모바일환경에서 클라우드 컴퓨팅 보안을 위한 효율적인 암호화기술

An Efficient Encryption Technique for Cloud-Computing in Mobile Environments

황 재 영*, 최 동 욱*, 정 연 호**
Jae-young Hwang*, Dong wook Choi*, Yeon-ho Chung*

요 약

본 연구에서는 보안 알고리즘을 구현하여 데이터에 대한 기밀성과 안정성을 확보하며 특히 클라우드 컴퓨팅 중 모바일 환경에 적합한 효율적인 암호화 기술을 제안하고자 한다. 이를 위해 PC 환경에서 알고리즘을 구현하고 기존의 암호화 알고리즘인 DES (Data Encryption Standard)와 비교 검증하였다. 기존 DES 암호화의 경우 초기 치환과 최종 치환이 공개되어 있어 알고리즘의 기밀성을 유지하기가 어려운 단점이 있다. 최근에는 이러한 DES 의 단점을 보완하기 위해 일반적으로 3중 DES 기법이 사용되나 암호화 시간이 길어지는 단점이 있다. 본 연구에서는 랜덤 인터리빙 (random interleaving) 알고리즘을 적용하여 단순히 치환에 그치는 것이 아니라 의사 난수를 이용한 치환 테이블을 적용함으로써 기밀성을 높였으며, 기존의 3중 DES 에 비하여 암호화 속도가 빠르고 무선 환경에서 높은 안정성을 제공한다.

Abstract

In this paper, we propose an efficient encryption algorithm for ensuring data privacy and security for cloud computing in mobile environments. As part of the evaluation of the proposed algorithm, we have implemented the algorithm in a PC environment and compared with the well-known encryption algorithm of the Data Encryption Standard (DES). The conventional DES algorithm is hard to maintain privacy, due to the fact that its initial and final permutation are known to the network. To prevent this critical weakness, a triple DES algorithm has been reported, but it has a disadvantage of long encryption time. In this study, we propose random interleaving algorithm that uses the permutation table for improving privacy further. The proposed algorithm is found to run faster than the triple DES algorithm and also offers improved security in a wireless communication system.

Keywords : DES, Interleaving, BBS, Cloud Computing

I. 서 론

최근 인터넷의 이용 활성화와 더불어 UCC, IPTV, 영상 콘텐츠 등 대용량 데이터가 증가하여 이에 대한 저장 및 처리의 필요성이 증대하고 있다. 웹하드 CDN (Content Delivery Network) 업체들의 수요 증가 및 기업의 IT 아웃소싱 확대로 인해 IDC (Internet Data Center)의 중요성이 점차 커지고 있으며, 기업의 응용프로그램이 점차 커져 서버의 고사양화가 진행되고 있다. 이와 같은 인터넷의 환경 변화에 따라 가상화 기술이 새로운 트렌드로 부상하고 있으며 이를 이용한 대표적인 기술로 클라우드 컴퓨팅 (Cloud Computing) 이 주목을 받고 있는데 "제2의 디지털 혁명" 이라고까지 불리며 새로운 컴퓨팅 플랫폼으로 평가받고 있다 [1]. 그러나 클라우드 컴퓨팅은 IT 자원을 소유하지 않고 일부 또는 전부를 아웃소싱하는 형태로 보안 문제가 중요시된다. 클라우드 컴퓨팅을 통해 데이터가 연동되고 자원을 다양하게 활용하는 것에는 데이터 보호와 자원의 관리 정책, 기업 비밀 관리나 개인의 프라이버시 측면에서의 문제점이 있다. 특히, 개인 및 기업 데이터에 대한 기밀성 (privacy) 보호를 위해서는 기본적으로 암호화 (encryption) 기술이 제공되어야 한다. 또한 클라우드 컴퓨팅의 경우 대용량 데이터 암호에 따른 고속 처리와 효율적인 키관리 기술, 빈번한 사용자 접근에 대한 인증 기술 및 보다 더 강력할 것으로 예상되는 DDoS 공격에 대한 대응 기술들이 요구된다 [2]. 기존의 클라우드 컴퓨팅 환경에서 주로 이용되는 DES 알고리즘의 경우 암호화 키 크기가 56 비트(패리티 비트 제외) 로 전수 조사를 할 경우 2^{56} 개의 키를 조사함으로써 구할 수 있게 된다. 현재의 기술로 병렬처리가 가능한 컴퓨터를 이용할 경우 단시간에 모든 경우의 수를 검색할 수 있는 단점을 가진다. 이를 보완 하기위

