招請論文

일반화된 Diffie-Hellman 키이분배방식의 안전성 분석

正會員 李 弼 中* 正會員 林 采 薰*

(On Security Analysis of Generalized Diffie-Hellman Key Distribution Systems)

Pil Joong LEE* Chae Hoon LIM* *Regular Members*

要 約 본 논문에서는 관용 암호시스템을 위한 키이관리 방법으로 1976년 Diffie와 Hellman이 처음 제안한 유한체 상에서 이산대수 문제의 어려움에 바탕을 둔 공개 키이분배 방식(DH KDS로 약칭)의 각종 변형들에 대해 그들의 안전성을 중점적으로 분석하여 보다 안전한 시스템을 설계하거나 설계한 시스템을 분석하는데 필요한 체계적인 접근법을 제시하고자 한다. 키이분배 방식에의 가능한 공격방법들을 분류하여 그들에 의해 깨어질 수 있는 시스템들을 기존 방식이나 설계 예를 통해 살펴봄으로써 이들을 피할 수 있는 방법들을 알아본다. 보안 안전성 분석의 도구로서 reducibility test 나 정보이론적 접근법, 그리고 프로토콜 분석법 등을 소개하고 이들을 각종 DH KDS의 변형들에 적용하여 그 안전성 여부를 결정한다.

ABSTRACT As an elegant solution of the key management scheme for a conventional cryptosystem, Diffie and Hellman introduced a public key distribution system(DH KDS, for short), whose security depends on the intractability of discrete logarithm problem over a finite field, and since then a lot of variants of DH-KDS have been proposed. In this paper, we present the systematic approach to analyzing the security of a generalized DH KDS and designing an efficient and secure scheme. We classify various attacking methods and point out a possible way to avoid these attacks through the examples of successful attack against those systems proposed so far or designed for this purpose. As security analysis tools, we present the reducibility test, the information-theoretic approach, and the protocol analysis technique, which we apply to variations of DH scheme to examine their security under all possible attacks.

1. 서 론

컴퓨터 통신망을 이용한 정보교환이 일반화되어감에 따라 특별히 보호되지 않는 통신로상에서의 정보 유출이나 불법 변경 혹은 상대방에 대한 인증 등의 정보 보안에 관한 문제가 새로이 제기되었고 이를 해결하는 가장 강력한 도구로서 암호법(cryptography)에 대한 연구가 널리 진행되어 왔다. 이 암호법은 크게 암호화(encryption) 및 복호화(decryption)에 동일한 키이를 사용하므로 비밀통신을 하고자 하는 두 사용자가 공통의 비밀키이를 공유해야 하는 관용 암호시스템(conventional cryptosystem)과, 계산상 풀기 어려운 문제(computationally intractable problem)를 이용하여 암호화 키이(encryption key)를 공개하더라도 이로부터 복호화 키이(decryption key)를 유도해 낼 수 없도록 함으로써 암호화 키이를 공개하여 누구나 원하는 상대방에게 비밀 메세지를 보낼 수 있게 해주는 공개키이 암호시스템(public key cryptosystem)[DH76]으로 나눌 수 있다. 속도나 자원(resource)의 사용 및 구현방법 등에 있어서 관용 암호시스템이 공개키이 암호시스템에 비해 우수하나 가장 문제가 되는 것이 키이관리(key management)이다. 즉 암호시스템을 사용하는 사회가 커질수

*浦項工科大學 電子電氣工學科
Dept. of Electronics and Electrical Engineering, POSTECH
論文番號 : 91-55

록 보내는 사람과 받는 사람만이 공유 하도록 키이를 관리하는 것이 어려워진다.

키이관리 문제를 해결하는 한 방안으로 관용 암호시스템을 이용하여 키이분배 센터(key distribution center : KDC)에서 모든 사용자들의 키이를 일괄적으로 분배 관리하는 방법을 들 수 있으며 구체적인 예로 IBM(International Business Machines)의 키이관리 방법[IBM78] 및 국제 표준화 기구(ISO : International Organization for Standardization)의 금융망 키이관리(BankingKey Management) 표준안 [ISO88] 등이 있다. 각 사용자는 KDC와의 공유 키이(terminal key or key-encrypting key)를 가지고 있으며 이는 실제로 암호통신에 사용되는 키이(session key or data key)를 분배해 주는데 사용된다. 즉 비밀통신을 하고자 하는 사용자는 상대방의 주소와 함께 KDC에 키이분배를 요청하면 KDC에서는 세션키이를 발생시켜 이를 두 사용자의 터미날키이로 각각 암호화하여 키이 분배를 요청한 사용자에게 전송하게 된다. 그러나 이 방식에는 여전히 해결되지 않은 몇 가지 문제점이 남아있다. 첫째는 터미날키이의 분배문제로 여전히 인가된 안전요원(authorized personnel) 에 의한 직접 분배(manual distribution of terminal keys)가 필요하다는 점이며 둘째로 KDC가 모든 사용자들의 세션키이를 발생 분배해 주므로 KDC에 대한 절대적인 신뢰를 기반으로 해야 한다는 점이다. 이 문제점들로 인해 KDC를 이용한 키이분배 방식은 기업체의 사설망이나 금융망등과 같은 폐쇄 사용자 그룹(closed group of users)에 주로 이용되고 있으며 보다 일반적인 공중통신망에서의 암호시스템을 위한 키이분배와 같은 범용의 키이분배 방식으로서는 한계가 있다.

보다 일반적인 키이분배 방식으로 Diffie-Hellman 방식(DH-KDS)으로 대표되는 공개키이분배 방식(public key distribution system)을 들 수 있다. 이는 1976년 Diffie와 Hellman [DH76]이 처음 제창한 이래 많은 연구가 진행되어온 분야로 공개키이 암호시스템의 개념을 이용함으로써 KDC를 이용한 키이분배 방식에서의 여러 결점들을 극복하였다. 각 사용자는 자신의 비밀키이에 대응하는 공개키이를 공개키이 디렉토리(public key directory)에 등록하여 누구나 이용할 수 있게 하며 원하는 상대방과의 세션키이는 이 공개키이 디렉토리를 엑세스(access)함으로써 계산할 수 있게 된다. 여기서 중요한 것은 원하는 상대방의 정확한 공개키이를 얻는 것으로 실제 구현을 위해서는 공개키이 디렉토리의 관리가 중요한 문제가 된다[D83]. 이 Diffie-Hellman 형의 키이분배 방식에 대해서는 2장에서 상세히 다루기로 한다. 이 Diffie-Hellman 방식의 변형[ITW82][Mi85][BW88][OVS84] 외에도 permutation polynomial 등을 이용한 공개 키이분배 방식들도 제안되고 있다[N86][V87].

공개 키이분배 방식의 변형으로 개인정보(ID : identification information)를 이용한 키이분배 방식(ID-based KDS)을 들 수 있다. 이는 1984년 Shamir[Sha84]에 의해 제안된 개인정보에 바탕을 둔 암호시스템(ID-based cryptosystem)의 일종으로 누구나 알 수 있고 또한 그 사람을 유일하게 식별해 줄 수 있는 ID(예를 들면 성명, 주민등록번호, 주소 등)를 공개키이로 사용함으로써 공개키이 디렉토리를 없앨 수 있다는 장점을 가지고 있다. 이는 Shamir 이전에 Kohnfelder [K78]나 Blom[B182][B184] 등에 의해 부분적으로 다루어진 내용들이나 Shamir에 의해 개념이 체계화되고 중요성이 인식되어졌다. ID를 이용한 키이분배 방식에는 크게 Okamoto 방식[OT89]과 같이 사전 통신(preliminary communication)이 필요한 방식과, Blom의 Symmetric Key Generation System(SKGS)[B184]이나 Matsumoto-Imai의 Key Predistribution System(KPS)[MI87] 등과 같이 사전 통신이 필요없이 센터로부터 분배받은 자신의 비밀정보와 다른 이용 가능한 공개정보들을 사용하여 세션키이를 계산하는 방식(noninteractive KDS)으로 나눌 수 있다. 전자인 Okamoto 방식의 경우는 RSA와 Diffie-Hellman 방식이 결합된

형태로 공개키이 디렉토리가 필요없이 smart card의 형태로 구현하기에 적절한 효율적인 방식이라고 할 수 있으나 사전 통신을 통하여 세션키이를 형성한다는 점에서 ID에 바탕을 둔 암호시스템의 원래 개념과는 약간 벗어난 것으로 ID를 이용한 Diffie-Hellman방식의 변형에 가깝다고 할 수 있다. 한편 후자의 경우는 사전 통신이 필요없다는 장점이 있는 반면 이는 잘 알려진 비밀공유 방식(secret sharing scheme or threshold scheme)[Sha79]과 유사한 것으로 대부분의 경우 일정수의 사용자들이 결탁(conspiracy)하면 센터의 비밀정보나 다른 사용자들의 비밀키이를 계산해 낼 수 있다는 문제점이 제기된다[T87][TI89][MI90]. 뿐만 아니라 모든 ID에 바탕을 둔 암호시스템이 가정하고 있는 것처럼 절대적으로 신뢰할 수 있는 카드발급 센터(trusted card issuing center)의 존재는 이 시스템이 Diffie-Hellman 방식처럼 공중 통신망에서와 같은 범용의 키이분배 방식으로는 적합하지 않다는 것을 의미한다. 그러나 터미날키이나 사전통신이 필요없다는 점에서 KDC를 이용한 키이분배 방식보다는 훨씬 개선된 것으로 폐쇄 사용자 그룹용에 이상적인 방식이라고 할 수 있다.

