. 부인봉쇄 특성을 매우 적은 비용으로 제공하기 위해 Lee와 Yeh는 무선 통신 시스템에 인증서 검증을 간소화하기 위한 일종의 위임이라는 개념을 도입하여 위임기반의 인증 프로토콜을 제안하였다 [LY05]. 그러나 최근 Lee 등은 Lee와 Yeh에 의해 제안된 프로토콜이 오프라인 인증 과정에서 부인봉쇄 특성을 제공할 수 없음을 보였다 [LCH09]. Lee 등은 자신들이 제안한 공격에 안전한 개선된 프로토콜을 제안하였다 [LCH09].

나. 연결불능성을 제공하는 인증 프로토콜

1) 초기 설정 단계

p, q는 $q|(p-1)$를 만족하는 두 소수라고 정의하자. g는 Z_p^*의 생성원이다. 프로토콜 초기 설정 단계에서 VLR (visited location register)와 HLR (home location register)는 비밀키 K_{VH}를 공유한다. ID_V와 ID_H는 VLR와 HLR의 신원정보 값으로 정의된다. $[M]_K$는 평문 M을 안전한 대칭키 암호화 기법과 키 K를 사용하여 생성한 암호문으로 정의되고 $h(M)$는 안전한 일방향함수를 사용하여 임의의 길이를 갖는 평문 M에 대해 생성한 k비트 해쉬값으로 정의된다. 해쉬 함수의 연속적인 사용을 정의하기 위해 다음과 같은 표기를 사용한다: $h^{(k+1)}(M) = h(h^{(k)}(M))$. 이때, $h^{(1)}(M) = h(M)$이다. 두 평문 M_1와 M_2의 연접을 다음과 같이 표기한다: $M_1 \| M_2$.

HLR는 난수인 x와 $v = g^x \bmod(p)$ 로 계산되는 (x,v)를 개인키/공개키 쌍으로 보관하고 있다. HLR은 MS (mobile station)에게 서비스를 제공하기 위해 난수 k를 생성하고 $\sigma = x + kK \bmod(q)$ 와 $K = g^k \bmod(p)$를 계산하여 (σ,K)를 안

전하게 보관한다. MS는 서비스를 제공받기 위해 인증 정보를 생성하기 위한 데이터로 (σ, K)를 HLR에게 받으며 (σ, K)는 MS의 저장 공간 (SIM 카드 등)에 저장된다.

2) 온라인 인증 단계

온라인 인증 단계를 수행하기 위하여 MS는 난수 n_1를 생성하고 해쉬 체인을 다음과 같이 생성하고 안전한 데이터 영역에 저장한다:

$$h^{(1)}(n_1), h^{(2)}(n_1), \ldots, h^{(n+1)}(n_1)(= N_1) \ .$$

각 통신 주체들은 온라인 인증을 위하여 다음의 절차를 수행한다:

Step 1. MS는 K를 VLR에게 전송한다.

Step 2. VLR는 난수 n_2를 생성하고 이를 ID_V와 함께 MS에게 전송한다.

Step 3. MS는 난수 t를 선택하고 저장된 $N_1 = h^{(n+1)}(n_1)$를 데이터베이스에서 복원하여 N_1, n_2, 그리고 ID_V에 대한 서명으로

$$r = g^t \bmod(p) \quad \text{와} \quad s = \sigma \times h(N_1 \| n_2 \| ID_V) + t \times r \bmod(q)$$

를 계산한다. 서명이 계산되면 r, s, K, N_1, ID_H, 그리고 ID_V를 VLR에게 전송한다.

Step 4. VLR는 다음의 조건식이 만족하는지 확인함으로써 전송받은 서명 정보를 검증한다: $g^s = (vK^K)^{h(N_1 \| n_2 \| ID_V)} r^r \bmod(p)$. 해당 조건식이 만족하지 않으면 VLR는 MS의 인증 요청을 거절하고 프로토콜을 종료한다. 위 조건식이 만족하면 암호문 $CT_1 = [N_1 \| n_2 \| K]_{K_{HV}}$, ID_H, 그리고 ID_V를 HLR에

게 전송한다.

Step 5. HLR는 전송받은 암호문 CT_1를 복호화하고 $N_1||n_2||K$를 복원한다. HLR는 자신이 서비스를 제공하는 모바일 사용자들의 정보를 저장한 데이터베이스에서 K에 대응되는 σ를 검색한다. 해당하는 값이 존재하면 임의로 선택한 난수 n_3에 대해 $C_1 = h(N_1||n_2||n_3||\sigma)$를 계산하고 또 다른 난수 k'를 선택하여 MS의 새로운 대리 서명키 (σ', K')를 다음과 같이 계산한다:

$$K' = g^{k'} \bmod(p) \quad \text{와} \quad \sigma' = x + k'K' \bmod(q).$$

HLR은 $CT_2 = [N_1||n_3||ID_V||\sigma'||K']_\sigma$와 $CT_3 = [CT_2||n_2||N_1||C_1]_{K_{HV}}$를 계산하고 CT_3, ID_H, 그리고 ID_V를 VLR에게 전송한다. HLR은 갱신된 대리 서명키 (σ', K')를 기존의 서명키인 (σ, K)대신에 저장한다.

Step 6. VLR는 전송받은 암호문 CT_3를 복호화하여 $CT_2||n_2||N_1||C_1$를 복원하고 n_2와 N_1을 확인한다. VLR은 $SK = C_1$을 MS과의 통신에서 사용할 세션키로 설정하고 CT_2와 ID_V를 MS에게 전송한다.

Step 7. MS는 CT_2를 복호화하여 복원된 $N_1||n_3||ID_V||\sigma'||K'$에서 N_1을 확인하고 세션키를 $SK = h(N_1||n_2||n_3||\sigma)$로 계산한다. MS는 (σ, K)대신에 (σ', K')를 저장한다.

3) 오프라인 인증 단계

프로토콜 구성의 설명에서 l의 초기 값은 N_1으로 설정되어있다. MS는 자신의 데이터베이스에서 $h^{(n-i+1)}(n_1)$를 찾아 $[h^{(n-i+1)}(n_1)]_{C_i}$를 계산하여 VLR에게 전송한다. 여기서 상수 n은 오프라인 인증의 회수 제한을 의미한다. VLS는 MS

에게 받은 값을 복호화하여 복원된 $h(h^{(n-i+1)}(n_1))$이 l과 동일한지 확인한다. 두 값이 동일하면 l을 $h^{(n-i+1)}(n_1)$로 갱신하고 새로운 세션키를 $C_{i+1} = h(l, C_i)$로 계산한다. VLS는 현재까지 수행한 오프라인 인증 회수에 대한 카운터 i를 $i = i+1$로 업데이트하고 $i \leq n$를 검사하여 오프라인 인증의 수행이 허용된 회수를 넘지 않았는지 검사한다.

제 4 장 프라이버시 보존형 데이터 처리 기술 개발

제 4 장 프라이버시 보존형 데이터 처리 기술 개발

제 1 절 연구 개발 필요성

1. 프라이버시 보존형 데이터 처리 기술

현대 사회는 모든 정보를 디지털화하여 저장하고 저장된 정보를 네트워크를 통해 공유하여 사용하는 사회로 변화하고 있다. 또한, 처리하는 자료의 양이 증가하고 다양한 서비스에 대한 요구가 늘어나면서 데이터를 직접 관리하는 대신 특화된 외부 저장 공간을 활용하는 환경의 사용이 크게 늘고 있다. 이에 따라 다양한 형태의 보안 문제가 발생하고 있으며, 이를 방지하기 위한 프라이버시 보존형 데이터 처리 기술에 대한 연구가 세계적으로 활발히 진행 중 이다.

1970년대 Yao[Yao82]에 의해 제시된 다자간 비밀 계산(Secure Multiparty Computation 또는 SMC)이 프라이버시 보존형 데이터 처리 기술의 시작이라 할 수 있는데, SMC 기술은 암호학 분야의 주요 연구 주제로 현재까지도 다양한 연구가 진행 중이다. 특히, 2000년대에 들어서 데이터마이닝, 침입탐지, 과학계산, 기하학적 계산, 통계적 분석, 인터넷 경매 등의 다양한 분야에 대한 다자간 계산 프로토콜이 연구되고 있다. 초기의 SMC 연구는 OT(Oblivious Transfer) 또는 BC(Bit Commitment)등의 암호 기반 기술을 사용한 이론적인 결과가 대부분으로, 이상적인 안전성을 만족하고 모든 문제에 일반적으로 적용 가능하다는 장점을 지니고 있지만 효율성의 문제 때문에 현실적인 적용을 기대하기는 힘들었다. 이 후의 연구는 OT, BC등의 효율성을 개선하여 실용적인 SMC 프로토콜을 제안하는 방향과 많은 계산량이 요구되는 암호 기반 기술을 사용하지 않고 랜덤화 기법 등을 이용하여 특

정 문제에 특화된 실용적인 프로토콜을 설계하려는 방향으로 진행되고 있다.

특히, 외부 저장 공간을 통한 프라이버시 정보의 유출 문제가 사회적으로 이슈화 되면서 외부 저장 공간에 민감한 프라이버시 정보를 안전하게 저장하고 활용하기 위한 프로토콜에 대한 연구가 활발히 진행 중이다. 대표적인 기술로는 암호화된 데이터에서 효율적인 데이터 검색이 가능한 검색 가능 암호 기술, 암호문에서 평문의 순서 정보를 추가적인 연산 없이 알 수 있는 순서 보존 암호화 기술, 암호문 사이에 대수적 연산이 가능한 homomorphic encryption 등이 있다.

2. 순서 보존 암호화 기술 연구 동향 및 필요성

순서 보존 암호화 기술은 누구든지 암호문으로부터 추가적인 연산 없이 대응하는 평문의 순서를 알 수 있는 암호화 기법이다. 즉, $m_1 > m_2$를 만족하는 모든 평문 m_1, m_2에 대해서 암호문이 $E_k(m_1) > E_k(m_2)$ 의 관계를 만족하는 것을 의미한다. 추가적인 연산이 필요 없이 순서 정보를 알 수 있다는 점에서 대수 비교 기능을 지니는 검색 가능 암호 기술 등과 차별된다. 평문의 정보를 최대한 숨기는 것을 목적으로 하는 암호화 기법에서 순서 정보를 노출시킨다는 것은 일견 모순되는 성질로, 이러한 순서 보존 암호화 기술은 오랫동안 많은 암호학자들의 관심을 끌어온 연구 분야이다.

순서 보존 암호화 기법은 일반적인 암호화 함수에 비해서 큰 정보를 공격자에 노출시키기 때문에 안전하지 않을 것이라는 의견이 지배적이며, 일반적인 암호화 함수가 지녀야 하는 여러 안전성 기준을 갖추지 못하는 것으로 알려져 있다. 그럼에도 불구하고 이러한 순서 보존 암호화 함수가 중요한 이유는, 데이터베이스에서의 검색, 정렬 등의 주요 연산 알고리즘이 원소의 대소비교를 통해서 이루어지는데, 순서 보존 암호화 기법은 이러한 대소비교를 추가적인 연산 없이 제공하고 있기 때문이다. 즉, 효율적이고 안전한 순서 보존 암호화 기법을 설계하는 것은 외부 저장 공간에 민감한 프라이버시를 효율적이고 안전하게 저장/활용할 수 있다는 것을 의미한다.