* 부경대학교 ** 부경대학교(교신저자)
투고 일자 : 2011. 8. 13 수정완료일자 : 2011. 10. 26
게재확정일자 : 2011. 11. 1
* 이 논문은 2010년도 부경대학교의 지원을 받아 수행된 연구임. (PK-2010-093)

해 3중 DES가 제시 되었으나 시간이 많이 걸리는 단점을 가지고 있다 [3]. 본 연구에서는 기존의 DES 암호화 알고리즘의 안전성 측면에서 단점으로 지적되고 있는 초기치환 및 최종치환을 랜덤 인터리버로 대체한다. 이를 통해 데이터에 대한 기밀성과 안정성을 확보하고 3중 DES에 비해 빠른 구현 속도를 제공할 뿐만 아니라 클라우드 컴퓨팅 중 모바일 환경에 적합한 효율적인 암호화 기술을 제안한다. 그리고 이러한 연구는 향후 스마트폰 시장에서의 클라우드 컴퓨팅 및 어플리케이션의 보안상 문제점에 대한 해결 방안도 제시할 수 있을 것으로 사료된다. 보안 알고리즘에 적용된 랜덤 인터리버의 경우 잡음에 강하고 높은 기밀성을 제공하는 특징을 가지고 있으며 본 연구에서 랜덤 인터리빙 구현은 Bulm Bulm Shub (BBS) 생성기를 이용하였다 [4]. 2장에서는 제안하는 클라우드 컴퓨팅 보안 기술을 기술하고, 시뮬레이션 및 성능비교를 3장에서 설명하며, 4장에서 결론을 맺고자 한다.

II. 클라우드 컴퓨팅 보안기술

1. BBS (Blum Blum Shub) 생성기

BBS 생성기는 그림 1에서와 같이 동작하며 0 과 1 로 된 비트열을 생성한다. BBS 생성기의 구체적인 동작은 다음과 같다. 먼저 4k + 3 형태 (k 는 임의의 정수) 를 갖는 두 개의 매우 큰 소수 p와 q를 찾는다. 그림 1 에 표시되어 있는 n 값은 p와 q의 곱으로 구한다. 그 다음, n 과 서로 소인 임의의 수 r 을 선택하게 된다. 종자값 (seed) 으로 쓰일 X_0 의 값은 r^2 를 n으로 나눈 나머지를 계산하여 구한다. 그 다음의 종자값은 현재의 값의 제곱을 n 으로 나눈 나머지를 이용하여 계산한다. 위와 같은 과정을 통하여 생성된 난수 정수열의 가장 낮은 자리 수를 추출하여 그것을 난수 비트로 취하여 생성하게 된다.

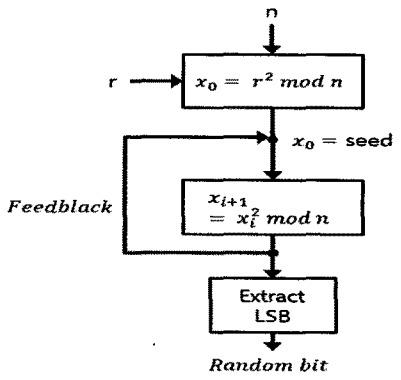


그림 1. Blum Blum Shub 알고리즘

Fig. 1. Algorithm of Blum Blum Shub

여기서, 만약 p와 q 값이 노출되어 있다면 가능한 x0 값을 시도하여 i번째 비트 값을 알아낼 수 있다. 따라서 생성기의 복잡도는 n 을 인수분해하는 것과 동일하다는 의미이다. 그러나 n 이 충분히 큰 수라면 이 수열은 예측 불가능하여 안전한 수열이 된다. n 의 값이 크다면 이전의 모든 비트를 알고 있다고 해도 다음 비트 값을 추출할 수 없으며 각 비트가 0이 되거나 1이 될 확률은 약 50% 로서 우수한 성능을 가지고 있다 [5].