마지막으로 Fiat-Shamir법[FS86]으로 대표되는 개인식별(identification) 및 디지탈서명(digital signature)의 기초로 최근에 와서 활발히 연구되고 있는 zero knowledge 이론[GMRa85]을 이용한 키이분배 방식을 들 수 있다[BK89][OO90]. 이 방식의 잇점은 전혀 정보가 새어나가지 않는다는 안전성 증명에 있으나 이를 위해 치루어야할 댓가, 즉 다른 방식의 몇 배에 해당하는 통신량과 횟수 및 계산량을 고려한다면 특수한 경우를 제외하고는 이를 범용의 키이분배 방식에 응용하는 것은 아직까지는 경제성 및 효율성 등에 비추어 적합하지 않은 것으로 보인다.

이상에서 간략하게 살펴보았듯이 정보보호를 위해서는 관용 암호시스템을 사용하더라도 키이의 관리를 위해서는 공개키이 암호시스템을 이용하는 것이 바람직하며, 특히 컴퓨터통신이 일반화되는 정보화 사회의 공중 통신망과 같은 범용의 컴퓨터망에서 각 가입자의 개인적인 정보보호를 위해 정보보안 서비스를 제공하고자 할 경우의 키이관리 방식으로는 공개 키이분배 방식이 가장 적절할 것이다. 한편 각 계열사나 해외지점 등을 연결하는 다국적 기업의 사설망이나 은행등의 금융망과 같은 폐쇄 사용자 그룹의 경우 중앙의 본부가 모든 가입자들이 신뢰할 수 있는 카드발급 센터의 기능을 할 수 있으므로 보다 키이관리가 간단한 ID에 바탕을 둔 키이분배 방식을 채택하는 것이 바람직하며 또한 암호시스템의 선도적인 이용계층이 이들인 점을 고려한다면 이에 대한 연구가 활발히 진행되고 있는 것은 당연한 결과일 것이다.

본 논문에서는 범용의 키이관리 방식으로 가장 주목받고 있는 Diffie-Hellman 형의 공개 키이분배 방식을 지금까지 제안된 각종 방식들에 대해 그들의 안전성을 중점적으로 분석하여 새로운 키이분배 방식의 설계시나 안전성 검토시의 보다 체계적인 접근방법을 제시하고자 한다. 대부분의 암호시스템들이 그러하듯이 모든 알려진 공격방법에 대해 견딜 수 있는 것으로 보아 안전한 것처럼 보이나 지금까지 알려진 방법으로는 그 안전성을 증명하는 것이 불가능한 예들이 많았다. 이들에 대해서는 보다 근본적인 프로토콜분석(protocol analysis)을 통해 가능한 어떤 공격에 의해서도 알아내거나 변조할 수 없는 요소가 있음을 보임으로써 간접적인 증명을 시도하였다.

우선 2장에서는 본 논문에서 다루고자하는 일반화된 Diffie-Hellman형의 키이분배 방식에 대해서 살펴보고 3장에서는 키이분배 방식에 대한 각종 공격방법들을 분류하고 각 공격방법에 의해 깨어질(break or crack) 수 있는 시스템의 예를 들어 설명한다. 4장에서는 안전성 증명의 도구로서 reducibility test 및 엔트로피(entropy) 함수 그리고 프로토콜 분석방법등을 소개하고 또한 키이분배 방식에서의 파라독스(paradox)를 살펴보기로 하며 마지막으로 5장에서 결론을 맺고자 한다.

2. Diffie-Hellman 형의 공개 키이분배 방식

이 장(chapter)에서는 1976년 Diffie와 Hellman 이 제안했던 모듈라멱승(modular exponentiation)에 바탕을 둔 공개 키이분배 방식과 그 문제점을 살펴보고 이를 극복하기 위한 보다 일반적인 DH군의 키이분배 방식(DH family of KDS)을 정의한다.

2.1 Diffie-Hellman의 키이분배 방식

DH방식은 암호학에서 잘 알려진 유한체(finite field) GF(p) 상에서 이산대수(discrete logarithm : DL)문제를 풀기 어렵다는 사실을 바탕으로 한다[O84][Mc90](여기서 p는 큰 소수(prime)이다). 즉 GF(p)의 원시원소(primitive element or generator) g에 대해 $Y \equiv_p g^X$($\equiv_p$는 congruence mod p로서 p로 나누었을 때의 나머지를 계산하는 연산기호임)가 주어졌을 때 이로부터 X를 계산하는 것이 매우 어렵다는 것으로 만일 p가 Pohlig-Hellman 방식[PH78]의 logarithm 계산이 효과적이 될 수 없도록 p-1 역시 큰 소수를 포함하도록 선택한다면 지금까지 알려진 최선의 알고리즘을 이용하더라도 필요한 기본 연산수(bit operation)는 대략 $e^{\sqrt{\ln p \ln \ln p}}$ 정도가 된다[A79].

이러한 성질을 이용하여 두 사용자 A와 B가 비밀 공유키이를 생성하는 DH방식을 나타내면 다음과 같다. 프로토콜은 완전 대칭이므로 한 사용자 A에 대해서만 기술하기로 한다.

- 사용자 A는 임의의 랜덤수(random number) X_A를 선택하여 비밀로 간직하고 $Y_A \equiv_p g^{X_A}$를 계산하여 그 결과를 B에게 전송한다.
- 사용자 A는 B로부터 받은 Y_B를 이용하여 $K \equiv_p (Y_B)^{X_A} \equiv_p g^{X_A X_B}$를 계산하여 비밀키이로 사용한다.

만일 각 사용자 i의 공개키이 $Y_i \equiv_p g^{X_i}$를 공개키이 디렉토리에 등록하여 모든 사용자들이 이용가능하게 한다면 두 사용자 간의 통신 대신에 공개키이 디렉토리를 액세스(access)함으로써 공유키이를 계산할 수 있게 된다. 이 DH방식의 안전성은 전적으로 주어진 공개정보(g, p, g^{X_A} g^{X_B})로 부터 $g^{X_A X_B}$를 구하는 것이 계산상 불가능하다는(computationally infeasible) 사실에 근거하는 것으로 통상 Diffie Hellman 문제로 불리며 Elgamal 암호시스템[E85][LL90] 등의 바탕이 되고있다. 이는 GF(p) 상에서의 DL 문제를 푸는 것과 동등한 것으로 추측되고 있으나 아직까지 증명되지않는 공개문제(open problem)로남아 있다. 한편 Shmuely[Shm85]와 McCurley[Mc88] 등 여러 학자들에 의해 독자적으로 이 DH방식에서 소수 p 대신에 두 큰 소수 p와 q의 곱인 합성수 m(통상 RSA modulus로 불리움)을 modulus로 사용함으로써 안전성을 높일 수 있다는 사실을 밝혀 내었다. 즉 합성수 m을 사용함으로써 DL 문제와 마찬가지로 풀기 어려운 것으로 알려져 있는 큰 합성수의 소인수분해(integer factorization : IF) 문제를 원래의 DH방식에 결합시켰을 뿐 아니라 이 합성수를 modulus로 이용하는 DH방식(composite Diffie Hellman : CDH)을 깨는 문제가 큰 합성수 m을 소인수분해하는 것 만큼 어렵다는 사실을 증명하였다. 따라서 두 문제 DL과 IF가 모두 계산상 불가능한 정도로 충분히 큰 modulus를 선택한다면 이 두 문제 중 적어도 하나가 어렵게 남아있는 한 CDH방식은 안전하게 남아있을 것이며 실제 IF 문제의 어려움만을 가정한다 하더라도 이 CDH의 안전성을 바탕으로 하는 KDS의 안전성을 증명함으로써 그 시스템이 IF문제 만큼 어렵다는 사실을 보일 수 있을 것이다. 본 논문에서 다루는 모든 키이분배방식 역시 합성수 m을 modulus로 사용하는 CDH의 일반형이 될 것이다.

위와같은 합성수 m을 modupus로 사용하는 정수 링(integer ring) Z/mZ 상에서의 unit group $(Z/mZ)^*$ 구조 뿐만아니라 imaginary quadratic field $Q(\sqrt{D})$(D : square free negative integer)에서 class group구조를 이용한 DH방식의 변형에 대해서도 이를 깨는 것이 D를 소인수

분해하는 것만큼 어렵다는 사실이 증명된다[BW88]. 또한 이 DH방식을 GF(p)상의 matrix ring으로 확장한 방식[OVS84]이나 유한체 상의 elliptic curve의 점들의 abelian group 구조를 이용하여 이 곡선상에서 DL 문제를 푸는 것이 어렵다는 사실에 바탕을 둔 DH방식의 변형[Mi85][Ko87] 등이 있다. 특히 후자의 경우 elliptic curve 상에서의 DL문제가 유한체 상에서보다 훨씬 안전도가 높은 것으로 알려져 암호이론을 연구하는 학자들에게 주요 연구대상이 되고 있다.