초기의 순서 보존 암호화 기법들은 주어진 평문을 상대적으로 크기가 큰 암호문 공간으로 난수화하여 배열하는 방식이었다. 하지만, 이러한 난수화 기법들은 공격자가 쉽게 평문의 값을 유추할 수 있다는 안전성 문제를 지니고 있다. 순서 보존 암호화에 대한 체계적인 연구는 2004년 Agrawal[AKS04] 등에 의해서 최초로 이루어졌다. Agrawal 등은 평문의 분포를 사용자만이 알고 있는 정보로 가정하고, 이 평문의 분포를 암호화에 이용하여 주어진 평문의 분포가 암호화 이후에 드러나지 않도록 암호화 함수를 구성하였다. 그들은 또한 체계적인 안전성도 제안하였는데, 그들은 공격자가 주어진 암호문 집합으로부터 암호화 이전의 평문 집합이 가지는 분포를 유추할 수 없는 경우, 안전한 순서 보존 암호화 기법으로 정의하였다. 하지만, 이러한 방식의 구성은 사용자가 암호화 작업을 시작하기 이전에 자신이 암호화할 (또는 미래에 갱신될) 모든 데이터에 대한 정보를 알고 있다는 가정하에서 안전성이 보장되기 때문에 현실적인 암호화 함수와는 약간 동떨어져 있다고 할 수 있다. 또한 공격자가 몇몇 평문-암호문 쌍을 획득하는 경우에 대한 안전성 논의가 충분하지 않다.

이후 2009년 Boldyreva[BCL09] 등은 증명 가능 안전성을 순서 보존 함수에 적용하기 위한 연구를 수행하였다. 이들은 전통적인 암호화 함수에 대한 증명가능 안전성 접근 방법에서 벗어나 유사 난수 생성 함수의 안전성 정의를 사용하여 순서 보존 암호화 기법에 대한 새로운 증명 가능 안전성 정의를 제안하였다. 유사 난수 함수의 안전성은 유사 난수 함수로 생성된 유사 난수열을 실제 난수열과 구별할 수 없는 것으로 정의된다. 이와 유사하게, Boldyreva 등은 임의의 순서 보존 암호화 함수가 임의의 순서 보존 함수와 구별할 수 없는 것으로 정의하였다. 하지만, 이러한 접근 방법은 암호화 함수가 가져야 하는 기본적인 안전성을 만족하지 못하는 것으로 여겨진다. 암호화 함수가 기본적으로 지녀야하는 안전성은 공격자가 암호문으로부터 평문의 정보를 유추할 수 없도록 보장하는 것이다. 하지만, Boldyreva 등이 제안한 방식에서는 순서 보존 암호화함수가 가능한 모든 순서 보존 함수의 집합에서 임의로 선택되는 방식을 사용했기 때문에 전체적인 분포가 모든 순서 보존 함수의 집합이 가지는 분포와 동일하다. 즉, 공격자는 각각의 암호

문이 복호화될 수 있는 평문 후보에 대한 분포를 가능한 모든 순서 보존 함수의 집합에서 계산이 가능한데, 이 분포를 살펴보면, 대부분의 암호문이 특정 평문과 대응할 확률이 매우 크게 나타나는 것을 알 수 있다. 또한 이러한 현상은 평문과 암호문의 수가 클수록 커지며 또한, 약간의 오차를 인정하는 경우 훨씬 심각한 문제를 발생시킨다.

이렇듯, 순서 보존 암호화 기법에 대한 연구는 현재 아직 이론적인 단계에 머물러 있으며, 우선적으로 순서 보존 암호화 기법에 대한 체계적인 안전성 정의가 필요한 상황이다. 이에 비해, 좀 더 실용적인 관점에서 암호화된 데이터의 범위 검색 문제 등에 접근하는 방안으로 bucket 기반 인덱스 기법들도 제안되고 있다.

Bucket 기반 인덱스 기법은 데이터가 속해 있는 전체 평문 구간을 bucket이라고 부르는 세부 구간으로 나누어 각각의 bucket에 임의로 선택한 난수화된 인덱스를 할당한다. 또한 각 평문은 별도의 암호화 시스템으로 암호화되어 데이터베이스에 인덱스와 함께 저장된다. 사용자는 데이터 검색을 위해서 각 bucket들의 구간 정보와 인덱스를 보관하고 있으며, 필요한 경우 원하는 bucket에 해당하는 인덱스를 서버에 질의한다. 서버는 질의받은 인덱스를 가지는 암호화된 데이터를 모두 사용자에게 전송한다. 사용자는 이 데이터를 복호화하여 원하는 데이터를 찾을 수 있다.

Bucket 기반 인덱스 기법은 우선 난수화된 인덱스와 독립적으로 암호화된 암호문을 저장하기 때문에 높은 안전성을 지닌다고 할 수 있다. 또한 검색 가능 암호 시스템과 같이 별도의 연산을 서버에 요구하지 않고 기존의 데이터베이스 환경에 바로 적용가능하다는 큰 장점을 지니고 있다. 반면 단점으로는, bucket 단위로 데이터가 처리되기 때문에 실제 사용자가 원하는 데이터가 bucket의 극히 일부분이더라도 bucket 내의 모든 원소를 전송받아 복호화하는 과정을 수행해야 하기 때문에 실제 사용자의 작업량이 증가한다는 점을 들 수 있다. 또한, 질의가 많아질수록 공격자는 서버에 저장된 암호문과, 질의-응답 정보, 평문 공간의 통계적인 분포 등을 이용하여 bucket들 간의 위치 정보를 추정할 수 있고 더 나아가 특정 평문의 값을 유추하는 것 또한 가능하기 때문에 이에 대한 안전성 논의가 필요하다.

제 2 절 순서 보존 암호화 기술 요구 사항 분석

1. 안전성 요구 사항

순서 보존 암호화 기법에 대한 안전성에 대한 연구는 아직 진행 중이며 보편적으로 인정받는 체계적인 안전성 정의는 정립되어 있지는 않다. 여기서는 현재까지 알려진 안전성 문제를 지적하고 순서 보존 암호화 기법들이 지녀야 할 기본적인 안전성을 논의하기로 한다. 또한, 순서 보존 암호화 기법이 대용량의 데이터베이스 등과 같은 효율성이 강조되는 응용 환경에 주로 적용되기 때문에, 안전성 보다는 효율성을 목적으로 설계되는 기법도 다수 존재한다.

가. 대칭키 암호화

순서 보존 암호화 기법에서는 일반적인 암호화 기법이 기본적으로 만족하는 선택 평문 공격(chosen plaintext attack)을 허용하지 않는다. 선택 평문 공격을 허용할 경우, 주어진 암호문에 대한 복호화가 가능하기 때문이다. 암호문 C가 공격자에 주어졌다고 가정하면, 공격자는 우선 임의의 평문 m_1을 선택하여 암호화 질의를 통해 얻은 $E_k(m_1)$를 암호문 C와 비교한다. 비교 결과 $E_k(m_1) < C$를 만족한다면, 공격자는 암호문 C에 대응하는 평문이 m_1보다 큰 값이라는 정보를 얻게 된다. 이러한 이진 검색 과정을 반복하면 정확한 평문 값을 계산할 수 있게 된다. 또한, 공개키를 사용하여 누구든지 암호화가 가능한 순서 보존 암호화 기법을 설계하는 것은 공격자에게 선택 평문 공격을 허용하는 것이기 때문에 결국 순서 보존 암호화 기법은 대칭키 기반으로 설계해야 한다.

나. 안전성 정의

순서 보존 암호화 기법의 안전성으로는 현재까지 2004년 Agrawal 등이 제안한

안전성 정의가 가장 널리 받아들여지고 있다. 이 안전성 정의에서 Agrawal 등은 다음과 같은 성질을 만족하는 순서 보존 암호화 기법을 안전한 것으로 정의했다.

우선 공격자 A는 주어진 암호화 기법을 바탕으로 두 개의 평문 집합 M_1과 M_2를 결정한다. 이 때, 두 평문 집합은 동일한 수의 평문을 포함하는 것을 가정한다, 즉 $|M_1| = |M_2|$. 사용자는 임의로 b(0 또는 1)를 선택하여 자신의 비밀키로 $E_k(M_b)$를 계산한다. $E_k(M_b)$는 M_b에 포함된 평문을 각 암호화한 암호문들의 집합 $\{E_k(m_1), E_k(m_2), ..., E_k(m_n)\}$을 의미한다. 사용자는 공격자에 $E_k(M_b)$를 보여주고, 공격자는 $E_k(M_b)$를 보고 사용자가 임의로 선택한 b의 값을 추정한다. 이 때, 공격자가 b를 정확하게 추정할 확률이 1/2보다 큰 경우 공격자가 공격에 성공한 것으로 가정한다. 이러한 공격자가 존재하지 않는 경우, 주어진 순서 보존 암호화 기법이 안전한 것으로 정의한다.

이러한 안전성 정의는 순서 보존 암호화 기법이 안전한 경우 암호화된 데이터 집합이 가지는 분포와 원 평문 집합이 가지는 분포 사이에 연관성을 찾기 어렵다는 것에서 기인하였다. 반대로, 평문 집합과 암호화된 집합의 분포가 어떠한 연관성을 지닌다는 것은, 암호화 과정에서 평문의 순서 정보 이외에 다른 정보가 충분히 은닉되지 못했다는 의미를 가진다.

순서 보존 암호화 기법에 대한 증명 가능 안전성 정의는 2009년 Boldyreva등에 의해서 최초로 정의되었다. 이들은 의사 난수 생성기의 증명 가능 안전성 정의에 주로 사용되는 개념을 사용하여 순서 보존 암호화 기법에 대한 안전성을 정의하였다. 의사 난수 생성기의 안전성은 다음과 같이 정의된다: seed로부터 의사 난수 생성기를 사용하여 생성한 의사 난수열과 동일 길이를 가지는 실제 난수를 어떠한 공격자도 구별하지 못하는 경우 이러한 의사 난수 생성기를 안전한 것으로 정의한다. 이와 유사하게, 비밀키로부터 결정되는 순서 보존 암호화 함수와 동일 평문구간-암호문구간에서 정의되는 모든 순서 보존 함수에서 임의로 선택된 함수를 어떠한 공격자도 구별하지 못하는 경우 이러한 순서 보존 암호화 함수를 안전한 것으로 정의하였다. 하지만, 이러한 암호화 방법은 기본적으로 순서 보존 함수 전체의

집합은 암호학적으로 안전하지 않다는 문제가 있는데, 공격자가 순서 보존 암호화 함수와 일반적인 순서 보존 함수를 구별하지 못하는 경우라도 주어진 암호문으로부터 높은 확률로 대응하는 평문을 추측할 수 있다는 것이다. 따라서 순서 보존 암호화 함수에 대한 증명 가능 안전성에 대한 정의는 추가적인 논의가 필요한 상황이다.

다. Bucket 기반 인덱스 기법의 안전성

일반적으로 bucket 인덱스는 bucket에 포함된 평문의 내용과 독립적으로 난수화 과정을 통해서 결정되기 때문에 데이터베이스에 저장된 bucket 인덱스로부터 해당 평문의 정보를 얻는 것은 이론적으로 불가능하다. 따라서 bucket 기반 인덱스 기법의 안전성을 논의할 경우, 각 평문-암호문 사이의 연관성 대신 각 bucket의 정보를 공격자가 어느 정도 정확하게 추정할 수 있는지에 초점을 맞추게 된다.

Bucket 기반 인덱스 기법에서 사용자는 bucket을 정의하는 단계에서 equi-width 또는 equi-depth의 방법에 기반하게 된다. Equi-width 방법은 각 bucket이 동일한 너비를 가지도록 분할하는 방식이고, equi-depth는 각 bucket에 동일한(또는 유사한) 개수의 원소들이 포함되도록 분할하는 방법이다. Equi-width 분할 방식을 사용한 경우, 공격자가 평문 집합의 분포를 알고 있다면 공격자는 데이터베이스에 저장된 암호문에서 각 bucket 인덱스의 비율을 계산하는 것으로 실제 bucket이 평문 구간에서 위치하는 범위를 추정할 수 있다. 예를 들면, 평문 집합이 정규 분포를 가진다고 가정하면 가장 큰 빈도를 보이는 bucket 인덱스에 해당되는 bucket이 평문 구간의 가운데에 위치하고, 반대고 가장 작은 빈도를 보이는 두 bucket 인덱스가 평문 구간의 양 끝에 위치한다는 점을 추정할 수 있다. 이러한 문제점 때문에 bucket을 분할할 때, equi-depth 방식에 기반하여 분할하는 것으로 가정한다.