2. Random Interleaver

현재의 디지털 이동 통신 시스템이나 이동 위성 통신과 같은 무선 시스템에서는 페이딩과 간섭, 그리고 부가 백색 가우시안 잡음 (AWGN)등에 의해 통신 채널 상에서 발생한 오류가 연집성을 지니고 있어 복호기 출력값의 최대 근사화비 (Maximum Likelihood Ratio)가 서로 상관 관계를 갖게 된다. 이런 상관 관계를 지닌 정보가 다음 단의 반복 복호기의 입력으로 사용되면 성능 열화 현상이 발생하게 된다. 그러므로 상관 관계가 있는 정보를 최대한 효과적으로 변환하여 상관 관계를 줄이고 연집 오류 (burst error)를 산발 오류 (random error) 로 바꾸어 주는 것이 중요하며 이러한 역할을 하는 것이 인터리버 (interleaver) 이다 [6]. 본 연구에서는 무선 환경에 효율적인 인터리버로 높은 랜덤성을 가지는 랜덤 인터리버를 적용하였다 [7].

그림 2 는 구현한 랜덤 인터리버 알고리즘을 보여주고 있다.

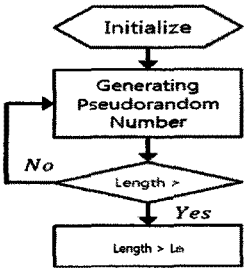


그림 2. 랜덤인터리버 순서도

Fig. 2. Flowchart of random interleaver

그림 2에서 보면, 초기화 과정에서 생성할 랜덤 비트 개수 (L_{th}) 를 설정하고 의사난수를 발생시킨다. 설정된 개수의 의사난수가 생성이 되면 저장한 후 종료된다.

3. DES (Data Encryption Standard) 알고리즘

미국 국립기술표준원에서 1977년 표준화한 암호화 알고리즘으로 64비트 평문을 이용하여 64비트 암호문을 생성한다. 두 개의 치환 (초기 및 최종) 과정과 16개의 Feistel [8] 라운드 함수로 구성되어 있으며 각 라운드는 키 생성기에 의해 암호키로 부터 생성된 48비트 라운드키를 사용한다 [9]. DES 암호화 과정은 그림 3에서 보여주고 있다.

그림 3에서 보면 먼저 64 비트 크기의 평문 입력과 64 비트 암호 키에서 패리티 비트가 제거된 56비트의 암호키가 입력이 된다. 56비트 암호키는 라운드키 생성기를 통하여 각 라운드에 사용될 48비트 길이의 라운드 키를 만들게 된다. 최초 암호키에서 28 비트 단위로 나누어 각각 쉬프팅하는 과정을 통하여 라운드 키가 생성이 된다.

64비트 입력 평문은 라운드 함수를 거치기전 초기치환 과정을 거침으로써 데이터를 재배열한다. 재배치된 데이터는 Feistel 암호를 이용하여 만들어진 라운드 함수를 16회를 거치게 되고 최종치환을 통해 초기치환으로 재배치된

데이터를 복원하게 된다.

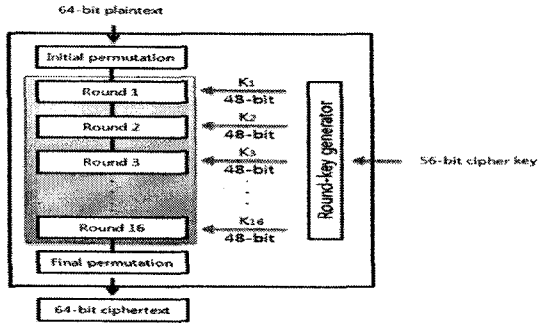


그림 3. DES 알고리즘

Fig. 3. Algorithm of DES

Initial Permutation	Final Permutation
58 50 42 34 26 18 10 02	40 08 48 16 56 24 64 32
60 52 44 36 28 20 12 04	39 07 47 15 55 23 63 31
62 54 46 38 30 22 14 06	38 06 46 14 54 22 62 30
64 56 48 40 32 24 16 08	37 05 45 13 53 21 61 29
57 49 41 33 25 17 09 01	36 04 44 12 52 20 60 28
59 51 43 35 27 19 11 03	35 03 43 11 51 19 59 27
61 53 45 37 29 21 13 05	34 02 42 10 50 18 58 26
63 55 47 39 31 23 15 07	33 01 41 09 49 17 57 25