이상에서 살펴 본 DH나 CDH방식 모두의 결정적인 결함은 그들이 항상 동일한 세션키이를 가져다 준다는 사실이다. 통상 KDS가 DES와 같은 보통정도의 안전도(of moderate strength)를 갖는 암호시스템에서 키이관리를 위해 사용된다는 사실을 고려한다면 이는 어떤 경로로든 단 한번의 세션키이의 유출로 인해 그 이후의 해당 사용자들 간의 모든 비밀 메세지는 당사자들이 모르는 사이에 완전히 노출된다는 사실을 의미하며 또한 이 사실을 발견했다 하더라도 KDS의 비밀키이 및 공유키이를 변경해야 하는 불편함을 초래하게 될 것이다. 따라서 대부분의 KDS에서는 매번 다른 세션키이를 얻을 수 있도록 두 사용자 간에 통신을 통하여 랜덤 정보(information depending on random numbers)를 주고 받는 것이 상례이다.

2.2 일반화된 DH군의 키이분배 방식

원래의 DH방식에서 항상 동일한 세션키이를 초래하는 결점을 보완하기 위해서 일반적인 DH형의 키이분배 방식에서는 두 사용자가 각각 랜덤하게 발생시킨 수를 어떤 형태로든 서로 교환하여 이 랜덤수에 의존하는 세션키이를 계산하게 된다. 우선 합성수 modulus m(=pq)과 unit group$(Z/mZ)^*$(1에서 m-1까지의 수 중에 m과 서로소(relatively prime)인 모든 수의 집합)의 충분히 많은 원소들을 발생시키는 기본원소(base element) g는 모든 사용자들에게 알려져 있다고 가정한다. 또한 원래의 DH방식에서와 마찬가지로 시스템을 사용하는 각 사용자 i는 자신의 비밀키이 S_i를 선택하여 비밀리에 간직하고 $P_i \equiv_m g^{S_i}$를 계산하여 공개키이 디렉토리에 등록한다. 이와같이 등록을 마치면 시스템에 가입한 어떤 두 사용자(사용자 i와 j)도 다음의 과정을 통하여 비밀 세션키이를 공유할 수 있다.

- 사용자 i는 랜덤수 R_i를 발생시켜 자신의 비밀키이 S_i와 공개정보 $P=\{m, g, P_i, P_j\}$를 이용하여 전송정보 $Z_i=F_i(P, S_i, R_i)$를 계산하여 사용자 j에게 전송한다.
- 사용자 i는 상대방으로부터 받은 전송정보 Z_j를 이용하여 세션키이 $K_i=H_i(P, S_i, R_i, Z_j)$를 계산한다($K=K_i=K_j$).

여기서 $F_i(\cdot)$는 사용자 i가 상대방에게 전송할 전송정보를 계산하는 함수이며 $H_i(\cdot)$는 상대방으로부터 받은 전송정보를 이용하여 세션 키이를 계산하는 함수이다. 이 두 함수의 성질 및 그 복잡도에 따라 전체 시스템의 성능이 좌우되므로 이들을 적절히 선택하는 것이 매우 중요하다. 위의 키이분배 방식을 입/출력 관계로 표시하면 다음과 같다.

DH-KDS :

입력 : $P=\{m, g, P_i, P_j\}$, $S=\{S_i, S_j\}$, $Z=\{Z_i, Z_j\}$

출력 : $K_i=H_i(P, S_i, R_i, Z_j)=K_j=H_j(P, S_j, R_j, Z_i)$

여기서 P는 공개정보, S는 비밀정보, Z는 전송정보를 각각 나타낸다. 이상에서 보듯이 이러한 DH방식의 변형은 매번 발생되는 랜덤수에 따라 다른 세션키이를 발생시키게 된다. 지금까지 제안된 대부분의 시스템들이 대칭형 프로토콜이고 이는 컴퓨터망을 통한 통신이 대부분 대화형 통신(interactive communication)인 점을 고려할 때 가장 일반적인 경우이므로 본 논문에서도 $F_i(\cdot)=F(\cdot)$이고 $H_i(\cdot)=H(\cdot)$인 경우 즉 쌍방의 계산이 같은 경우를 주로하여 분석하기로 한다. 따라서 각종 KDS의 예에서도 따로 언급이 없으면 대화형 통신을 다룬 것이다. 그러나 secure E-mail과 같은 응용에서는 일방향 키이분배 프로토콜(KDS using one-way communication) 역시 중요하므로 이 역시 따로 언급을

하며 취급을 할 것이다. 그리고 예로 든 KDS가 다른 논문에서 modulus p를 사용하였다 하더라도 지수에 inverse가 사용되지 않는 한 모두 modulus m을 사용하여 설명하였다.

3. 공격방법의 분류 및 성공 예

키이분배 방식에 대한 공격방법들은 공격자(adversary)들이 입수 가능한 정보 및 그들의 공격형태에 따라 다음과 같이 분류할 수 있다. 공격자들이 보유하고 있는 정보량에 따라 전송정보 및 공개정보만이 이용가능한 ciphertext-only attack과, 이들 정보뿐만 아니라 과거의 세션키이 및 그에 해당하는 전송정보들을 가진 공격자들에 의한 known-key attack으로 나눌 수 있고, 공격자들의 공격양상에 따라서는 wire-tapping등의 수동적인 방법으로 단순히 통신로상에 전달되는 정보의 관측에 의한 passive attack과, 보다 악의적인 방법으로 타 사용자를 사칭하여 통신을 시도 혹은 통신에 응답하거나 두 합법적인 사용자들의 중간에서 서로에게 합법적인 사용자로 가장하여 응답함으로써 그들을 동시에 속이는 active impersonation attack으로 나눌 수 있다. : Ciphertext- Only Passive(COP) attack, Ciphertext-Only Impersonation(COI) attack, Known-Key Passive(KKP) attack, Known-Key Impersonation(KKI) attack.

이 장에서는 이들 각 공격방법에 의해 공격이 성공할 수 있는 시스템을 예로 들어 이들을 설명하기로 한다. 공격이 성공했다 함은 공격자가 이들 각종 공격에 의해 합법적인 사용자와 세션키이를 공유하게 됨으로써 해당 세션동안의 모든 메세지를 불법적으로 읽을 수 있다는 것을 의미한다.

Known-key attack은 일반 암호시스템에서 known-plaintext attack과 유사하며 Yacobi[Y90]에 의해서 최초로 언급된 공격방법으로 통상 KDS가 DES와 같이 보통 정도의 안전도를 갖는 암호시스템에서 키이분배를 위해 사용된다는 점에서 그리고 어떤 경로로든(암호문분석, 사용자의 부주의등) 세션키이의 노출 가능성을 배제할 수 없다는 점에서 충분히 고려되어야 할 공격방법이라 할 수 있다. 과거의 세션키이가 현재의 다른 세션키이를 공격하는데 새로운 정보를 줄 수 있다는 사실은 세션키이의 발생이 충분히 랜덤화되지(randomized) 않았다는 것을 의미한다. 즉 매 세션키이의 발생시마다 세션키이의 불법 계산에 필수적인 정보가 숨어져서 보존됨으로써 일단 하나의 세션키이가 노출되면 새로운 세션키이의 불법계산은 훨씬 쉬워지게 되는 것이다. 제 4장에서 상세히 다루겠지만 이 known-key attack은 키이분배 방식에서의 파라독스(paradox)를 낳게 된다. 즉 ciphertext-only attack하에서 시스템이 안전하다는 증명은 곧 known key attack하에서 그 시스템이 깨어질 수 있다는 것을 의미하게 되는 경우가 생긴다.

한편 일방향 키이분배 방식의 경우 두 사용자가 랜덤 정보를 주고 받는 대화형 프로토콜과는 달리 세션키이가 한 사용자에 의해 발생된 랜덤수에만 의존하므로 replay attack에 의해 과거의 전송정보를 재전송함으로써 항상 같은 세션키이를 생성시킬 수 있다는 문제점이 있다. 물론 공격자가 해당 세션키이를 모른다면 후속되는 확인과정(handshake)이나 암호문 전송후의 응답여부에 의해 이를 검출할 수 있겠지만 과거의 세션키이를 알고있는 공격자의 경우 언제나 impersonation attack이 가능할 것이다. 따라서 일방향 키이분배 방식에서는 이와같은 문제점을 해결하기 위해 시간/날자 등을 이용한 time-stamped protocol[DS81]을 사용하는 것이 일반적이다.

3.1 Ciphertext-Only Attack

Ciphertext-Only Passive(COP) attack은 가장 초보적인 공격형태로 누구나 쉽게 얻을 수 있는 전송 정보 Z와 공개정보 P만을 이용하여 해당 사용자(들)의 세션키이를 계산하고자 하는 공격방법으로 다음의 입/출력 관계가 이를 나타낸다.