2. 효율성 요구 사항

현재까지 제안된 순서 보존 암호화 기법의 경우 이론적인 접근을 시도하여, 계산량 또는 저장량 등에서 실제 데이터베이스에 적용하기에는 비효율적이라는 단점을 지니고 있다. 이에 실용적으로 적용 가능한 효율성을 가지는 순서 보존 암호화 기법의 개발이 필요하다.

Bucket 기반 인덱스 기법의 경우, bucket 단위로 질의-응답, 복호화가 수행되기 때문에 실제 요구하는 데이터 외에 동일 bucket에 저장된 자료에 대한 연산이 추가된다. 이러한 추가적인 연산의 양을 최소화 시키는 것이 bucket 기반 인덱스 기법의 주요 연구 주제이다.

제 3 절 순서 보존 암호화 기법 개발안

1. Pivoting 기반 순서 보존 암호화 기법

가. Ideal OPE

구체적인 순서 보존 암호화 기법을 설계하기 인전에 가장 이상적인 안전성을 가지는 순서 보존 암호화 기법을 고려해 보기로 한다. 평문 집합 $S=\{d_1, d_2, ..., d_n\}$에 대해, 각각의 평문은 순서대로 정렬된 것으로 생각한다. 즉, $i<j$라면, $d_i > d_j$를 만족한다. 이 때 이러한 평문에 대한 어떠한 정보도 공격자에 노출시키지 않으면서 순서 정보만을 보여주는 방법은 각 평문이 집합 S내에서 가지는 순서 정보만을 표시하는 것이다. 즉 이 예에서, $S=\{d_1, d_2, ..., d_n\}$를 $E(S)=\{1,2,...,n\}$로 암호화하는 것을 생각할 수 있다. 하지만, 이러한 방법은 정당한 사용자조차도 암호문으로부터 원래 평문을 복호화 할 수 없다는 단점을 지니고 있다. 이것을 보완하기 위해 순서 정보 후에 안전성이 보장된 암호화 기법을

사용한 암호문을 추가하여 $E_k(S) = \{1||E_k(d_1), 2||E_k(d_2), ..., n||E_k(d_n)\}$를 저장한다, 여기에서 $E_k(d_i)$는 평문 d_i를 비밀키 k를 사용하여 암호화한 암호문을 의미한다. 즉, 평문 집합에서 i번째 크기를 가지는 평문 d가

$$i||E_k(d)$$

로 암호화된다. 이 때, i를 고정된 크기로 표현하면 누구든지 암호문을 보고 평문의 순서 정보를 알아낼 수 있다. 반대로 암호문의 뒷부분은 안전성이 검증된 암호화 기법을 사용하여 암호화가 이루어져 있으므로 어떠한 공격자라도 그 암호문으로부터 평문의 정보를 추가적으로 얻는 것은 불가능하다. 이러한 방식의 암호화 기법은 가장 완벽한 순서 보존 암호화 방법을 제공하지만, 여러 단점을 지니고 있어 실제 환경에서 사용하기에는 무리가 있다.

이 기법이 가지는 단점은 모든 평문이 동시에 사용자에 주어진 경우에는 사용자가 쉽게 평문을 정렬하여 앞에서 설명한 암호화 과정을 수행할 수 있지만, 평문의 일부가 암호화된 후에 추가로 평문이 주어지는 경우에는 효율적인 암호화가 불가능하다는 점이다. 이 경우, 사용자는 이미 암호화된 평문을 모두 복호화하여 새로 주어진 평문의 순서를 결정해야 하며 또한 이전의 암호문들을 일부 수정해야 한다.

나. Pivoting

앞 절에서 설명한 Ideal OPE의 단점을 개선하기 위해서 다음과 같은 변형을 생각할 수 있다. Ideal OPE의 기본이 되는 구조는 주어진 평문 집합에서 평문이 가지는 순서 정보를 암호화에 활용하는 것인데, 이러한 순서 정보는 모든 평문 집합이 공개되어 있을 경우에만 계산이 가능하다는 문제를 가지고 있다. Ideal OPE에서 각 평문의 순서 정보(order)는 다음과 같은 방식으로 계산이 된다. $Ord(d_i, S) = |\{x|x > d_i, x \in S\}|$. 이러한 Ord함수를 다음과 같이 변형한다.

$$Ord'(d_i, S) = |\{x|x > d_i, x \in S'\}|$$

여기에서 집합 S'는 평문 공간에서 임의로 선택된 집합이다. Ord'를 Ord 대신 사용하여 Ideal OPE 암호화를 수행하는 경우, 특정 평문의 순서 정보를 얻기 위해서 전체 평문 집합이 필요하지 않기 때문에 전체 평문 집합에 대한 정보가 부족한 경우에도 암호화가 가능하다. 위의 정의에서 평문의 순서 정보를 얻기 위해서 기준으로 사용된 S'를 기준(pivoting) 집합으로 부르기로 한다.

S'는 임의로 선택된 집합이기 때문에 선택하는 방법에 의해서 평문의 순서가 정확하지 않을 확률이 존재한다. 즉, $S' = \{p_1, p_2, ..., p_r\}$이라 가정할 때, 모든 평문 $d_i < d_{i+1}$에 대해서 $d_i < p_j < d_{i+1}$을 만족하는 p_i가 존재하는 경우에는 평문의 순서가 정확하게 계산이 되지만, 그렇지 않은 경우 두 이웃하는 평문의 순서가 동일하게 계산이 되기 때문이다. 임의의 평문에 대해서 정확한 암호화를 수행하기 위해서는 결국 S'이 매우 많은 원소를 포함하고 있어야 하는데, 이는 암호화 과정의 효율성을 저하시키는 원인이 된다.

다음 절에서 제안하는 순서 보존 암호화 방법에서는 기준 집합을 활용한 순서 결정과정을 효율적으로 개선하여 효율적인 암호화가 가능하도록 변형한 것이다. 간략하게 설명하면, 주어진 평문의 순서를 결정할 때 전체 기준 집합 S' 전체를 동시에 활용하지 않고 S'의 하나의 값과 비교를 하고, 비교 결과에 따라 다음 비교에 활용될 기준점(pivot point)가 결정되는 과정을 반복하여 적용시킨다. 또한 S'이 하나의 비밀키로부터 생성이 되도록하여 사용자의 저장 공간도 효율적으로 개선하였다.

다. Pivoting 기반 순서 보존 암호화 기법

초기 준비 단계에서 평문 공간 M과 암호문 공간 N에 대한 정보가 주어지고, 사용자는 암호화에 필요한 비밀키(K)와 기준점 생성에 사용할 의사 난수 생성기를 결정한다. 편의상 적당한 자연수 n, m에 대해서 $M = [1, 2^m]$, $N = [1, 2^n]$으로 정의한다.

암호화 단계에서 하나의 평문이 주어지면 pivot 생성 단계, pivot을 이용한 암호화 단계, 평문 조정 단계를 총 n번 반복 수행하여 n-bits의 암호문을 생성한다. 암호화 단계의 기본 구조는 그림 4-1에 나타나 있다.

평문 a가 주어진 경우의 암호화를 예를 들어 설명하면 다음과 같다. 사용자는 비밀키 K를 이용하여 첫 번째 기준점을 생성한다.

$$p_1 = PRNG(K \| 1)$$

이 때, $PRNG(seed)$는 $seed$로부터 난수열을 생성하는 의사 난수 생성기이다. 또한, 생성한 기준점은 $1 \leq p_1 \leq 2^m$ 을 만족하도록 한다. 사용자는 생성된 기준점(p_1)을 평문 a와 비교하여 $a < p_1$ 을 만족하는 경우 암호문 bit b_1 = 0, 반대의 경우($a \geq p_1$) b_1 = 1 로 결정한다.

평문 조정 단계에서 b_1 = 0의 경우, $[1, p_1]$의 구간을 평문 전체 공간 $[1, 2^m]$로 재조정하고 평문의 크기를 이에 따라 조정한다. 조정된 평문 a'은 다음과 같이 결정된다.

$$a' = a \times \frac{2^m}{p_1}$$

b_1 = 1의 경우, $[p_1, 2^m]$의 구간을 평문 전체 공간 $[1, 2^m]$로 재조정하고 평문의 크기를 이에 따라 조정한다.

$$a' = (a - p_1) \times \frac{2^m}{2^m - p_1}$$

사용자는 다음 반복 단계에서 a 대신 위에서 계산된 a'를 평문으로 사용하여 다음 암호화 bit를 계산한다. 이러한 반복 단계를 총 n번 반복하여 매 단계마다 한 bit의 암호문을 출력하는 데, 각 $i (1 \leq i \leq n)$ 번 단계에서의 기준점은

$$p_i = PRNG(K \| i \| b_1 \| b_2 \| ... \| b_{i-1})$$

로 계산된다. 평문 a에 대한 최종 암호문은 $b_1 \| b_2 \| b_3 \| ... \| b_n$ 이다.

이 암호화 과정은 Ideal OPE와 달리 암호문으로부터 복호화가 가능하기 때문에

추가적인 암호문을 저장하지 않는다. 위의 암호문으로부터 복호화는 암호화 단계의 역산으로 비밀키 K를 알고 있는 사용자만이 정확한 pivot을 계산할 수 있고 해당하는 평문을 복원할 수 있다. 암호문을 $b_1 \| b_2 \| b_3 \| ... \| b_n$라 가정하면, 평문은 다음과 같이 계산된다.

Step 1. $p_1 = PRNG(K \| 1)$

Step 2. $x = p_1 \times b_1$, $a = 2^m$로 초기화

Step 3. $i(2 \leq i \leq n)$에 대해서 반복

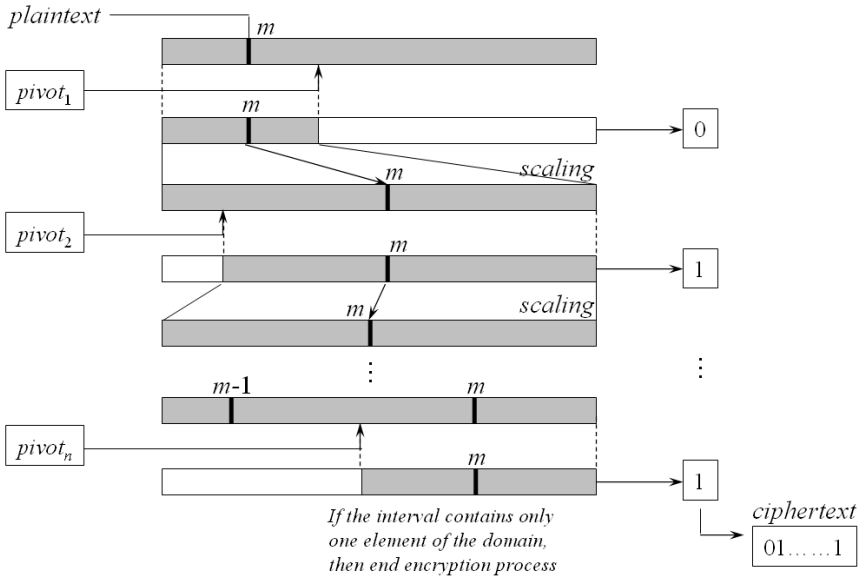
3-1) $b_{i-1} = 0$이면, a값을 $a \times \frac{p_{i-1}}{2^m}$로 갱신