그림 4 초기 및 최종 치환 테이블

Fig. 4. Table of initial and final permutation

그림 4에서 보면 최종치환은 단순히 초기치환을 통하여 재배열된 데이터를 원래의 상태로 되돌리는 역할만을 수행한다. 그림 5는 라운드 함수의 내부 알고리즘을 보여주고 있다. 첫 번째 라운드에서는 초기치환을 거친 평문이 입력이 되고 2번째부터 16번째의 라운드 함수에는 이전 라운드 함수의 출력값이 입력이 된다. 입력 데이터는 32 비트 단위로 나뉘어 혼합기 (Mixer)와 스와퍼 (Swapper)를 통하여 섞는 과정을 거치게 된다. 여기서 $f(R_{I-1}, K_I)$ 함수는 DES 함수로서 우측 32비트 (R_{I-1})와 라운드키 (K_I)를 입력으로 받아 수행한다.

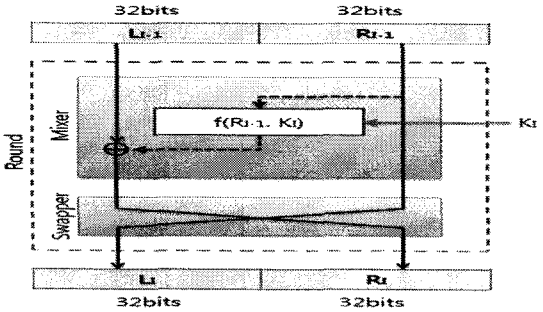


그림 5. 라운드 함수

Fig. 5. Round function

DES 함수의 구체적인 동작은 그림 6에서 보여주고 있다. 입력된 32 비트 데이터 (R_{I-1}) 는 48 비트로 확장 치환하는 Expansion P-box 과정을 거치고 난 뒤 라운드 키 (K_I) 와 배타적 논리합 (eXclusive OR) 을 수행하며 혼돈 (confusion) 과정 [2, 9] 을 수행한다. 그 다음에 straight P-box 에서 정해진 테이블을 이용하여 다시 재배열 과정을 거치게 된다.

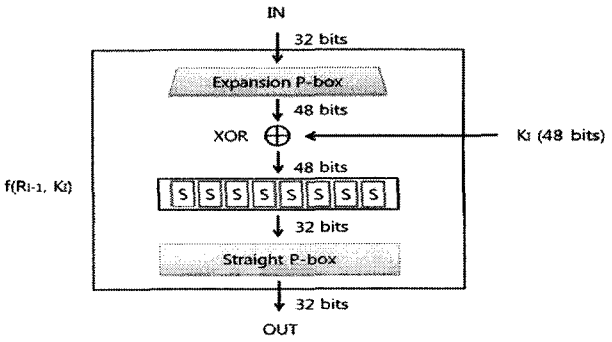


그림 6. DES 함수

Fig. 6. DES Function

III. 실험 및 고찰

제안된 알고리즘을 구현하기 위해 Microsoft에서 개발된 언어인 C#과 Visual Studio 2010 professional 컴파일러를 사용하였다. 본 논문에서는 기존의 DES 알고리즘에서 초기 치환 및 최종 치환에서 기존의 테이블을 대신하여 랜덤 인터리버를 이용하여 구현하였다. 여기서 랜덤 인터리빙에서 생성된 길이는 DES 서 정의된 64 비트를 치환하기 위해서 64 개의 값을 생성하게 된다. 제안한 알고리즘의 흐름도는 그림 7에서 보여주고 있다. 암호화가 시작되면 초기화를 통해 암호화될 문자와 암호키를 설정한다. 여기서 각 라운드에 사용될 라운드 키는 암호 키를 이용하여 생성한다. 그 다음에, 랜덤 인터리빙을 이용하여 초기치환을 수행하고 16번의 라운드 함수를 수행한다. 라운드 함수가 수행되고 나면 초기 치환의 역순으로 최종치환을 수행하고 최종적으로 암호화된 64비트 값을 얻게 된다. 본 연구에서는 전술한 BBS 의사 난수 생성기의 종자값 (그림 1 의 X_0) 은 그림 3에서 언급한 DES 암호화 알고리즘에서 사용된 키 값 (즉, 그림 3 의 56 비트 암호화 비트와 패리티 비트를 합한 64 비트) 을 이용하여 생성하였다. 그리고 인터리빙 과정에서 사용된 정렬 방식은 버블 정렬을 이용하였는데 버블정렬은 식(1) 에 의하면 항상 일정한 시간 복잡도를 가지고 길이 n 값이 상대적으로 작은 64 이어서 효율적인 정렬 방식으로 알려져 있다 [7].