COP : 입력 : P, $Z=\{Z_1, Z_j\}$, 출력 : $K=G(P, Z)$

여기서 G(·)는 공격자가 세션키이 계산을 위해 이용할 수 있는 모든 함구를 지칭한다. 이는 암호해석(cryptanalysis) 방법 중 가장 기본적인 ciphertext-only attack에 해당한다. 이러한 공격을 막는 시스템을 만들기는 아주 쉽고 기존의 발표된 어느 시스템도 이 공격에는 다 안전하다고 볼 수 있다.

Ciphertext-Only Impersonation(COI) attack은 공격자가 가진 정보는 COP attack과 마찬가지로 제한되어 있으나 COP attack과는 달리 타 사용자를 가장하여 보다 적극적으로 프로토콜에 관여함으로써 합법적인 사용자와 같은 세션키이를 계산하고자 시도하는 모든 가능한 공격방법을 지칭하며 다음의 입/출력 관계가 이를 나타낸다.

COI : 입력 : P, $\underline{Z}=\{Z_1, \underline{Z_j}\}$

출력 : $\underline{K}=G(P, \underline{Z})$

여기서 $\underline{Z_j}$는 공격자가 사용자 j를 가장하여 사용자 i에게 전송하는 전송정보를 나타낸다. COI attack에 의해서 쉽게 깨어지는 시스템의 예로 다음의 키이분배 방식들을 살펴보자.

[예 3.1.1][설명용 설계 예]

- 사용자 i는 랜덤수 R_i를 발생시켜 $Z_i=_m g^{R_i}$를 계산하여 j에게 전송한다.
- 사용자 i는 j로부터 받는 Z_j를 이용하여 세션키이 $K_i\equiv_m(Z_j \cdot P_j^{-1})^{(R_i-S_i)}\equiv_m g^{(R_i-S_i)(R_j-S_j)}$를 계산한다.

이 시스템의 경우 공격자 a는 사용자 j를 가장하여 자신이 선택한 랜덤수 R_a로 $\underline{Z_j}\equiv_m g^{R_a}P_j$를 계산하여 사용자 i에게 전송하면 i가 계산하는 세션키이 $K_i\equiv_m g^{R_a(R_i-S_i)}$을 공격자 a역시 $\underline{K}\equiv_m (Z_i \cdot P_i^{-1})^{R_a}$와 같이 계산할 수 있게 되므로 COI attack에 의해 쉽게 깨어진다. ///

다음의 예는 일방향 프로토콜로 COI attack이 가능한 경우이다(사용자 i가 통신을 시작하는 것으로 가정한다).

[예 3.1.2][MTI86]

- 사용자 i는 랜덤수 R_i를 발생시켜 $Z_i\equiv_m g^{R_i}$를 계산하여 j에게 전송한다.
- 사용자 i와 j는 각각 세션키이 $K_i\equiv_m P_j^{S_i \cdot R_i}=K_j\equiv_m(Z_i \cdot P_i)^{S_j}\equiv_m g^{S_iS_j+S_jR_i}$를 계산한다.

이 시스템 역시 공격자 a가 사용자 i를 가장하여 전송정보 $\underline{Z_i}\equiv_m g^{R_a} \cdot P_i^{-1}$를 전송하면 사용자 j는 세션키이로 $g^{R_aS_j}$을 계산할 것이므로 공격자 역시 $P_j^{R_a}$로 이를 계산할 수 있다. ///

위의 두 예에서 볼 수 있듯이 COI attack에 의해 깨어지는 시스템은 공격자가 impersonation에 의해 세션키이 계산에서 일부항을 상쇄시키거나 대체시킴으로써 합법적인 사용자가 계산하는 세션키이를 공격자 역시 자신이 전송한 전송정보 및 공개정보로부터 계산할 수 있는 경우이다. 이 COI attack은 4.5절에서 소개되는 비교적 안전한 대부분의 시스템들에서 보듯이 공격자가 유효한 전송정보를 발생시킬 수 없도록 전송함수의 계산에 두 사용자의 비밀키이를 포함시키거나, 세션키이 계산함수에 사용자가 발생시킨 랜덤수에 의존하면서 어떤 형태의 전송에 의해서도 영향을 받지 않는 항을 포함시킴으로써 대부분 효과적으로 막을 수 있다.

3.2 Known-Key Passive(KKP) Attack

KKP attack은 과거의 세션키이 및 해당 전송정보(past history)를 가진 공격자가 현재의 전송정보를 관측하여 그 세션키이를 계산하고자 시도하는 공격방법으로 다음의 입/출력 관계로 정의될 수 있다.

KKP 입력 : P, Z, Z′, K′

출력 : K=G(P, Z, Z′, K′)

여기서 홑 따옴표(′)는 과거 세션의 관련 정보를 의미하며 P, Z, K는 2.2절에서 정의한 것과 같다. 이 KKP attack에 의해 쉽게 깨어 질수 있는 시스템으로 다음 예를 살펴보자.

[예 3.2.1][YS89]

- 사용자 i는 자신이 발생시킨 랜덤수 R_i와 자신의 비밀 키이 S_i를 더하여 전송정보 $Z_i=R_i+S_i$를 계산하여 j에게 보낸다.
- 사용자 i는 j로부터 받은 전송정보 Z_j를 이용하여 세션키이 $K_i=_m(g^{Z_j} \cdot P_j^{-1})^{R_i}\equiv_m g^{R_iR_j}$를 계산한다.

여기서 $R_i=Z_i-S_i$라는 사실을 주목하면 세션키이는 $K_i=_m g^{(Z_i-S_i)(Z_j-S_j)}\equiv_m g^{Z_iZ_j}.g^{-Z_iS_j}.g^{-Z_jS_i}.g^{S_iS_j}$이므로

과거의 세션키이 K'_1에 해당 전송정보 Z_1' Z_j 로부터 $g^{S_1S_j}$를 계산할 수 있고(첫 세항은 전송정보 및 공개정보로부터 쉽게 계산할 수 있으므로) 따라서 현재의 세션키이도 계산가능하게 된다. 이 시스템에서 세션키이를 자세히 살펴보면 비록 세션키이가 해당 세션에서 발생시킨 랜덤수들에 의해 전적으로 결정되는 것처럼 보이나 사실은 세션키이의 전개 형태에서 볼 수 있듯이 랜덤수에 의존하는 항은 항상 해당 세션의 전송정보 및 공개정보로부터 쉽게 계산될 수 있는 것들이고 유일하게 공격자에 의해 계산하기 어려운 항인 마지막 항 $g^{S_1S_j}$는 항상 모든 세션 키이에서 고정된 정보로 보존되고 있다는 것을 알 수 있다. 따라서 일단 과거의 세션키이와 해당 전송정보를 알고 있는 공격자의 경우 이 고정된 마지막 항을 알 수 있게 되어 더 이상 이 시스템에서 비밀은 남아 있지 않게 되므로 시스템이 깨어지는 것은 당연하다. ///

다음의 예는 일방향 프로토콜로 KKP attack이 가능한 경우이다(사용자 i가 통신을 시작하는 것으로 가정한다).

[예 3.2.2][KW90]

- 사용자i는 랜덤수 R_1을 발생시켜 전송정보 $Z_1 \equiv_m g^{R_1}$ $P_j^{S_j} \equiv_m g^{R_1 \cdot S_1 S_j}$을 계산하여 j에게 보낸다.
- 사용자 i의 세션키이는 $K_1 \equiv_m g^{K_1}$이고 사용자 j는 받은 전송정보 Z_1을 이용하여 세션키이 $K_j \equiv Z_1 \cdot P_1^{S_j} \equiv_m g^{K_1 S_j} \cdot g^{S_1 S_j} \equiv_m g^{K_1}$을 계산한다.

과거의 전송정보 $Z_1' \equiv_m g^{K_1'} \cdot g^{S_1 S_j}$와 세션키이 $K' \equiv_m g^{K_1'}$을 얻은 공격자는 $g^{S_1 S_j}$을 쉽게 구할 수 있고 이를 이용하여 새 전송정보 $Z_1 \equiv_m g^{R_1} \cdot g^{S_1 S_j}$에서 새 세션키이 $K \equiv_m g^{K_1}$을 쉽게 계산할 수 있다. ///

3.3 Known-Key Impersonation(KKI) Attack

KKI attack은 공격자가 일단 과거의 세션에서 COI attack을 시도하여 그 공격이 성공하지는 못하였지만 그 후 어떤 경로로든 해당 세션키이를 얻어 이를 보유하고 있다는 가정하에서 출발하여 현재의 세션에서 impersonation attack을 시도하는 공격방법으로 다음의 입/출력 관계가 이를 나타낸다.

$$\text{KKI} : \text{입력} : P,\ \underline{Z}=\{Z_1, Z_j\},\ \underline{Z}'=\{Z_1', Z_j'\},\ K'$$
$$\text{출력} : \underline{K}=G(P, \underline{Z}, \underline{Z}', K')$$

여기서 Z_j와 Z_j'는 각각 현재 및 과거의 세션에서 공격자가 사용자 j를 가장하여 사용자 i에게 전송하는 전송정보를 나타내며 K'은 공격자에 의해 전송된 Z_j'을 이용하여 사용자 i에 의해 계산된 과거의 세션키이이다.