3-2) $b_{i-1} = 1$이면, a값을 $a \times \frac{2^m - p_{i-1}}{2^m}$로 갱신

3-3) $p_i = PRNG(K \| i \| b_1 \| b_2 \| ... \| b_{i-1})$ 계산

3-4) x값을 $x + (a \times p_i) \times b_i$로 갱신

Step 4. 최종 x값을 평문으로 출력

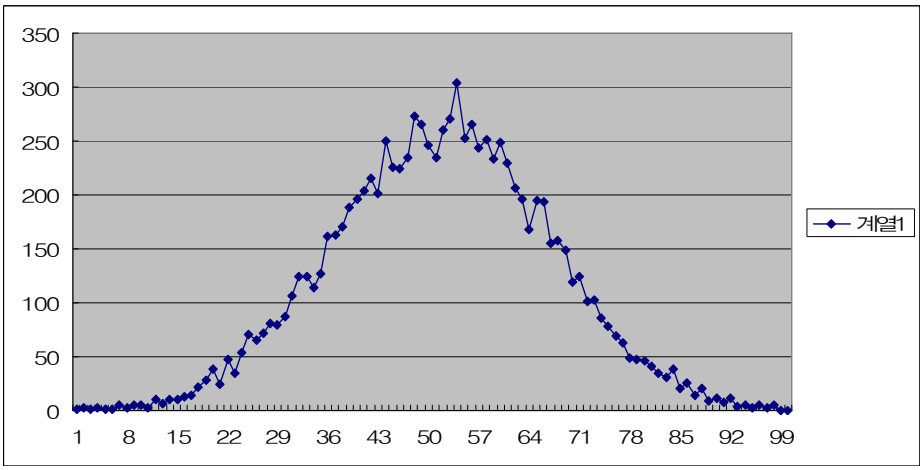


[그림 4-1] Pivoting 기반 순서 보존 암호화

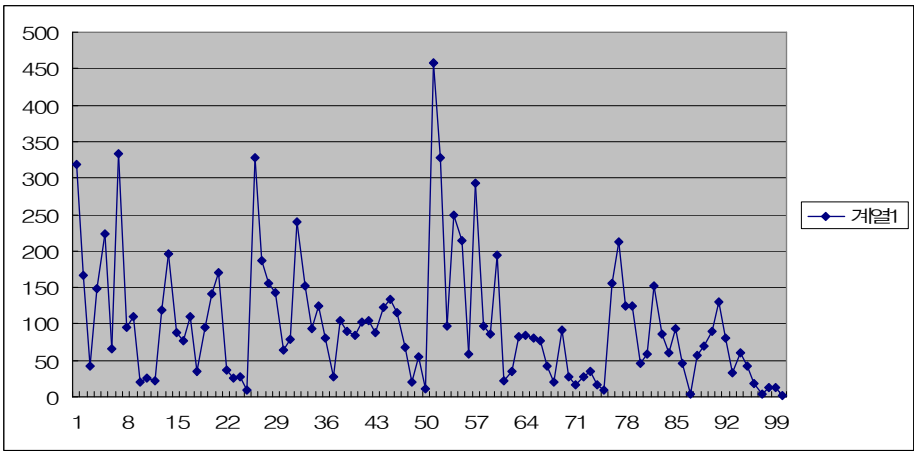
라. 제안된 기법의 안전성 분석

제안된 순서 보존 암호화 기법에 대한 안전성 분석을 위해 임의로 선택된 여러 평문 집합에 대한 암호화를 수행하여 평문과 암호문의 분포 사이의 연관성을 살펴보았다.

아래 그림 4-2,3의 결과는 정규 분포를 가지는 평문 10,000개로 이루어진 평문 집합에 대한 암호화 결과를 보여준다. 평문은 10-bits, 암호문은 20-bits로 구성되었으며 비밀키를 변화시켜 가며 100번 반복 수행하였다.

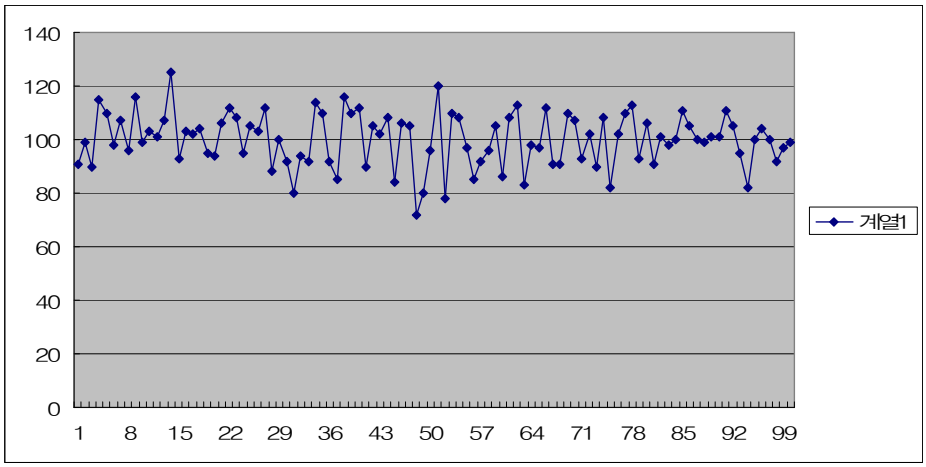


[그림 4-2] 정규 분포를 지니는 평문 집합

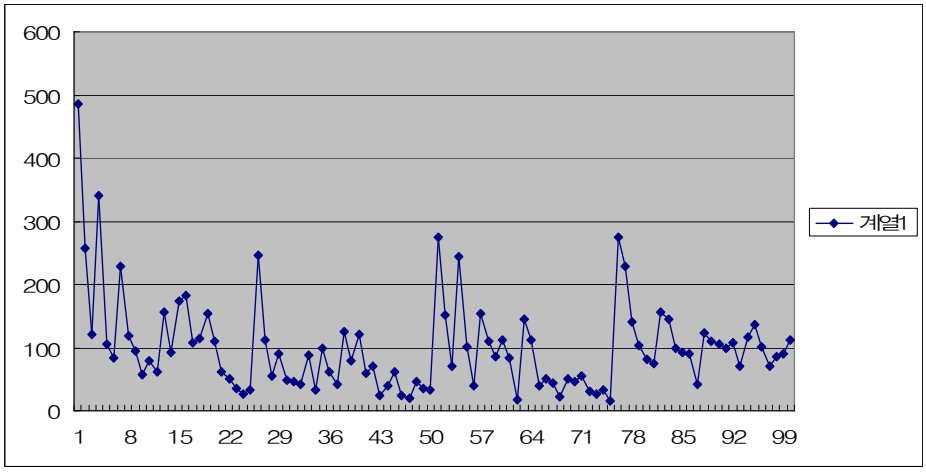


[그림 4-3] 암호화 결과(1), 100번 반복 수행 평균값

또한, 동일한 실험을 균일 분포를 가지는 평문에 대해 반복하였다 (그림4-4, 4-5).



[그림 4-4] 균일 분포를 지니는 평문 집합



[그림 4-5] 암호화 결과(2), 100번 반복 수행 평균값

위의 두 실험 결과로부터, 서로 다른 분포를 가지는 평문 집합에 대해 암호화를 적용한 결과 암호문 집합이 가지는 분포는 거의 동일하다는 사실을 알 수 있다. 이는 제안된 순서 보존 암호화 기법이 Agrawal등이 제시한 안전성 기준을 만

족하고 있다는 것을 의미한다.

2. Bucket 기반 인덱스를 사용한 범위 검색 기법

순서 보존 암호화 기법을 사용하는 가장 주된 목적은 암호화된 데이터 집합 내에서 순서 정보를 이용한 범위 검색, 정렬 등의 연산을 자유롭게 수행하기 위해서이다. 이러한 문제에 대한 다른 접근 방법으로 검색 가능 암호 기술이나 bucket 기반 인덱스 기법 또한 많은 주목을 받고 있다. 특히, bucket 기반 인덱스 기법의 경우 암호학적인 프리미티브를 사용하지 않기 때문에 효율적인 검색이 가능하며, 또한 기존 데이터베이스의 구조에 큰 변화를 가하지 않고 적용이 가능하다는 장점을 지니고 있다.

Bucket 기반 인덱스 기법은 평문 공간을 일정한 크기를 지니는 작은 부분 집합(bucket)으로 분할한 후 동일 bucket 내의 평문들을 임의로 선택된 bucket index로 대체하여 저장하는 방법으로 암호화를 수행한다. 이 결과, bucket 인덱스로부터 원 평문을 복호화하는 것이 불가능해지기 때문에 bucket 인덱스 이외에 원 평문을 독립적으로 암호화한 암호문을 저장하고 추가로 bucket 인덱스를 저장하는 형태로 데이터베이스에 저장된다.

다음의 예로 살펴보면, 표1은 사용자id와 월급을 나타내는 데이터베이스의 예이고, 표4-2는 표4-1의 데이터베이스를 bucket 기반 인덱스로 암호화 한 결과이다. 표4-2를 살펴보면 e-tuple은 데이터베이스의 각 행을 암호화한 암호문이고 각 속성에 bucket 인덱스가 추가된 형태를 지닌다. id_number의 경우, 각 bucket에 3개의 평문이 포함되도록 [0, 15], [16, 30], [31, 52], [55, 100]의 4개의 bucket으로 분할되었으며, 각각 π, σ, ρ, τ의 bucket 인덱스를 할당받았다.

id_number	salary
68	480
7	340
11	790
31	630
29	435
57	724
51	587
14	412
21	345
39	480
55	607
17	530

[표 4-1] 데이터베이스 예(1)

E-tuple	bucket index (id_number)	bucket index (salary)
1100110011100…	τ	β
1000011100010…	π	α
10100110011111…	π	δ
11110100001111…	σ	δ
10010110011110…	ρ	β
11101111000010…	τ	δ
10000000011100…	σ	γ
11010110000010…	π	α
10110110110010…	ρ	α
01010110100010…	σ	β
11010110100011…	τ	γ
10010110101010…	ρ	γ

[표 4-2] Bucket 기반 인덱스 적용 예

가. 순환 버킷 질의(Cyclic Bucket Query)

1) Bucket Index 정렬 문제

일반적으로 bucket 인덱스는 bucket에 포함된 평문의 내용과 독립적으로 난수화 과정을 통해서 결정되기 때문에 데이터베이스에 저장된 bucket 인덱스로부터 해당 평문의 정보 또는 각각의 인덱스 사이의 어떠한 연관성을 추정하는 것은 이론적으로 불가능하다. 하지만, 공격자가 사용자와 서버 사이의 질의-응답 정보를 추가적으로 수집하는 경우 암호화된 평문에 대한 정보를 추가적으로 얻을 수 있다.

표4-1,2에 제시된 bucket 기반 인덱스 기법 활용 예를 사용해서 살펴보도록 하자. 사용자가 [10, 20]에 해당하는 id_number를 질의하는 경우, 사용자는 π와σ의 bucket 인덱스를 서버에 질의하게 된다. 이 후 질의를 통해 공격자가 $\{\pi,\sigma\}$, $\{\sigma,\rho\}$, $\{\rho,\tau\}$의 세 질의에 대한 정보를 얻었다고 가정하자. 이 경우 공격자는 $\{\pi,\sigma\}$의 질의를 통해서 bucket 인덱스 π와 σ가 인접한 bucket이라는 정보를 얻을 수 있다. 이와 유사한 방법으로 정보를 수집하여, bucket 인덱스 사이에 $\pi < \sigma < \rho < \tau$ 의 연관성을 추정할 수 있다. 이렇듯 공격자가 여러 질의-응답 정보를 이용하여 bucket 인덱스 사이의 연관성을 추정하는 것을 bucket 인덱스 정렬 문제라 부르기로 한다.

Bucket 인덱스 정렬 문제가 심각한 이유는 bucket 기반 인덱스 기법에서 공격자가 암호화된 데이터의 평문 분포를 알고 있다고 가정하는 것이 일반적이기 때문이다. 즉, 위의 예에서 $\pi < \sigma < \rho < \tau$ 라는 정보, 각 bucket에 포함된 암호문의 빈도, 평문의 분포로부터 인덱스 π에 해당하는 bucket의 양 끝 값을 대략적으로 추정할 수 있다. 만약, 평문의 분포가 정규 분포를 지니고 있는 경우라면, 중간값에 해당하는 bucket의 경우는 양 끝 값의 차이가 매우 적을 것이므로 공격자가 해당 bucket에 포함된 평문의 값을 거의 정확하게 유추할 수 도 있게 된다.