$$\sum_{i=1}^{n-1} i = \frac{n(n-1)}{2} = O(n^2) \qquad (1)$$

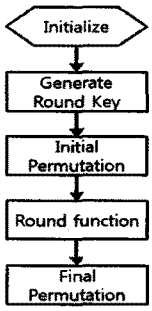


그림 7 제안된 알고리즘 흐름도
Fig. 7. Flowchart of proposed algorithm

기존의 연구에서는 입력받은 평문의 초기치환을 위해 이미 공개된 치환 테이블에 의하여 데이터를 재배치하지만, 제안된 알고리즘에서는 BBS 난수 생성기의 난수를 이용하여 치환테이블을 생성한다. 생성된 치환테이블에 따라 평문을 재배열하고 라운드 함수를 거쳐 이를 다시 원래의 배열 순서로 치환하기 위해 최종치환 과정을 거치게 되면 암호문이 완성된다. 기존의 DES의 경우 치환 테이블이 고정되어있어 같은 평문에서 초기치환 과정을 거칠 경우 항상 같은 치환결과가 도출되어 DES 알고리즘에서 안정성 측면에서 의미를 부여하지 못한다. 그러나 제안한 암호화 알고리즘은 의사 난수 생성기를 이용하여 초기치환과 최종치환 테이블을 생성하므로 같은 평문을 치환할 경우 2^{64} 가지의 경우의 수가 생성되고 이는 16개의 라운드 함수를 거쳐 암호화가 진행이 된다. 정보이론 관점에서도 공개된 테이블의 경우 엔트로피 값은 0 bit 이나 식 (2)에서 보듯이 난수 생성기에 의해 생성된 테이블의 경우 6 bits 의 엔트로피 값을 가진다. 이때 i의 값은 테이블의 각각의 순서이다. [10]

$$H(S)=\sum_{i=1}^{64}\frac{1}{64}log_2\frac{1}{64}=6\,(bits) \qquad (2)$$

그림 8과 그림 9는 기존의 DES, 3중 DES 그리고 제안한 DES 알고리즘을 치환 및 암호화 속도를 중심으로 각각 비교 및 분석한 것이다. 여기서 x축은 암호화를 하는 평문 데이터 크기를 나타내고 y 축은 평문 데이터를 암호화 하는데 걸리는 암호화 소요 시간이다.

그림 8에서 볼 수 있듯이 치환 속도측면에서 기존의 DES 속도가 나은 것을 볼 수 있으나, 전술한 바와 같이 기밀성과 보안성이 낮아서 실질적으로는 3중 DES 를 많이 적용하고 있다. 따라서 3중 DES 와 제안하는 DES 알고리즘과의 비교에서는 제안하는 알고리즘이 더 적은 시간이 소요됨을 알 수 있다. 그리고 보안성 측면에서는 3중 DES 가 장점을 가지고 있으며 제안하는 알고리즘도 랜덤 인터리버를 통한 기밀성 및 보안성은 3중 DES 와 비교할 만하다고 할 수 있다. 그림 9 에서는 전체적인 암호화 속도에 따른 DES 알고리즘의 비교 및 분석을 보여주고 있다. 기존 DES 보다는 전체적인 시간이 더 많이 걸리지만 현실적인 DES 알고리즘인 3중 DES 보다는 우수한 암호화 속도를 보여주고 있음을 알 수 있다.

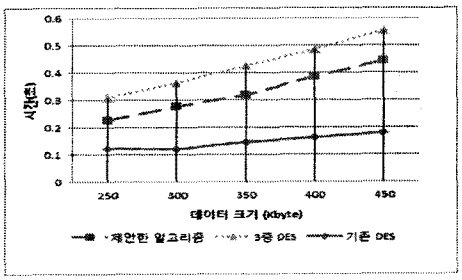


그림 8. 기존 DES, 3중 DES와 제안된 DES의 치환 속도 비교
Fig 8. Comparison of conventional DES, triple DES and proposed DES relative to permutation speed

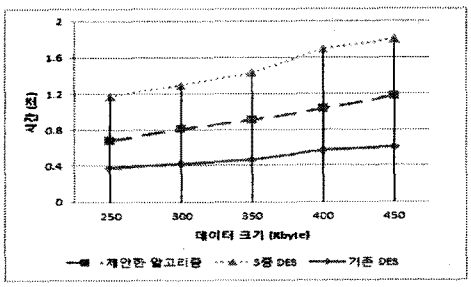


그림 9. 기존 DES, 3중 DES와 제안된 DES 의 암호화 속도 비교
Fig 9. Comparison of conventional DES, triple DES and proposed DES relative to encryption speed

IV. 결 론

기존의 DES 알고리즘은 보안성 및 기밀성 측면에서 취약하여 무선 환경에서 적용은 적합하지가 않다. 이러한 DES 알고리즘의 문제점으로 인하여 DES 를 3중으로 구현하거나 보다 개선된 알고리즘을 사용하고 있다.