이 KKI attack은 공격자가 시도할 수 있는 가장 강도높은 공격방법이나 대신 공격자가 그에 필요한 정보를 획득하기가 그만큼 어렵다. 즉 공격자가 과거 세션에서 COI attack을 시도하여 실패하면 공격자에 의해 선택된 전송정보 Z_j'을 이용하여 사용자 i에 의해 계산된 K'은 사용자 i나 공격자에 의해 계산된 세션키이와 다를 것이므로 이 키이가 암호문에 사용될 가능성은 거의 없다.(단 한번의 암호문 전송만으로도 상대방이 해당 세션키이를 갖고 있지 않다는 사실이 탄로난 것이므로 공격자가 얻을 수 있는 암호문은 적어도 한번의 전송량뿐이다). 따라서 해당 세션키이 K'을 얻는 것은 COP attack보다 훨씬 어렵다고 할 수 있다.

보다 제한된 KKI attack으로는 과거 세션의 전송정보와 해당 세션키이를 가지고 현재의 세션에서 impersonation attack을 시도하는 경우를 들 수 있다. COI 및 KKP attack에는 잘 견디나 이 KKI attack에 깨어지는 시스템은 공격자가 현재 세션에 대한 COI attack에서 사용자 i로 하여금 과거의 세션키이 K'와 동일한 세션키이를 계산하도록 전송정보를 조작할 수 있는 경우로 제한되어 있다. 위에서 설명한 KKI attack의 두 경우에 대해 공격이 성공할 수 있는 시스템의 예를 들어 본다.

[예 3.3.1][Y90]의 Example2

- 사용자 i는 자신이 발생시킨 랜덤수 R_1로 전송정보 $Z_1 \equiv_m g^{R_1}$를 계산하여 j에게 보낸다.

- 사용자 i는 j로부터 받은 전송정보 Z_j를 이용하여 세션키이 $K_i \equiv_m Z_j^{R_i} \cdot P_j^S \equiv_m g^{R_iR_j+SS_j}$를 계산한다.

이 시스템은 다음과 같이 KKI attack에 의해 깨어진다. 우선 공격자는 과거 세션에서 사용자 j를 가장하여 자신이 선택한 랜덤수 R_a'을 이용하여 전송정보 $Z_j' \equiv_m g^{R_a'}$을 전송하면 사용자 i는 세션키이 $K_i' \equiv_m g^{R_iR_a'+SS_j}$을 계산한다. KKI attack은 이 K_i'를 알고 있다고 가정하므로 공격자는 이로부터 $g^{SS_j} \equiv_m K_i' \cdot (Z_i')^{-R_a'}$을 쉽게 계산할 수 있다. 따라서 현재 세션에서 COI attack을 감행하면, 즉 임의의 랜덤수 R_a를 선택하여 $Z_j \equiv_m g^{R_a}$를 전송하면 $K_i \equiv_m g^{R_iR_a+SS_j} \equiv_m Z_i^{R_a} \cdot g^{SS_j}$이므로 사용자 i가 계산하는 세션키이를 공격자 역시 쉽게 계산할 수 있다. ///

[예 3.3.2] [설명용 설계 예]

- 사용자 i는 랜덤수 R_i를 발생시켜 이 R_i를 사용자 j에게 전송한다.
- 사용자 i는 j로부터 받는 전송정보 R_j를 이용하여 세션키이 $K_i \equiv_m (P_j^{SS})^{R_i \oplus R_i} \equiv_m g^{SS_j(R_i \oplus R_i)}$를 계산한다.

여기에서 $\oplus$는 bit-by-bit exclusive-or이다.

이 시스템은 과거의 전송정보 및 해당 세션키이를 알고 있는 공격자에 의해 쉽게 깨어질 수 있다. 즉 공격자는 현재 세션의 COI attack에서 단지 R_j대신 $R_a = (R_i' \oplus R_j') \oplus R_i$를 사용자 i에게 전송하면 그가 알고 있는 과거의 세션키이 $K' \equiv_m g^{SS_j(R_i' \oplus R_i')}$과 동일한 세션키이를 i로 하여금 계산하게 할 수 있다. ///

이상에서 살펴본 known-key attack의 특성을 다음과 같이 정리할 수 있다. Known key attack은 알려진 세션키이가 공격자에게 새로운 정보를 제공할 때만 의미가 있다. 즉 세션키이가 매 세션마다 보존되는 고정된 항의 비밀정보(fixed secret term preserved in each session)를 포함하고 있고 이 비밀정보를 전송 정보의 조작에 의해 계산 가능한 시스템이나, 혹은 전송정보를 조작하여 알려진 세션키이의 함수로서 공격자가 계산 가능한 세션키이를 합법적인 사용자로 하여금 계산할 수 있는 시스템 등은 이 known-key attack에 의해 항상 깨어진다. 전자의 경우 이 비밀정보를 단순히 해당 세션의 전송정보로부터 계산할 수 있으면 KKP attack이 가능하고 그렇지 않은 경우는 과거 세션에서 COI attack에 의해 공격자의 랜덤수를 세션키이에 미리 심어 둠으로써 그후 해당 세션키이를 입수하였을 때 이 고정된 비밀정보를 쉽게 계산할 수 있다(KKI attack). 따라서 이 known-key attack하에서 안전한 시스템이 되기 위해서는 매 세션키이의 계산에 있어서 어떤 형태로든 고정된 항의 비밀정보가 보존되어서는 안된다는 사실을 알 수 있다. 일방향 프로토콜의 경우는 앞에서도 언급했듯이 time stamp를 사용하지 않으면 replay attack에 의해 언제나 그가 알고있는 세션키이를 생성시킬 수 있다.

4. 안전성 증명의 방법들

이 장에서는 DH 방식의 키이분배 프로토콜의 안전성을 증명하는 방법들을 알아보고 이들을 바탕으로 안전한 시스템을 설계하는데 도움이되는 접근방법(design approach)들을 제시하고자 한다.

3장에서도 언급했듯이 키이분배 방식의 안전성은 전송정보의 계산함수 $F(\cdot)$와 세션키이 계산함수 $H(\cdot)$ 및 그들의 상관 관계에 의해 전적으로 결정된다. 일반적으로 이들 함수들의 다양한 형태 및 다양한 공격 양상들로 인해 모든 시스템에 대해서 일률적으로 적용할 수 있는 증명방법은 찾기 힘들다. 따라서 여기서는 Yacobi와 Shmuely [YS89]에 의해 처음으로 KDS에 적용되어 상당히 효과적인 것으로 인정된 Reducibility에 의한 방법을 중심으로 하여 전송 도중의 정보누출(information leakage)정도를 알아보는 정보이론(information theory)에서의 엔트로피 함수(entropy function)와, 주어진 시스템에 대하여 모든 가능한 공격 방법들을 고려하여 안전성을 검토하는 프로토콜의 분석에 의한 방법(protocol analysis)등을 소개한다. 마지막으로 이들 증명 과정을 통하여 다양한 공격양상들에

대해 그들을 무력화시킬 수 있도록 함수 F(·)와 H(·)가 갖추어야할 조건들을 살펴보기로 한다.

4.1 Reducibility에 대하여

70년대 초에 Cook[C71]과 Karp[Ka75]에 의해 소개된 polynomial time reducibility의 개념(Cook reducibility 혹은 Turning reducibility와 Karp reducibility 혹은 many-one reducibility)은 그 후 많은 계산상 어려운 문제들(computationally intractable problem)을 복잡도(complexity)에 따라 분류하고 식별해 내는 데 지대한 공헌을 해왔다.[GJ79] 이들은 다음과 같이 정의된다.

◆만일 문제 B를 풀 수 있는 (가상적인) 알고리즘을 oracle로 가지고 문제 A를 polynomial time으로 풀 수 있는 Oracle Turing Machine (OTM) M이 존재한다면 문제 A는 문제 B로 *Turing reducible*하다($A \propto_T B$)고 한다. 즉 문제 B를 푸는 알고리즘을 서브루틴으로 가지고 이를 자유로이 호출하여 문제 A를 polynomial time으로 풀 수 있다면 $A \propto_T B$이다.

◆만일 문제 A에 속하는 instance들 만을 문제 B의 instance로 변환(transformation)하는 polynomial time으로 계산 가능한 함수 f(·)가 존재한다면 문제 A는 문제 B로 *many-one reducible* 하다($A \propto_m B$)라고 한다. 즉 문제 A에 속하는 모든 x에 대하여 $x \in A$ iff $f(x) \in B$인 polynomial time으로 계산 가능한 함수 f(·)가 존재 한다면 $A \propto_m B$ 이다.