2) 순환 버킷 질의(Cyclic Bucket Query)

Bucket 인덱스 정렬 문제에 의한 안전성 훼손을 방지하기 위한 방법으로 순환 버킷 질의(cyclic bucket query)를 제안한다. 순환 bucket 질의는 사용자가 실제 질의하고자 하는 bucket과 이웃하고 있는 bucket을 동시에 질의하는 방법이다. 즉, 예를 들어 평문 공간이 $B_1 < B_2 < B_3 < \ldots < B_n$으로 분할되었다고 가정하고, 사용자가 $\{B_{n-1}, B_n\}$에 대한 질의를 수행하고자 하는 경우, 사용자는 서버에게 $\{B_{n-1}, B_n, B_1\}$를 질의한다. 물론 서버는 B_{n-1}, B_n, B_1에 속해 있는 암호화된 데이터를 사용자에게 전송하지만, 사용자는 원래 질의하고자 했던 B_{n-1}, B_n에 대해서만 복호화를 수행한다. 이 결과, 서버에서 사용자로의 데이터 전송량은 약간 증가하지만, 사용자의 계산량은 변하지 않는다.

공격자가 bucket 인덱스 정렬 공격을 수행한 경우, 만약 순환 버킷 질의를 수행했다면, 공격자는 bucket 인덱스 사이의 연관성을 순환 형태로 얻게 되어 어떠한 인덱스가 첫 번째 bucket에 대응하는지를 알 수 없게 된다. 앞의 표1,2의 예로 살펴보면, 공격자가 최종적으로 얻는 정보는 $\ldots < \pi < \sigma < \rho < \tau < \pi < \ldots$ 의 정보로 최대 네 가지의 경우의 수를 가지게 된다. 결국, 공격자는 $\pi < \sigma < \rho < \tau$, $\sigma < \rho < \tau < \pi$, $\rho < \tau < \pi < \sigma$, $\tau < \pi < \sigma < \rho$ 의 네 가지 중에서 어느 것이 실제 순서 인지를 확인할 수는 없다. 이 경우 공격자가 bucket의 위치를 추정할 확률은 1/4로 bucket 인덱스 정렬 공격을 수행하지 않은 상태에서 임의로 추정할 확률과 동일하다.

나. Bucket 기반 인덱스 기법과 순서 보존 암호화 기법의 결합

Bucket 기반 인덱스 기법은 기본적으로 모든 연산을 bucket 단위로 수행하기 때문에 질의-응답 과정에서 항상 원하지 않은 데이터가 검색 결과에 포함되는 false positive가 발생하게 된다. 위의 표1,2를 예로 살펴보면, 만약 사용자가 [0, 10]의 자료를 검색하고자 하는 경우, 실제 해당하는 데이터는 하나 뿐 이지만 서버는 bucket 인덱스 π에 해당하는 세 자료를 사용자에 전송하고 사용자 또한 세

자료를 모두 복호화하기 전에는 해당 자료가 자신이 원하는 자료인지를 확인할 방법이 없다. 이러한 문제는 bucket 기반 검색 기법들이 지니는 근본적인 문제로 bucket 기반 검색 기법들의 효율성 개선을 위해서 필히 해결해야 하는 매우 중요한 문제라고 할 수 있다.

다음에서 bucket 기반 검색 기술과 순서 보존 암호화 기법을 결합하여 false positive 문제를 해결하는 새로운 기법을 제시한다. 설명의 편의를 위해서 다음 표3과 표4를 사용한다. 표3은 위의 표1과 동일하고 표4는 표3의 데이터에 새로 제안한 기법을 적용한 결과이다.

id_number	salary
68	480
7	340
11	790
31	630
29	435
57	724
51	587
14	412
21	345
39	480
55	607
17	530

[표 4-3] 데이터베이스 예(2)

우선, 사용자는 암호화를 위한 비밀 키 K를 랜덤하게 생성하고, 대칭키 암호 알고리즘을 이용하여 데이터베이스 내에 있는 데이터를 암호화한다. 표 4-4의 E-tuple 열의 첫 번째 행 1100110011100… = $E_K(68,480)$을 의미한다. 여기에서 $E_K(\ )$는 비밀 키 K를 가지는 대칭키 암호 알고리즘이며, E-tuple은 표 4-3의 각각의 행을 암호화한 값을 의미한다.

E-tuple	E-id_number		E-salary	
	B-index	ind-id_number	B-index	ind-salary
1100110011100…	τ	4501	β	4221
1000011100010…	π	4401	α	6541
1010011001111…	π	3015	δ	7069
1111010000111…	σ	3851	δ	9831
1001011001110…	ρ	7951	β	8537
1110111100010…	τ	7900	δ	4207
1000000001100…	σ	647	γ	7631
1101011000010…	π	4599	α	6299
1011011011010…	ρ	2001	α	4851
0101011010010…	σ	4560	β	4211
1101011010011…	τ	3966	γ	2157
1001011010101…	ρ	3999	γ	6780

[표 4-4] Bucket 기반 검색 기법 적용 예

다음 인덱스 생성 단계로 bucket 인덱스 생성과 bucket 내의 데이터에 대한 인덱스 생성의 두 단계로 구성된다. Bucket 인덱스 생성 과정에서 사용자는 데이터베이스 내의 데이터들의 전체 구간 $B=[a,b]$를 세부 bucket $B_1=[a_0(=a),a_1)$, $B_2=[a_1,a_2)$, …, $B_n=[a_{n-1},a_n(=b)]$로 분할한다. 구간을 분할할 때, 각각의 bucket에 동일한 수의 데이터가 포함되도록 equi-depth 방식에 의해서 분할한다. 그 다음 사용자는 각각의 bucket에 대해 임의의 인덱스를 생성하여 할당하고, 사용자는 검색을 위해 각각의 bucket에 대해 시작점과 끝점 및 인덱스를 저장한다.

표 4-3의 salary에 대해 살펴보자. Salary 전체의 범위가 $B=[300,800]$ 일 때, $B_1=[300,420)$, $B_2=[420,500)$, $B_3=[500,620)$, $B_4=[620,800]$의 네 부분으로 나눌 수 있다. 각각의 bucket B_1, B_2, B_3, B_4에 대해 α, β, γ, δ의 인덱스를 할당한다. 할당된 인덱스 α, β, γ, δ는 표 4-4에 보듯이 각각의 속성 정보 E-id_number과 E-salary의 B-index에 저장된다. 그 후 사용자는 추후 검색을 위해

$(300,420,\alpha)$, $(420,500,\beta)$, $(500,620,\gamma)$, $(620,800,\delta)$를 저장한다. 인덱스는 사용자만 알고 있는 비밀키를 포함하는 hash 함수, 난수 발생기 등을 이용한 다양한 방법으로 쉽게 생성할 수 있다.

Bucket 내의 데이터에 대한 인덱스 생성과정에서는 평문 데이터의 분포가 알려진 경우에도 안전성을 보존하면서, 효율적인 검색을 가능하게 하는 bucket 내의 데이터에 대한 인덱스를 생성한다. 사용자는 bucket $B_i = [a_{i-1}, a_i)$에 대해 bucket의 길이 $a_i - a_{i-1}$보다 큰 소수(prime) m_i를 선택하고, $0 < q_i < m_i$를 만족하는 q_i를 선택한다. 사용자는 B_i내의 데이터 t에 대해 $(t - a_{i-1}) \cdot q_i \bmod m_i$를 계산한다. Modulo multiplication을 통해 각각의 평문 데이터에 일종의 난수화가 적용되어 평문 데이터의 분포를 공격자가 알 수 없도록 변환된다. 사용자는 각각의 bucket에 대해 m_i와 q_i를 사용자만이 아는 비밀 값으로 저장한다. 이러한 과정을 통해 $B_i = [a_{i-1}, a_i)$ 내에 속해 있는 데이터는 bucket $B_i^* = [0, m_i)$ 내의 데이터로 변환된다.

예를 들어 표 4-3의 salary를 보면 $B_1 = [300, 420)$에는 340, 345, 412의 세 데이터가 포함되어 있다. 이 경우 B_1의 길이는 $420 - 300 = 120$이며, $m_1 = 487$, $q_1 = 81$으로 설정한 것으로 가정한다. 그러면, 계산 결과 340은 $(340 - 300) \cdot 81 \bmod 487 \equiv 40 \cdot 81 \bmod 487 \equiv 318 \bmod 487$이 되어 318로 변환되며, 345는 236으로, 112는 306으로 각각 변환된다. 즉 $B_1 = [300, 420)$에는 있는 데이터 340, 345, 412는 $B_1^* = [0, 487)$에 있는 데이터 318, 236, 306로 변환된다.

또한 $B_2 = [420, 500)$에 대해서는 $m_2 = 373$, $q_2 = 71$으로 설정하면, 435, 480, 480의 세 데이터는 $B_2^* = [0, 373)$에 있는 데이터 319, 157, 157로 변환된다. 유사하게 $B_3 = [500, 620)$에 대해서는 $m_3 = 523$, $q_3 = 221$, $B_4 = [620, 800]$에 대해서는 $m_4 = 811$, $q_4 = 323$을 설정하여 데이터를 변환할 수 있다.

그 다음 단계로, 사용자는 각각의 bucket들 보다 큰 크기를 지니는 하나의 구간을 설정하여, $B_i = [a_{i-1}, a_i)$로부터 변환된 $B_i^* = [0, m_i)$ 내의 데이터를 선택

한 특정 구간 내의 데이터로 변환한다. 이 특정 구간을 target bucket, $TB=[c,d]$ 라고 하며, 비밀 값 m_i들을 공격자가 알 수 없도록 하기 위해 TB의 길이는

$$|TB| = d-c \gg \max_{1 \le i \le k} m_i$$

을 만족하도록 선택한다. 여기에서 $\gg$는 매우 크다는 것을 의미한다. $B_i^* = [0, m_i)$ 내의 데이터를 $TB=[c,d]$ 내의 데이터로 변환하는 방법은 다음과 같다. $x \in B_i^* = [0, m_i)$ 에 대해 다음과 같이 정의된 함수 F는 $B_i^* = [0, m_i)$ 내의 데이터를 $TB=[c,d]$ 내의 데이터로 변환하는 선형 변환(linear transformation)임을 알 수 있다.

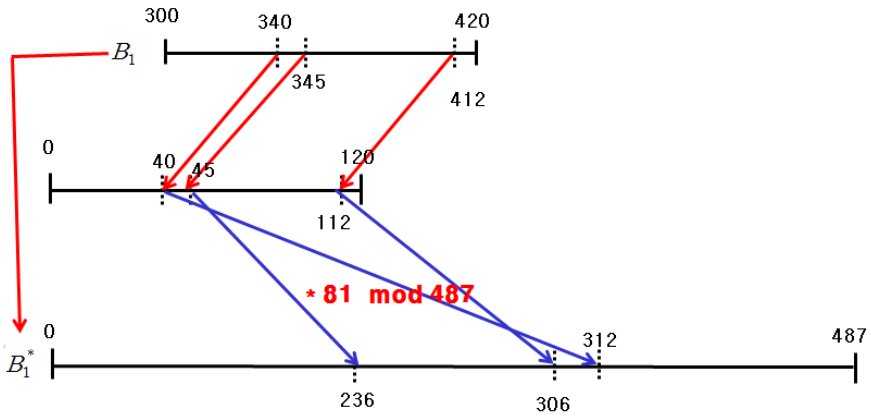
$$F_{B_i^*}(x) = c + \frac{x}{m_i} \times (d-c) \quad .$$

$y \in B_i$가 modulo 곱셈에 의해 변환된 값을 $\bar{y} \in B_i^* = [0, m_i)$ 라고 하자. 사용자는 $\lfloor F_{B_i^*}(\bar{y}) \rfloor$ 와 $\lfloor F_{B_i^*}(\bar{y}+1) \rfloor$ 을 계산한다, 여기에서 $\lfloor t \rfloor$ 은 t보다 작은 가장 큰 정수를 의미한다. 그 후 $\lfloor F_{B_i^*}(\bar{y}) \rfloor \le y^* \le \lfloor F_{B_i^*}(\bar{y}+1) \rfloor$ 을 만족하는 y^*를 랜덤하게 선택하여 $\bar{y} \in B_i^*$를 $y^* \in TB$로 변환한다. 또한, 이 값 y^*을 평문 y에 대한 bucket 내부 인덱스로 정의한다. y^*의 값을 일정 범위 내에서 임의로 선택한 것은 동일 평문이 여러 개일 때, 동일한 값을 가지는 데이터가 TP에서 동일 값으로 변환되는 것을 막기 위해서 이다. 이는 동일한 평문 데이터가 동일 암호문으로 변환되는 것으로부터 공격자가 추가적인 정보를 알 수 있기 때문이다.