그러나 3중 DES의 경우 키의 크기가 168 비트로 커져 DES에서의 전수 조사 공격으로 인한 문제는 감소하나 DES 암호화 및 복호화를 3번 수행해야 하므로 많은 시간과 비용이 발생하게 된다[11]. 본 연구에서 DES 의 치환 테이블을 의사 난수 생성기를 이용하여 생성함으로써 안정성을 높일 수 있었다. 제안한 알고리즘과 치환 및 암호화 속도 측면에서의 비교 분석에서 제안한 DES 알고리즘의 장점을 확인할 수 있었다. 이는 보안성에 대한 희생을 최소화하면서 얻을 수 있었다 . 향후 과제로서는 랜덤 인터리버 뿐만 아니라 DES 알고리즘의 라운드 함수를 병렬로 처리하여 치환 및 암호화 소요시간을 더욱 단축하고 기밀성 및 보안성이 우수한 암호 알고리즘을 구현할 계획이다.

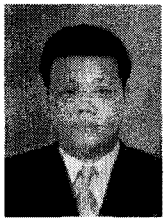
참 고 문 헌

[1] 김지연 외 3명, "클라우드 컴퓨팅 환경의 가상화 기술 취약점 분석연구", 한국정보보호학회, 제 19권 제 4호, pp. 72-77, 2009
[2] 박춘식, "클라우드 컴퓨팅에서의 보안 고려사항에 관한 연구", 한국산학기술학회논문지, 제12권, 9호, pp.44, 2011

[3] Behrouz A. Forouzan, "암호학과 네트워크 보안", Mcgraw-Hill Korea, 2008
[4] L. Blum, M. Blum, and M. Shub, "A Simple Unpredicatable Pseudorandom Number Generator", SIAM Journal on Computing, Vol. 15, No.2, pp. 364-383, 1983
[5] 신상호, 최장희, 유기영, "난수 및 의사난수행성기에 대한 평가도구의 분석", 대한전자공학회 하계학술대회, 제33권 1호, pp. 1648-1651, 2010
[6] Juha Heiskala, "OFDM 무선랜", 브레인코리아, 2003
[7] 진익수, 노예철, 주유상, 강법주, "터보 부호의 인터리버 분석", 정보통신산업진흥원, 주간기술동향 905호, 1999
[8] 원동호, "현대 암호학", 도서출판그린, 2005
[9] National Institute of Standards and Technology, http://www.nist.gov
[10] Thomas M. Cover, Joy A. Yhomas, "Elements of Information Theory", Wiley, 2005
[11] 이완복, 김정태, "파이프라인 구조의 3DES 암호알고리즘의 설계 및 구현", 해양정보통식학회논문제 제10권 제2호, pp.333-337, 2006

황 재 영(Jae-young Hwang)

2010년 부경대 전자정보통신공학과(공학사)
2010년~현재 부경대 정보통신공학과 석사과정
※주관심분야 : 이동통신, 클라우드컴퓨팅



최 동 욱(Dong wook Choi)

2005년 부경대 전자정보통신공학과(공학사)
2009년~현재 부경대 정보통신공학과 석사과정
※주관심분야 : 채널코딩, 다중접속기술

정 연 호 (Yeon ho Chung)
正會員

1984년 경북대학교 전자공학과 (공학사)
1992년 The Imperial College, University of London, U.K. (공학석사)
1996년 Liverpool University, U.K.(공학박사)
1995년 영국 Freshfield Communications Ltd. 연구원
2004년 영국 Plymouth University 초빙연구원
2006년 미국 펜실베니아주립대학교 객원교수
2001년-현재 부경대학교 정보통신공학과 교수
※주관심분야 : 적응변조 및 부호화기술, OFDM, IDMA