위의 정의에서 볼 수 있듯이 Turing reduction이 many-one reduction보다 일반적인 reduction이라고 할 수 있다. 어느 경우나 문제 A가 문제 B로 reducible하다($A \propto B$)는 것은 만일 문제 B를 polynomial time으로 풀 수 있으면 문제 A 역시 polynomial time으로 풀 수 있고 문제 A가 intractable하다(polynomial time으로 풀 수 있는 알고리즘이 존재하지 않는다)면 문제 B 역시 마찬가지라는 것을 의미한다. 따라서 문제 B는 최소한 문제 A 만큼은 어렵다고 할 수 있다. 이러한 사실을 이용하면 알고리즘이 알려지지 않은 많은 계산상 어려운 문제들을 비슷한 정도의 어려움을 갖는 complexity class들로 분류하는 것이 가능하다[W87]. 예를 들어 만일 어떤 문제 B를 쉽게(이후부터 어떤 문제가 '쉽다' 혹은 '어렵다'는 말은 polynomial time으로 풀 수 있는지의 여부를 뜻하는 의미로 사용 하겠음)풀 수 있는 알고리즘을 아직까지 찾을 수 없었다면 과연 이 문제가 어느 정도 어려운 문제인지, 즉 쉽게 풀 수 있는 데도 그 방법을 찾지 못하는 것인지 아니면 정말로 풀기 어려운 문제에 속하는지를 판별할 수 있는 가장 쉬운 방법은 이미 풀기 어려운 것으로 알려져 있는 문제들의 그룹(예 : NP class)중의 어떤 다른 문제 A가 이 문제 B로 reducible 한지를 조사해 보는 것이다. 따라서 만일 어떤 문제 A가 풀기 어렵다는 사실을 이용하여 주어진 문제 B 역시 풀기 어렵다는 것을 증명하고자 하거나 어떤 문제 B를 쉽게 풀 수 있다는 사실을 이용하여 주어진 문제 A 역시 쉽다는 것을 증명하고자 한다면 $A \propto B$임을 보이는 것으로 충분할 것이다. 물론 모든 문제에 대해 항상 reduction이 가능한 것은 아니므로 여기에도 한계는 있지만 이 reducibility test는 계산이론(computability theory)이나 알고리즘 연구 등에서 지금까지 알려진 가장 강력한 도구로서 널리 사용되어 왔다.

한편 위에서 살펴본 polynomial time reduction은 reduction 과정에서 해당문제의 각 instance에 대한 평균적인 어려움의 정도가 보존되지 않는다는 결함이 있다(즉 polynomial time reduction 은 average case complexity가 보존되지 않는다). 이는 곧 reducibility를 이용한 증명에서 그 증명이 특수한 경우에만 한정될 수 있다는 것을 의미한다. 예를들어 만일 문제 A에서 발생할 빈도가 높은 예(instance)가 문제 B에서는 좀처럼 일어 나지 않는 예로 reduction이 된다면(문제 B에서는 드물게 일어나는 특수한 경우를 서브루틴으로 이용하여 문제 A를 푼다고 생각해보자) 비록 $A \propto B$일지라도 이것이 곧 문제 B가 문제 A만큼 어렵다는 증명이 될 수는 없다. 이러한 문제점에 착안하여 Levin[L84]이

최초로 average case complexity를 보존하는 reduction을 이용한 증명을 제안한 이후 이에 대한 많은 연구가 진행되어 왔다[J84][Gu87][Go88][L88].

Average case complexity를 보존하는 reduction 을 다루기 위해서는 우선 주어진 문제에 대하여 그 문제의 가능한 경우들에 대한 확률분포(probability distribution on problem instances)가 주어져야 하며 이와같이 확률분포와 함께 주어지는 문제를 distributional problem이라고 부른다. A∝B가 average case complexity를 보존한다는 말은 그 reduction이 문제 A의 확률분포상에서 생각했을 때 평균적으로 polynomial time으로 계산 가능해야 하고(efficiency), 계산 결과가 문제 A의 유효한 해(solution)가 되어야 하며(validity), 또한 문제 A에서 일어날 가능성이 높은 경우는 역시 문제 B에서도 일어날 가능성이 높은 경우로 대응(mapping)되어야 한다는(domination property)의미이다. 더 자세한 내용은 참고문헌[BCGL89]를 참조하기 바란다.

4.2 Reducibility를 이용한 증명방법

4.1절에서 살펴본 reduction을 이용하면 대부분의 암호시스템이 기반으로 하는 가정들을 바탕으로 키이분배 방식에서의 안전성 여부를 증명할 수 있다. 즉 큰 합성수의 소인수분해 문제나 유한체 상에서의 이산대수 문제 등은 잘 알려진 어려운 문제들이므로 이들을 바탕으로 주어진 키이분배 방식이 최소한 이들 문제만큼은 어렵다는 것을 이들 문제로부터 주어진 키이분배 방식으로의 reduction이 가능함을 보임으로써 증명할 수 있다.

물론 이 reduction은 worst case에 한정 되어서는 의미가 없으므로 위에서 언급한 확률분포를 보존하는 reduction이어야 한다. 대부분의 DH군의 키이분배 방식에서 전송정보 $Z_I=F_I(P, S_I, R_I)$는 균일한 확률분포(uniform distribution)를 갖는 것으로 가정할 수 있으므로 ([YS89]에서와 같이 예외인 경우는 해당 확률분포를 고려해야 하겠지만) 이를 위해서는 주어진 키이분배 방식의 모든 가능한 경우에 대하여 reduction이 가능함을 보이는 것으로 충분할 것이다. 즉 문제 A를 2장에서 언급한 Composite Diffie-Hellman(CDH) 방식을 깨는 문제라고 가정하고(이는 곧 modulus m을 소인수분해하는 문제와 동등하다고 할 수 있다) 증명하고자 하는 새로운 키이분배 방식을 깨는 문제를 문제 B라고 하면(이후부터는 특별한 언급이 없는한 문제 A와 문제 B를 이와 같이 가정하겠음) 문제 B의 모든 가능한 전송정보 F(·)에 대하여 문제 A가 문제 B로 reduction이 가능함을 보이는 것은 문제 B의 모든 경우에 대하여 이를 깨는 것이 문제 A만큼은 어렵다는 증명이 될 것이다(uniform hardness proof).

그런데 주어진 키이분배 방식을 깨는 문제 B는 3장에서 설명했듯이 가능한 공격방식에 따라 공격자가 이용할 수 있는 정보 즉 문제 B의 입력정보가 다르므로 이를 세분하여 각각의 공격방식에 대한 안전성 여부를 증명할 필요가 있다. 따라서 문제 B를 가능한 공격방식에 따라 첨자(subscript)를 써서 다음과 같이 표기하기로 한다. 즉 ciphertext-only passive attack하에서 주어진 키이분배 방식을 깨는 문제를 B_{cop}으로 표기하며 다른 공격방식들에 대해서도 마찬가지로 각각 B_{col}, B_{kkp}, B_{kkl} 등으로 표기하도록 한다.

우선 reduction이 가능한 문제는 ciphertext-only attack에 한정된다는 것을 주목할 필요가 있다. Known-key attack시의 공격자가 가진 추가된 정보 즉 과거의 세션키이는 시스템이 현재의 세션키이를 계산하는 것과는 무관한 정보가 되기 때문이다. 따라서 주어진 시스템의 안전성에 관한 검토는 다음과 같이 일단 passive attack과 impersonation attack으로 분리한 다음, 각각의 경우에 대하여 known-key attack은 ciphertext-only attack의 안전성을 바탕으로 추가된 정보의 유무에 따라 안전성의 여부를 검토하기로 한다.

◆주어진 시스템의 모든 가능한 전송 정보 F(·)에 대하여 $A\propto B_{cop}$이 가능한지 조사한다.

만일 reduction이 가능하다면 이는 곧 COP attack에 대해 주어진 시스템이 안전하다(modulus m을 소인수 분해하는 것이 어렵다는 가정하에서)는 것을 의미한다. 만일 reduction을 시킬 수 없다면 다른 증명방법을 시도하는 수 밖에 없다. 다음으로 과거의 세션키이가 공격자에게 새로운 정보를 제공하는지 여부를 조사한다. 만일 과거의 세션키이가 아무런 새로운 정보를 제공하지 않는다면 이는 곧 B_{kkp} 역시 B_{cop}과 마찬가지로 어렵다는 것을 의미하므로 KKP attack에 대해서도 안전하다는 증명이 된다. 가장 쉬운 방법은 과거의 세션키이를 누구나 자신이 선택한 임의의 랜덤수 및 공개정보를 이용하여 발생시킬 수 있다는 것을 보이는 것이다. 이것은 세션키이가 아무런 새로운 정보를 포함하지 않다는 것을 의미하기 때문이다.

◆모든 가능한 사용자 및 공격자의 전송정보에 대하여 $A \propto B_{col}$가 가능한지 조사한다. 만일 가능하다면 위에서와 마찬가지로 공격자가 과거 세션에서 impersonation attack을 시도한 후 그의 전송정보가 포함된(폐기된) 세션키이를 얻었다고 가정했을때 이것이 다른 세션에서의 impersonation attack에 새로운 정보를 제공하는지 여부를 조사한다.