표 4-3의 $B_2 = [420, 500)$ 를 예로 들어 변환 과정을 다시 살펴보면 다음과 같다. B_2에 속해 있는 세 개의 데이터 435, 480, 480는 modulo multiplication에 의해 $B_2^* = [0, 373)$ 에 속해 있는 세 개의 데이터 319, 157, 157로 변환된다. 또한 TP를 $TP=[0,10000]$로 선택하고 F함수를 아래와 같이 정의한다.

$$F_{B_2^*}(x) = \frac{x}{373} \times (10000) \quad .$$

319에 대해 $\lfloor F_{B_2^*}(319) \rfloor = 8522$, $\lfloor F_{B_2^*}(320) \rfloor = 8579$ 를 만족한다. 그러면 319를 8522와 8579 사이의 랜덤 값 8537로 변환할 수 있다. 즉, $B_2 = [420, 500)$에 있는 데이터 435는 TP내의 원소 8537로 변환됨을 알 수 있으며, 435에 대한 인덱스는 8537로써 표 4의 ind-salary에 저장된다. 이제 B_2^*에 속해 있는 동일한 두 데이터 157에 대한 변환을 살펴보자. 157에 대해 $\lfloor F_{B_2^*}(157) \rfloor = 4209$, $\lfloor F_{B_2^*}(158) \rfloor = 4235$ 를 만족한다. 사용자는 4209와 4235 사이의 두 개의 임의의 값 4211, 4221을 선택한다. 그러면 B_2에 속해 있는 동일한 두 개의 데이터 480은 B_2^*를 통해 TP 내의 두 데이터 4211과 4221로 변환됨을 알 수 있다. 따라서 두 개의 데이터 480에 대한 인덱스는 4211, 4221로서 표 4의 ind-salary에 저장된다.



[그림 4-6] 각 데이터에 modulo multiplication 적용 결과

사용자는 위의 과정을 거쳐서 생성한 암호화된 데이터베이스를 서버에 저장한다. 데이터에 대한 질의 과정은 일반적인 bucket 기반 검색 기법과 동일하다. 즉, 사용자는 자신이 저장하고 있는 bucket 범위 정보를 활용하여 자신이 원하는 데이터가 포함된 bucket을 찾고, 해당 bucket 인덱스를 포함하고 있는 데이터들을 서버에 요청한다. 이 때, bucket 인덱스 정렬 문제를 고려하여 앞 절에서 설명한 순환 버킷 질의(cyclic bucket query)를 수행할 수 있다. 서버는 사용자가 요청한

bucket 인덱스와 일치하는 데이터를 모두 사용자에 전송한다. 표 4-3의 예를 들어 사용자는 salary가 [600, 700]인 데이터를 원한다고 하자. [600,700] = [600,620) ∪[620, 700)이 되며, 사용자가 가지고 있던 bucket 정보로부터 [600, 620)⊂ [500, 620), [620, 700)⊂[620, 800] 임을 알 수 있다. 이 때 사용자는 표 4에서 보듯이, bucket [500, 620)과 [620, 800)에 대응되는 인덱스 및 데이터의 종류 정보를 서버에 전송한다. 이 경우 $(E-salary\,;\gamma,\delta)$의 질의를 서버에 하게 되며, 순환 버킷 질의를 고려하는 경우, 다음 bucket 인덱스를 포함하여 $(E-salary\,;\gamma,\delta,\alpha)$를 질의하게 된다.

서버는 사용자의 질의에 대해 해당 bucket 인덱스와 일치하는 자료를 검색하여 사용자에 전달한다. 위의 예에서 서버가 사용자로부터 $(E-salary\,;\gamma,\delta,\alpha)$를 받았다면, 서버는 표 4-4에서 E-salalry의 B-index가 γ,δ,α인 2, 3, 4, 6, 7, 8, 9, 11, 12 행을 사용자에게 전송한다.

사용자는 서버에서 전송된 암호화된 데이터 중 필요한 데이터를 얻기 위해서 다음과 같은 과정을 거치게 된다. 순환 버킷 질의를 사용한 경우 부가적으로 전송된 bucket에 포함된 데이터를 제외시키고, 비밀로 저장하고 있는 값 m_i를 호출한다. 그 다음 $B_i^* = [0, m_i)$ 를 $TP = [c, d]$로 변환하는 함수

$$F_{B_i^*}(x) = c + \frac{x}{m_i} \times (d-c)$$

의 역변환 함수

$$F_{B_i^*}^{-1}(x) = \frac{x-c}{d-c} \times (m_i)$$

을 이용하여, $TB = [c, d]$에 있는 데이터를 $B_i^* = [0, m_i)$ 로 변환시킨다. 즉 $x \in TB$에 있을 때, $\lfloor F_{B_2^*}^{-1}(x) \rfloor \in B_i^*$ 이다. 이 후, 사용자는 비밀로 저장하고 있는 값 q_i과 modulo 곱셈 과정의 식 $(t-a_{i-1}) \bullet q_i \bmod m_i$를 이용하여

$\lfloor F_{B_2^*}^{-1}(x) \rfloor \cdot q_i^{-1} \bmod m_i + a_{i-1}$를 계산하여 $B_i = [a_{i-1}, a_i)$에 속해 있는 평문 데이터를 복원한다. 여기에서 q_i^{-1} 계산은 시간을 소비하는 역원 연산이기 때문에, 사용자는 q_i^{-1}를 사전에 계산하여 비밀 값으로 저장하면 단순한 곱셈 계산으로 수행이 가능하다. 이 과정을 통해 사용자는 복원된 평문 데이터로부터 필요한 암호문에 대해서만 복호화 과정을 수행한다. 이상에서 보는 바와 같이 단순한 계산만으로 인덱스로부터 평문 복원이 가능하기 때문에, 서버로부터 전송받은 전체 암호화된 데이터 E-tuple을 복호화하는 시간에 비해 효율적으로 수행할 수 있다.

앞의 예를 다시 살펴보면 사용자는 2, 3, 4, 6, 7, 8, 9, 11, 12 행을 서버에게 전송받고, 이 중 E-salalry의 B-index가 α인 데이터는 순환 버킷 질의에 의해 부가적으로 전송받은 데이터이므로, 사용자는 3, 4, 6, 7, 11, 12 행만을 조사할 필요가 있다. 사용자는 3, 4, 6, 7, 11, 12 행의 ind-salary를 이용하여, salary가 [600, 700]에 속해 있는 데이터에 대해서만 E-tuple을 복호화하여 필요한 데이터를 출력한다. 예를 들어 3행의 ind-salary의 값은 7631이고, B-index는 γ이다. 사용자는 γ라는 인덱스를 통해 7631라는 데이터가 $B_3 = [500, 620)$ 및 $B_3^* = [0, 523)$라는 bucket에서 변환되었다는 것을 알 수 있고, 또한 $q_3 = 221$임을 알 수 있다. 우선 $TP = [0, 10000]$에서 $B_3^* = [0, 523)$로의 역변환을 통해

$$\lfloor F_{B_i^*}^{-1}(7631) \rfloor = \left\lfloor \frac{7631}{10000} \times (523) \right\rfloor = 399$$

를 얻을 수 있다. 따라서 $399 \cdot 221^{-1} \bmod 523 + 500 = 587$ 이라는 평문 데이터를 복원할 수 있으며, 이 데이터는 [600, 700]에 속하지 않으므로 E-tuple을 복호화할 필요가 없다. 이러한 과정을 통해 4행과 11행이 salary가 [600, 700]에 속해 있다는 것을 알 수 있고, 사용자는 표 4의 4행과 11행의 E-tuple만을 복호화하여 원하는 데이터를 얻을 수 있다.

이러한 방법을 통해서 사용자는 false positive로 발생한 추가적인 검색 결과들을 단순한 대수 연산을 적용하여 확인할 수 있기 때문에 기존에 전체 암호문을 복호화하는 방법으로 확인하는 과정에 비해서 매우 효율적인 bucket 기반 검색을

수행할 수 있다.

제 5 장 결 론

제 5 장 결 론

IT 환경의 급격한 발전에 따라 보안이 적용되어야 할 대상이 급속히 증가하고 있으며, Cloud Computing, RFID/USN, 차량 네트워크, 위치 기반 서비스, 사물 통신 등과 같은 신규 정보 통신 서비스 환경이 등장하고 있다. 이에 따라 개인 정보가 실시간으로 수집·저장·가공되어 서비스에 활용되면서, 서비스 이용 과정에서 개인 정보 유출에 대한 우려가 커지고 있는 실정이다.

이에 본 과제는 유비쿼터스 환경에서의 프라이버시 침해 방지를 위해 경량 암호 기술, 프라이버시 보호 핵심 암호 기술, 양자 암호 프로토콜의 세 분야를 포함하는 프라이버시 강화 암호 기술 개발을 최종 목표로 하고 있다.

당해연도에 주요 연구 추진 실적으로 경량 암호 기술 분야에서는 경량 hash 알고리즘 설계 논리 개발 및 경량 그룹 서명 프로토콜 설계에 대한 연구를 수행하였고, 프라이버시 보호 핵심 암호 기술 분야에서는 순서 보존 암호화 설계 논리 개발에 대한 연구를 진행하였다. 경량 hash 알고리즘 설계 논리 개발에 대한 연구에서는 경량 hash 후보 알고리즘 4종을 설계하였고, 경량 그룹 서명 프로토콜 설계에 대한 연구에서는 Lattice 기반의 그룹 및 링 서명, 안전한 delegation 기반의 익명 인증 시스템을 설계하였다. 또한 순서 보존 암호화 설계 논리 개발에 대한 연구 부분에서는 Bucket 기반 순서 보존 암호화 및 Pivot을 사용한 순서 보존 암호화 기법을 설계하였다.

차년도에는 당해연도에 설계한 기술에 대한 심층적인 안전성 평가 및 분석, 효율성 개선을 통한 설계안 개선 및 수정을 진행하여 최종안을 선정할 예정이며, 새로이 양자 암호 프로토콜 분야에서는 양자 서명 프로토콜 설계에 대한 연구를 수행할 예정이다.

본 과제에서 수행하는 프라이버시 강화 암호 기술 개발은 다양한 정보 통신 서비스에 기반 프리미티브 기술을 제공하여 정보 통신 서비스의 신뢰성 향상에 기여할 수 있으며, 이를 통해 관련 산업 발전 및 활성화에 기여할 것으로 전망된다.

참 고 문 헌

약 어 표

부 록

참고문헌

[ACH05] G. Ateniese, J. Camenisch, S. Hohenberger, and B. de Medeiros, “Practical group signatures without random oracles,” ePrint Archive, Report 2005/385, 2005.