이상에서의 증명방법은 비록 모든 시스템의 증명에 적용될 수 있는 것은 아니지만(특히 프로토콜의 정상적인 절차를 벗어나서 세션키이를 불법 계산하려고 시도하는 impersonation attack의 경우, 공격자의 모든 가능한 공격양상을 완전히 파악하는 것은 어려우므로 이에 대해 reduction을 시키는 것은 불가능한 것이 대부분이다.) 많은 경우에 안전한 시스템을 찾아내거나 설계하는데 있어서 효과적인 접근방법을 제시해 준다. 그러면 위의 방법으로 다음 시스템들의 안전성 여부를 고찰해 보자.

[예 4.2.1][MTI86]

- 사용자 i는 랜덤수 R_i를 선택하여 $Z_i \equiv_m g^{R_i}$를 계산하여 j에게 전송한다.
- 사용자 i는 j로부터 받은 Z_j를 이용하여 세션키이 $K_i \equiv_m P_j^{R_i} \cdot Z_j^{S_i} \equiv_m g^{(\quad)(\quad)}$를 계산한다.

이 시스템을 CDH방식을 깨는 문제 A를 기준으로 그 안전성을 검토하기 위해 이 두 문제를 다음과 같이 입/출력 관계로 나타내 본다.

A :: 입력 : g, m, g^{X_i}, g^{X_j} 출력 : $g^{X_iX_j}$

B :: 입력 : g, m, g^{S_i}, g^{S_j}, g^{R_i}, g^{R_j}

출력 : $g^{(\quad)(\quad)}$

i) 이 시스템은 모든 가능한 전송정보에 대하여 $A \propto B_{cop}$가 성립한다. 따라서 COP attack하에서 이 시스템을 깨는것은 RSA modulus를 소인수분해하는 것만큼 어렵다.

(증명) 어떤 전송정보에 대해서도 R_i와 S_j를 임의로 선택하고 $g^{S_i} \equiv_m g^{X_i}$, $g^{R_j} \equiv_m g^{X_j}$로 두어 oracle B_{cop}의 입력으로 하면 그 출력 $g^{X_iX_j+(\quad)}$로부터 $g^{X_iX_j}$를 계산할 수 있으므로(임의로 선택한 R_i와 S_j를 이용하여) $A \propto B_{cop}$이 성립한다.

ii) 이 시스템은 KKP attack하에서도 역시 안전하다. 즉 $A \propto B_{kkp}$가 성립한다.

(증명) 주어진 시스템은 세션키이가 알려진다 하더라도 아무런 새로운 정보를 제공하지 않는다. 즉 누구나 자신이 선택한 랜덤수를 가지고 $P_i^{R_j}$, $P_j^{R_i}$와 같이 세션키이를 생성할 수 있으므로 B_{kkp}는 B_{cop}와 마찬가지로 어렵다는 사실이 입증된다.

iii) Impersonation attack하에서의 안전성 여부는 reduction에 의한 방법으로는 증명이 불가능하다. 즉 모든 가능한 impersonation 방법에 대하여 A로부터 reduction을 시킬 수 있는 방법이 없다. 따라서 이와 같은 경우는 후술하는 프로토콜 분석 등을 통해 정상적인 프로토콜의 진행절차를 벗어나서 세션키이를 불법 계산하려고 할때 어떤 impersonation에 의해서도 계산 불가능한 요소가 있음을 보임으로써 간접적인 방법으로 안전성 여부를 검토하도록 한다. ///

[예 4.2.2] ([예 3.1.1]의 변형)

- 사용자 i는 랜덤수 R_i를 선택하여 j에게 전송한다.
- 사용자 i는 j로부터 받은 R_j를 이용하여 세션키이 $K_i \equiv_m (g^{R_j} \cdot P_j^{-1})^{(R_i + \quad)} \equiv_m g^{(R_i + \quad)(\quad)}$를 계산한다.

i) 이 시스템은 COP attack에 대해 안전하다. 즉 $A \propto B_{cop}$이 성립한다.

(증명) 어떤 R_i와 R_j에 대해서도 oracle B_{cop}의 입력으로 $g^{S_i} \equiv_m g^{-X_i}$, $g^{S_j} \equiv_m g^{-X_j}$를 주면 oracle은 그 출력으로 $g^{(R_i+X_i)(R_j+X_j)} \equiv_m g^{R_iR_j} \cdot g^{R_iX_j} \cdot g^{X_iR_j} \cdot g^{X_iX_j}$을 줄것이고 이로부터 $g^{X_iX_j}$를(첫 세항은 전송정보 및 공개정보로부터 쉽게 계산할 수 있으므로) 계산할 수 있다. 따라서 $A \propto B_{cop}$이 성립한다.

ii) 이 시스템은 COI attack하에서도 안전하다. 즉 $A \propto B_{coi}$가 성립한다.

(증명) 임의의 R_i와 공격자에 의해 전송되는 R_a에 대해서도 oracle의 입력을 $g^{S_i} \equiv_m g^{-X_i}$, $g^{S_a} \equiv_m g^{-X_a} \cdot g^{R_a}$로 두면 그 출력 $(g^{R_a} \cdot g^{X_i} \cdot g^{-R_a})^{(R_i+X_i)} \equiv_m g^{X_i(R_i+X_i)} \equiv_m g^{X_iR_i} \cdot g^{X_iX_i}$로부터 $g^{X_iX_i}$을 계산할 수 있으므로 $A \propto B_{coi}$이 성립한다.

iii) 위 시스템은 [예 3.2.1]에서와 유사하게 KKP attack에 의해 쉽게 깨어진다. ///

4.3 키이 분배방식에서의 파라독스(paradox)

Yacobi[Y90]는 디지탈 서명(digital signature)에서의 "Rabin paradox"에 주목하여 known-key attack하에서는 키이분배 방식에서도 마찬가지로 파라독스가 성립한다는 것을 발견했다. 우선 Rabin의 디지탈 서명[R79]에서의 파라독스를 살펴보자. Rabin의 디지탈 서명은 RSA modulus $N=pq$에 대해서 제곱근(square root mod N)를 구하기 어렵다는 사실을 바탕으로 한다. 각 사용자 i의 공개키이(b_i, N_i)와 비밀키이(p_i, q_i)에 대하여 공개키이 암호화 과정은 $C=E(P) \equiv_{N_i} P(P+b_i)$로 정의된다. 따라서 주어진 암호문 C에 대하여 복호화 과정은 $P \cdot (P+b_i) \equiv_{N_i} C$를 푸는 것으로 이는 곧 $(P+b_i/2)^2 \equiv N_iC+b_i^2/4$를 푸는 것과 같으므로 modulus N_i에 대한 제곱근을 구하는 문제로 귀착된다. N_i의 인수 p_i와 q_i를 알고 있는 사용자 i는 각각의 소수에 대하여 제곱근을 쉽게 구할 수 있으므로 이를 해독할 수 있다[Be70][R80]. 그런데 두 소수의 곱 N에 대한 제곱근은 4개가 존재하므로 이들중에서 올바른 메세지를 골라내는 것이 필요한데 이는 Rabin이 구체적으로 제시하지 않은 것으로 최근에 이를 해결한 방법들이 제안되었다[KIT87][HK89].

한편 디지탈 서명의 생성은 복호화 과정과 같이 메세지 M에 대하여 $X \cdot (X+b_i) \equiv N_iM$을 만족하는 X를 구하는 것이므로(만일 이를 만족하는 X가 존재하지 않으면 M에서 몇 비트를 바꾸어 그 근처에서 위의 식을 만족하는 X를 구한다) 서명을 변조하는 것(forging a signature)은 곧 modulus N_i로 제곱근을 구할 수 있다는 것으로 이는 곧 modulus N_i를 소인수분해할 수 있음을 보이는 것이다. 즉 다음과 같이 제곱근을 구하는 알고리즘을 oracle로 이용하면 N의 인수들을 계산할 수 있다. 임의의 수 X를 랜덤하게 선택하여 $X^2 \equiv N_iC$를 계산한 후 이 C를 oracle의 입력으로 주면 oracle은 출력으로 4개의 제곱근(X, Y, N_i-X, N_i-Y)중에서 하나를 줄 것이고 이는 1/2의 확률을 가지고 Y나 N_i-Y가 될 것이다. 이로부터 $\gcd(X+Y, N_i)$나 $\gcd(X+N_i-Y, N_i)$를 계산하면 곧 p_i혹은 q_i를 구할 수 있기 때문이다. 따라서 이 서명방식은 큰 합성수를 소인수분해하는 것 만큼 어렵다는 증명이 된다.