[ACJ00] G. Ateniese, J. Camenisch, M. Joye, and G. Tsudik, “A practical and provably secure coalition-resistant group signature scheme,” Proceedings of CRYPTO'00, LNCS 1880, pp.255-270, 2000.

[AHM11] J. P. Aumasson, L. Henzen, W. Meier and M. Naya-Plasencia, "Quark: A Lightweight Hash", Proceedings of CHES'10, LNCS 6225, pp 1-15, 2011.

[AIK00] K. Aoki, T. Ichikawa, M. Kanda, M. Matsui, S. Moriai, J. Nakajima and T. Tokita, “Specification of Camellia-a 128-bit Block Cipher”, Proceedings of SAC'00, LNCS 2012, pp. 39-56, Springer-Verlag, 2001.

[Ajt99] Miklos Ajtai, “Generating Hard Instances of the Short Basis Problem,” Proceedings of ICALP'99, LNCS 1644, pp 1-9, 1999.

[AKS04] R. Agrawal, J. Kiernan, R. Srikant, and Y. Xu, "Order-preserving Encryption for Numeric Data", Proceedings of SIGMOD'04, pp.563-574, 2004.

[AM03] G. Ateniese and B. de Medeiros, “Efficient Group Signatures without Trapdoor,” Proceedings of ASIACRYPT'03, LNCS 2894, 246-268, 2003.

[AP09] Joël Alwen and Chris Peikert, “Generating Shorter Bases for Hard Random Lattice,” Proceedings of STACS'09, pp. 75-86, 2009.

[AST03] G. Ateniese, D. X. Song, and G. Tsudik, “Quasi-efficient revocation in group signatures,” Proceedings of FC'02, LNCS 2357, pp. 183-197, 2003.

[BB94] B. den Boer and A. Bosselaers, "Collisions for the Compression Function

of MD5", Proceedings of Eurocrypt'93, LNCS 765, pp. 293-304, 1994.

[BB04] D. Boneh, and X. Boyen, “Short signature without random oracles,” Proceedings of EUROCRYPT'04, LNCS 3027, pp. 56-76, 2004.

[BBH04] D. Boneh, X. Boyen, and H. Shacham, “Short group signatures,” Proceedings of CRYPTO'04, LNCS 3152, pp. 41-55, 2004.

[BCL09] A. Boldyreva, N. Chenette, Y. Lee, and A. O'Neill, "Order-preserving Symmetric Encryption", Proceedings of Eurocrypt 2009, LNCS 5479, pp.224-241, 2009.

[Beb02] G. Bebek, "Anti-tamper Database Research: Inference Control Techniques", Technical Report EECS 433 Final Report, Case Western Research University, 2002.

[BK10] Zvika Brakerski and Yael Tauman Kalai, “A Framework for Efficient Signatures, Ring Signatures and Identity Based Encryption in the Standard Model,” ePrint Archive, Report 2010/086, 2010.

[BKM06] Adam Bender, Jonathan Katz, and Ruggero Morselli, “Ring Signatures: Stronger Definitions, and Constructions Without Random Oracles,” Proceedings of TCC'06, LNCS 3876, pp. 60-79, 2006.

[BKN02] M. Bellare, T. Kohno, and C. Namprempre, "Authenticated Encryption in SSH: Provably Fixing the SSH Binary Packet Protocol", Proceedings of ACMCCS'02, pp.1-11, 2002.

[BMW03] Mihir Bellare, Daniele Micciancio, and Bogdan Warinschi, “Foundations of Group Signatures: Formal Definitions, Simplified Requirements, and a Construction Based on General Assumptions,” Proceedings of EUROCRYPT'03, LNCS 2656, pp. 614-629, 2003.

[Boy10] X. Boyen, “Lattice Mixing and Vanishing Trapdoors : A Framework for Fully Secure Short Signatures and more,” Proceedings of PKC'10, LNCS 6056, pp. 499-517, 2010.

[BR00] P. Barreto and V. Rijmen, "The Whirlpool Hashing Function", Submitted to NESSIE, September 2000. Revised May 2003. Available at: http://www.larc.usp.br/~pbarreto/WhirlpoolPage.html.

[BS04] D. Boneh and H. Shacham, “Group signatures with verifier-local revocation,” Proceedings of ACM CCS'04, pp. 168-177, 2004.

[BSZ05] M. Bellare, H. Shi, and G. Zhang, “Foundation of Group Signatures: The Case of Dynamic Groups,” Proceedings of CT-RSA'05, LNCS 2045, pp. 499-517, 2005.

[BW06] X. Boyen and B. Waters, “Compact group signatures without random oracles,” Proceedings of EUROCRYPT'06, LNCS 4004, pp. 427-444, 2006.

[BW07] X. Boyen and B. Waters, “Full-domain subgroup hiding and constant-size group signatures,” PKC 2007, LNCS 4450, pp. 1-15, 2007.

[Can06] C. D. Canniere, "Trivium: A Stream Cipher Construction Inspired by Block Cipher Design Principles ", Proceedings of ISC'06, LNCS 4176, pp 171-183, 2006.

[CDK09] C. D. Canniere, O. Dunkelman and M. Knerevi?, "KATAN and KTANTAN ? A Family of Small and Efficient Hardware-Oriented Block Ciphers", Proceedings of CHES'09, LNCS 5747, pp 272-288, 2009.

[CDS94] Ronald Cramer, Ivan Damgard, and Berry Schoenmakers. “Proofs of Partial Knowledge and Simplified Design of Witness Hiding Protocols,” Proceedings of CRYPTO'95, LNCS 839, pp. 174-187, 1994.

[CDV05] A. Ceselli, E. Damiani, S. D. C. di Vimercati, S. Jajodia, S. Paraboschi, and P. Samarati, "Modeling and Assessing Inference Exposure in Encrypted Databases", ACM Transactions on Information and System Security, Vol.8, No.1, pp.119-152, Feb. 2005.

[CG05] J. Camenisch and J. Groth, “Group signatures: Better efficiency and new theoretical aspects," Proceedings of SCN'04, LNCS 3352, pp. 120-133,

2005.

[CH91] David Chaum and Eug`ene van Heyst, “Group Signatures,” Proceedings of EUROCRYPT'91, LNCS 547, pp. 257–265, 1991.

[CHK10] David Cash, Dennis Hofheinz, Eike Kiltz, and Chris Peikert, “Bonsai Trees, or How to Delegate a Lattice Basis,” Proceedings of EUROCRYPT'10, LNCS 6110, pp. 523-552, 2010.

[CL02] J. Camenisch and A. Lysyanskaya, “Dynamic accumulators and application to efficient revocation of anonymous credentials,” Proceedings of CRYPTO'02, LNCS 2442, pp. 61-76, 2002.

[CL04] J. Camenisch and A. Lysyanskaya, “Signature schemes and anonymous credentials from bilinear maps,” Proceedings of CRYPTO'04, LNCS 3152, pp. 56-72, 2004.

[CM98] J. Camenisch and M. Michels, "A Group Signature Scheme with Improved Efficiency ," Proceedings of ASIACRYPT'98, LNCS 1514, pp. 160-174, Springer Verlag, 1998.

[CPH07] Giorgio Calandriello, Panos Papadimitratos, Jean-Pierre Hubaux, and Antonio Lioy, “Efficient and Robust Pseudonymous Authentication in VANET,” Proceedings of ACM VANET'07, pp. 19–28, 2007.

[CS97] J. Camenisch and M. Stadler, "Efficient Group Signature Schemes for Large Groups," Proceedings of CRYPTO'97, LNCS 1294, pp. 410-424, 1997.

[DC98] J. Daemen and C. S. K. Clapp, "Fast Hashing and Stream Encryption with PANAMA", Proceedings of FSE'98, LNCS 1372, pp 60-74, 1998.

[Dob98a] H. Dobbertin, "Cryptanalysis of MD4", Journal of Cryptology 11:4, pp. 253-271, 1998.

[Dob98b] H. Dobbertin, "The First Two Rounds of MD4 are Not One-Way", Proceedings of FSE'98, LNCS 1372, pp. 284-292, 1998.

[Dob96] H. Dobbertin, "Cryptanalysis of MD5 Compress", May. 1996. Available at: www-cse.ucsd.edu/users/bsy/dobbertin.ps.

[DP06] C. Delerablee and D. Pointcheval, “Dynamic fully anonymous short group signatures,” Proceedings of VIETCRYPT'06, LNCS 4341, pp. 193-210, 2006.

[DVJ03] E. Damiani, S. D. C. di Vimercati, S. Jajodia, S. Paraboschi, and P. Samarati, "Balancing Confidentiality and Efficiency in Untrusted Relational DBMSs", Proceedings of ACMCCS'03, pp.27-31, Oct. 2003.

[FDW04] M. Feldhofer, S. Dominikus, and J. Wolkerstorfer, “Strong authentication for RFID systems using the AES algorithm”, Proceedings of CHES'04, LNCS 5747, pp. 357-370, Aug. 2004.

[FS87] Amos Fiat and Adi Shamir, “How to Prove Yourself: Practical Solutions to Identification and Signature Problems,” Proceedings of CRYPTO'87, LNCS 263, pp. 186-194, 1987.

[GGH97] Oded Goldreich, Shafi Goldwasser, and Shai Halevi, “Public-key Cryptosystems from Lattice Reduction Problems,” Proceedings of CRYPTO'97, LNCS 1294, pp. 112-131, 1997.

[GKV10] Samuel Dov Gordon, Jonathan Katz, and Vinod Vaikuntanathan, “A group signature scheme from lattice assumptions,” To appear in ASIACRYPT'10.

[GPV08] Craig Gentry, Chris Peikert, and Vinod Vaikuntanathan, “Trapdoors for hard lattices and new cryptographic constructions,” Proceedings of STOC'09, pp. 197-206, 2008.

[Gro07] J. Groth, "Fully anonymous group signatures without random oracles," Proceedings of ASIACRYPT'07, LNCS 4833, pp. 164-180, 2007.

[GS02] Craig Gentry and Mike Szydlo, “Cryptanalysis of the revised NTRU signature scheme,” Proceedings of EUROCRYPT, LNCS 2332, pp. 299-320, 2002.

[HGP03] Jeffrey Hoffstein, Nick Howgrave-Graham, Jill Pipher, Joseph H. Silverman, and William Whyte, “NTRUSIGN: Digital Signatures using the NTRU Lattice,” Proceedings of CT-RSA'03, LNCS 2045, pp. 122-140, 2003.

[HIL02] H. Hacıgümüş, B. Iyer, C. Li, and S. Mehrotra, "Executing SQL over Encrypted Data in the Database-Service-Provider Model", Proceedings of ACM SIGMOD'02, pp.4-6, Jun. 2002.

[HMT04] B. Hore, S. Mehrotra, and G. Tsudik, "A Privacy-preserving Index for Range Queries", Proceedings of the 30th VLDB Conferences, pp.720-731, Jun. 2004.

[JBC05] 장환석, 박해룡, 천동현, 전길수, 송정환, “저가의 RFID 태그에 적합한 암호 알고리즘 구현에 관한 고찰”, 정보보호학회지, 제15권 4호, pp. 72-79, 2005.

[Jou04] A. Joux, "Collisions for SHA-0", Rump session of CRYPTO'04, 2004.

[KR00] Hugo Krawczyk and Tal Rabin, “Chameleon signatures,”Proceedings of NDSS'00, pp. 3-12, 2000.

[KTO07] Yuto Kawahara, Tsuyoshi Takagi, and Eiji Okamoto, “Efficient Implementation of Tate Pairing on a Mobile Phone Using Java,” Proceedings of CIS'07, LNCS 4456, pp. 396-405, 2007.

[LCH09] T.-F. Lee, S.-H. Chang, T. Hwang, and S.-K. Chong, "Enhanced Delegation-Based Authentication Protocol for PCSs," IEEE Transactions on Wireless Communications, vol. 8, no. 5, pp. 2166-2171, May 2009.

[LRS99] A. Lysyanskaya, R. L. Rivest, A. Sahai, and S. Wolf, “Pseudonym Systems,” Proceedings of SAC'99, LNCS 1758, pp.184-199, 1999.

[LV09] B. Libert, and D. Vergnaud, “Group Signatures with Verifier-Local Revocation and Backward Unlinkability in the Standard Model,” Proceedings of CANS'09, LNCS 5888, pp. 498-517, 2009.

[LY05] W.-B. Lee and C.-K. Yeh, "A new delegation-based authentication protocol for use in portable communication systems," IEEE Transactions on Wireless Communications, vol. 4, no. 1, pp. 57-64, Jan. 2005.

[MV03] Daniele Micciancio and Salil P. Vadhan, “Statistical Zero-Knowledge Proofs with Efficient Provers: Lattice Problems and more,” Proceedings of CRYPTO'03, LNCS 2729, pp. 282-298, 2003.

[NIS08] NIST Hash Policy. Available at: http://csrc.nist.gov/groups/ST/hash /policy.html.

[NIS10] NIST Homepage for Hash Project. Available at: http://csrc.nist.gov/ groups/ST/hash/sha-3/.

[NN04] L. Nguyen and R. Safavi-Naini, “Efficient and provably secure trapdoor-free group signature schemes from bilinear pairings,” Proceedings of ASIACRYPT'04, vol. 3329, pp. 372-386, 2004.

[NR06] Phong Q. Nguyen and Oded Regev, “Learning a Parallelepiped: Cryptanalysis of GGH and NTRU Signatures,” Proceedings of EUROCRYPT'06, LNCS 4004, pp. 271-288, 2006.

[OS03] S. C. G. Ozsoyoglu, and D. Singer, "Anti-tamper Databases: Querying Encrypted Databases", Proceedings of the 17th Annual IFIPWG 11.3 Working Conference on Database and Applications Security, 2003.

[REC04] Damith C. Ranasinghe, Daniel W. Engels, and Peter H. Cole, “Low-Cost RFID Systems : Confronting Security and Privacy”, Proceedings of Auto-ID Labs Research Workshop, September 2004.

[Reg05] Oded Regev, “On Lattices, Learning with Errors, Random Linear Codes, and Cryptography,” Proceedings of STOC'05, pp. 84-93, 2005.

[Ruc10] Markus Rückert, “Strongly Unforgeable Signatures and Hierarchical Identity-based Signatures from Lattices without Random Oracles,” Proceedings of PQCrypt'10, LNCS 6061, pp. 182-200, 2010.

[SA09] Y. Sasaki and K. Aoki, "Finding Preimages in Full MD5 Faster Than Exhaustive Search", Proceedings of Eurocrypt'09, LNCS 5479, pp. 134-152, 2009.

[Sho94] Peter W. Shor, “Algorithms for Quantum Computation: Discrete Logarithms and Factoring,” Proceedings of FOCS, pp. 124-134, 1994.

[SLW09] M. Stevens, A. Lenstra, and B. d. Weger, "Chosen-Prefix Collisions for MD5 and Colliding X.509 Certificates for Different Identities", Proceedings of CRYPTO'09, LNCS 5677, pp. 55-69, 2009.

[Szy03] Mike Szydlo, “Hypercubic Lattice Reduction and Analysis of GGH and NTRU Signatures,” Proceedings of EUROCRYPT'03, LNCS 2656, pp. 433-448, 2003.

[TTA00] Telecommunications Technology Association, Hash Function Standard Part 2: Hash Function Algorithm Standard (HAS-160), TTAS.KO-12. 0011/R1, Dec. 2000.

[TX03] G. Tsudik and S. Xu, “Accumulating composites and improved group signing,” Proceedings of ASIACRYPT'03, LNCS 2894, pp. 269-286, 2003.

[Wan97] X. Wang, "The Collision attack on SHA-0", 1997. Available at: www.infosec.edu.cn.

[Wan98] X. Wang, "The Improved Collision attack on SHA-0", 1998. Available at: www.infosec.edu.cn.

[Wan10] Jin Wang, “Ring Signature and Identity-Based Ring Signature from Lattice Basis Delegation,” ePrint Archive, Report 2010/378, 2010.

[WD08] J. Wang, and X. Du, "LOB: Bucket Based Index for Range Queries", Proceedings of the Ninth International Conference on Web-age Information Management, pp.86-92, 2008.

[WLF05] X. Wang, X. Lai, D. Feng, H. Chen, X. Yu, "Cryptanalysis of the Hash Functions MD4 and RIPEMD", Proceedings of Eurocrypt'05, LNCS 3494,

pp. 1-18, 2005.

[WY05] X. Wang, H. Yu, "How to Break MD5 and Other Hash Functions", Proceedings of Eurocrypt'05, LNCS 3494, pp. 19-35, 2005.

[WYY05a] X. Wang, Y. L. Yin, H. Yu, "Collsion Search Attacks on SHA-1", http://theory.csail.mit.edu/~yiqun/shanote.pdf, February 2005.

[WYY05b] X. Wang, A. C. Yao and F. Yao, "Cryptanalysis on SHA-1", Proceedings of CRYPTOGRAPHIC HASH WORKSHOP, October 31-November 1, 2005.

[WYY05c] X. Wang, H. Yu, and Y. L. Yin, "Efficient Collision Search Attacks on SHA-0", Proceedings of CRYPTO'05, LNCS 3621, pp. 1-16, 2005.

[WYY05d] X. Wang, Y. L. Yin and H. Yu, "Finding Collisions in the Full SHA-1", Proceedings of CRYPTO'05, LNCS 3621, pp. 17-36, 2005.

[Yao82] A. Yao, "Protocols for Secure Computations (Extended Abstract)", Proceedings of FOCS'82, pp.160-164, 1982.

약어표

AES	Advanced Encryption Standard
BC	Bit Commitment
CCA	Chosen Ciphertext Attack
CPA	Chosen Plaintext Attack
CVP	Closest Vector Problem
DDHP	Decisional Diffie-Hellman Problem
DES	Data Encryption Standard
DHP	Diffie-Hellman Problem
ECRYPT	European Network of Excellence in Cryptology
HAS	Hash Algorithm Specification
HMAC	Hash-based Message Authentication Code
IND	Indistinguishability
LWE	Learning With Error
MAC	Message Authentication Code
MANET	Mobile Ad-hoc Network
NIST	National Institute of Standards and Technology
NIZK	Non-Interactive Zero-Knowledge
OPE	Order Preserving Encryption
OT	Oblivious Transfer
PORTIA	Privacy, Obligation, and Rights in Technologies Information of Assessment

PRIME	Privacy and Identity Management for Europe
PrimeLife	Privacy and Identity Management in Europe for Life
PRNG	Pseudo Random Number Generator
RFID	Radio Frequency IDentification
ROM	Random Oracle Model
SDHP	Strong Diffie-Hellman Problem
SECOQC	Secure Communication based on Quantum Cryptography
SHA	Secure Hash Algorithm
SIS	Small Integer Solution
SMC	Secure Multiparty Computation
SoK	Signature of Knowledge
SPN	Substitution-Permutation Network
STD	Standard Model
SVP	Shortest Vector Problem
TTA	Telecommunications Technology Association
USN	Ubiquitous Sensor Networks
VANET	Vehicular Ad-hoc Network
V2R	Vehicle-to-Roadside Unit
V2V	Vehicle-to-Vehicle

부록

1. 논문 (6건)

순번	논문 제목	학술지 명칭	연도, 호	구분
1	Augmented rotation-based transformation for privacy-preserving data clustering	ETRI Journal	2010, 32(3)	SCI
2	Bounded on distributed entanglement	Journal of Physics A	2010, 43(38)	SCI
3	Improved delegation-based authentication protocol for secure roaming service with unlinkability	IEEE Communication Letters	2010, 14(9)	SCI
4	A note on parameters of random substitutions by γ-diagonal matrices	IEICE Trans. Fundamentals	2010, E93-A(6)	SCIE
5	Quantum User Authentication for Multiparty Quantum Communications	AQIS 2010	2010	비SCI
6	Unconditionally Secure User-authenticated Quantum Key Distribution	AQIS 2101	2010	비SCI

2. 특허 (국내 6건 출원, 국제 4건 출원 중)

순번	특허명	출원 번호	출원국	출원일
1	프라이버시 보호 기능을 갖는 교통 카드		한국	2010. 12.03
2	레티스 환경에서 ID 기반 대리 서명 방법		한국	2010. 12.13
3	데이터 관리 장치 및 데이터 관리 방법		한국	2010. 12.17
4	유사 난수 생성기를 활용한 pivot 결정 방법 및 이를 기반으로 한 순서 보존 암호화 기법		한국	2010 예정
5	기밀성과 무결성을 제공하는 통합 암호 모듈		한국	2010 예정
6	강한 위조 불가능성을 만족하는 래티스 기반의 링 서명		한국	2010 예정
7	Method of data encryption, queries, and encrypted data searching based on bucket over database		미국	2011 예정
8	Integrated cryptographic module providing confidentiality and integrity		미국	2011 예정
9	Strongly unforgeable ring signature from lattices in the standard model		미국	2011 예정
10	A method for determining pivots using a pseudo-random number generator and an order-preserving encryption scheme which is based on the method		미국	2011 예정

3. 기술문서 (10건)

구분	제 목	주 요 내 용	제출자	제출 일시
TDP	프라이버시 강화 암호 기술 개발 요구 사항 정의서 V1.0	프라이버시 강화 암호 기술의 사용자 및 시스템 요구 사항 정의	장구영 외 3명	2010. 08.05
TDP	프라이버시 강화 암호 기술 요구 사항 추적표	프라이버시 강화 암호 기술의 요구 사항을 추적한 표를 기술	장구영 외 3명	2010. 08.05
TDP	프라이버시 강화 암호 기술 개발 사업 프로세스 정의서 V1.0	프라이버시 강화 암호 기술 개발 사업에 대한 사업 프로세스 정의	장구영 외 2명	2010. 10.04
TM	해쉬 함수 최신 동향 분석서	해쉬 함수 설계 및 분석에 대한 최신 연구 동향 분석	장구영 외 4명	2010. 10.21
TM	Bucket 기반 인덱스 생성 기법	Bucket이라는 세부 구간 내에서 순서가 보존되는 암호화 기법을 적용하는 효율적인 암호 데이터 검색 기법 제안	장구영 외 4명	2010. 11.01
TM	Lattice 기반 그룹 서명 분석서	Lattice 기반 그룹 서명 기법 분석	장구영 외 4명	2010. 11.01
TM	그룹 서명 분석서	그룹 서명에 대한 동향 및 응용 환경에 따른 경량 그룹 서명 기법 분석	장구영 외 4명	2010. 11.01
TM	순서 보존 암호화 기법에서의 증명 가능 안전성 분석	기존에 제안된 순서 보존 암호화 기법에서의 증명 가능 안전성이 가지는 문제점을 지적하고, 이를 보완하는 새로운 순서 보존 암호화 기법 제시	조남수 외 3명	2010. 11.30
TM	Pivoting 기법을 이용한 순서 복원 가능 암호화 기법	Pivoting 기법을 사용하는 순서 보존 암호화 시스템에 masking 기법을 적용하여, 순서 복원 가능 암호 시스템 구성하는 방법을 제시	조남수 외 3명	2010. 11.30
TM	Pivoting 기반 순서 보존 암호화 기법에 대한 안전성 분석	Pivoting을 사용한 순서 보존 암호화 기법에 대한 통계적인 안전성 분석 자료 제공	조남수 외 3명	2010. 11.30

주 의

1. 이 연구보고서는 한국전자통신연구원의 주요사업으로 수행한 연구결과입니다.

2. 이 보고서의 내용을 발표할 때에는 반드시 한국전자통신연구원에서 수행한 주요사업 결과임을 밝혀야 합니다.