그러나 이 증명방법은 곧 chosen message attack하에서 이 시스템을 쉽게 깰 수 있다는 것을 의미하기도 한다. 즉 공격자는 서명자를 oracle로 이용하여 위의 증명과정을 그대로 수행함으로써 modulus N_i를 소인수분해할 수 있기 때문이다. 이와같은 문제점을 극복하기 위해 Rabin은 서명과정을 랜덤화하여 공격자가 서명을 변조하기 위해서는 임의의 수에 대하여 제곱근을 구하는 것이 요구되도록 하였다. 즉 서명하고자 하는 메세지 M에 서명자가 랜덤하게 선택한 수 R(미리 정해진 길이의 수 : 예 60비트)을 suffix로 덧붙여 공개된 hash function $h(\cdot)$로 압축한 후 $X \cdot (X+b_i) \equiv N_i h(M||R) = C$ ($M||R$은 M과 R의 concatenation)를 만족하는 X를 구하여 (M, R, X)를 메세지 M에 대한 서명으로 삼았다. 이렇게 하면 공격자는 더 이상 서명자를 원하는 메세지의 제곱근을 구해주는 oracle로 이용할 수 없으므로 chosen message attack하에서도 원래의 서명방법에서 생기는 파라독스를 피할 수 있을 것이다. 한편 원래의 서명방법에서와 같은 디지탈 서명에서의 파라독

스는 한때 피할 수 없는 것처럼 오인된적도 있었으나 최근에는 트리(tree)구조의 인증(authentication) 과정을 통하여 메세지의 각 비트에 대한 서명을 랜덤하게 생성함으로써 이를 극복한 서명방식들이 제안되었다[GMRi88][BM 88][NY89].

키이분배 방식에서도 역시 ciphertext-only attack하에서의 안전성 증명이 곧 known-key attack하에서 시스템이 깨어질 수 있다는 것을 의미하는 역설(paradox)이 성립한다. [예 3.2.1], [예 3.3.1] 및 [예 4.2.2] 등에서 보았듯이 세션키이가 매 세션마다 보존되는 고정된 비밀정보($g^{S_1S_2}$)를 포함하고 있고 따라서 일단 알려진 세션키이로부터 이를 계산할 수 있다면 known-key attack하에서 이 시스템이 깨지는 것은 당연한 결과이다. 일반적으로 세션키이가 매 세션마다 보존되는 비밀정보를 갖는 경우는 세션키이의 계산과정에서 서로 교환된 전송정보 중의 랜덤수에 무관한 사용자의 비밀키이들 만으로 구성되는 항(보통 $g^{S_1S_2}$ 의 형태)이 세션키이에 포함되는 경우이다.

이와같은 시스템은 위에서 살펴본 예들에서도 명백히 드러나듯이 이 고정된 비밀정보를 제외한 나머지 부분은 전송정보의 관측(KKP attack)이나 혹은 과거 세션에서의 impersonation attack에 의해 공격자의 랜덤수를 해당 세션키이에 포함시킨 후 이를 얻었을때(KKI attack) 계산이 가능하므로 결국 ciphertext-only attack하에서 이 시스템을 깨는 것은 최소한 이 고정된 비밀정보를 계산하는 것만큼 어렵다는 증명이 된다. 그러나 이제 이 세션키이에 남아 있는 유일한 비밀정보인 사용자의 비밀키이에 의존하는 고정된 항은 일단 세션키이가 알려지면 쉽게 계산할 수 있으므로 이는 곧 이 시스템이 known-key attack하에서 쉽게 깨어진다는 것을 의미한다. 이상에서 살펴본 키이분배 방식에서의 파라독스를 3장의 [예 3.2.1]와 [예 3.3.1]로 reduction에 의한 증명과정을 통해 다시 한번 확인해 보기로 하자.

[예 3.2.1]의 파라독스

우선 기준으로 삼은 문제 A와 [예 3.2.1]을 COP attack하에서 깨는 문제 B_{cop}를 다음과 같이 입/출력 관계로 나타낸 다음 $A \propto B_{cop}$임을 보여 이 시스템이 COP attack하에서 안전함을 보이고 그러나 과거의 세션 키이를 가진 공격자가 이 증명과정을 모방(simulation)함으로써 KKP attack 하에서는 이 시스템을 쉽게 깰 수 있음을 보인다.

A :: 입력 : g, m, g^{X_1}, g^{X_2}

출력 : $g^{X_1X_2}$

B_{cop} :: 입력 : g, m, g^{S_1}, g^{S_2}, S_1+R_1, S_1+R_1

출력 : $g^{X_1X_2}$

i) 이 시스템은 B_{cop} 및 B_{ca}에 대하여 안전하다.

(증명) 이 시스템은 전송함수가 일반적인 모듈라 멱승이 아니라 사용자의 비밀키이와 랜덤수의 합으로 주어지므로 R_1와 S_1가 [0, m)에서 균일한 확률분포를 갖는다고 가정하면 Z_1는 [0, 2m-1)에서 삼각분포(triangular distribution)를 갖게 된다. 따라서 우선 이 구간에서 삼각분포를 갖는 Z_1를 전송했을때 전송도중의 정보누출 여부를 고려해야 한다. 이는 다음 절에서 상세히 다루기로 하고 여기서는 이 전송함수가 안전하다는 가정하에서 안전성 및 파라독스가 성립함을 증명하도록 한다.

[0, 2m-1)구간의 삼각분포상에서 임의의 Z_1, Z_2를 택하고 $g^{S_1}=g^{-X_1}$, $g^{S_2}=g^{-X_2}$로 oracle의 입력을 주면 그 출력 $(g^{Z_1}\cdot g^{X_1})^{R_1} \equiv_m g^{(Z_1+X_1)(Z_2+X_2)} \equiv_m g^{Z_1Z_2}\cdot g^{Z_1X_2}\cdot g^{Z_2X_1}\cdot g^{X_1X_2}$로부터 문제 A의 해 $g^{X_1X_2}$를 계산할 수 있다. 그런데 주어진 S_1에 대하여 사실은 Z_1는 $[S_1, S_1+m)$구간에 놓이게 되므로 [0, 2m-1)에서 삼각분포에 따라 임의로 선택한 Z_1가 항상 유효한 전송함수가 되는 것은 아니다. S_1가 [0, m)에서 균일한 확률분포를 가질때 위의 reduction이 유효한 reduction이 될 평균 확률, 즉 위의 reduction에서 사용한 Z_1가 유효한 전송함수가 될 확률은 $S_1=s$가 주어졌을때 Z_1가 삼각분포상의 구간 [s, s+m)에 있을 확률에 대한 s의 평균이 된다. 즉 $E_s\{Pr(s \leqq Z_1 < s+m|s)\} = \int_0^m \frac{1}{m}\left[\int_s^m \frac{z}{m^2}dz + \int_m^{s+m}\frac{2m-z}{m^2}\right.$

dz]ds=2/3. 따라서 위의 reduction $A \propto B_{cop}$은 유효한 증명이라고 할 수 있다.

B_{coi}에 대해서도 마찬가지로 삼각분포상에서 Z_I를 랜덤하게 선택하면 공격자에 의해 전송되는 임의의 Z_a에 대해서도 $g^{S_I} \equiv_m g^{-X_I}$, $g^{S_J} \equiv_m g^{-X_J} g^{Z_a}$로 oracle의 입력을 주면 그 출력 $(g^{Z_a} \cdot g^{X_J} g^{-Z_a})^{R_I} \equiv_m g^{X_I(X_I+Z_a)} \equiv_m g^{X_I X_J} \cdot g^{X_I Z_a}$로부터 $g^{X_I X_J}$를 계산할 수 있다. 따라서 B_{cop}와 마찬가지로 $A \propto B_{coI}$가 성립한다.

ii) 이 시스템은 KKP attack하에서 paradox가 성립한다.

(증명) 공격자가 가진 과거의 세션키이가 위의 reduction $A \propto B_{cop}$에서 oracle B_{cop}의 역할을 대신할 수 있으므로 A의 해를 구할 수 있고 A의 해를 구할 수 있다면 B_{cop}은 쉽게 풀 수 있다는 것은 당연하므로 이 시스템은 KKP attack하에서 쉽게 깨어진다. ///

[예 3.3.1]의 파라독스

A :: 입력 : g, m, g^{X_I}, g^{X_J}
출력 : $g^{X_I X_J}$

B_{cop} :: 입력 : g, m, g^{S_I}, g^{S_J}, g^{R_I}, g^{R_J}
출력 : $g^{R_I R_J + S_I S_J}$

i) 모든 가능한 전송정보에 대하여 $A \propto B_{cop}$이 성립한다.

(증명) 어떤 전송정보에 대하여도 R_I, R_J를 임의로 선택하고 $g^{S_I} = g^{X_I}$, $g^{S_J} = g^{X_J}$로 oracle의 입력을 주면 그 출력 $g^{R_I R_J + X_I X_J}$로부터 A의 해 $g^{X_I X_J}$를 쉽게 구할 수 있다.

ii) 이 시스템은 KKI attack하에서 파라독스가 성립한다.

(증명) 이 시스템에서는 단순한 과거의 세션키이 및 전송정보만으로는 R_I나 R_J를 알 수 없으므로 위의 reduction에서 oracle B_{cop}의 역할을 대신할 수 없다. 따라서 이 경우는 KKI attack하에서 파라독스가 성립한다. 즉 공격자가 선택한 랜덤수 R_a'를 포함하는 세션키이 $g^{R_I R_a' + X_I X_J}$와 $g^{R_I'}$이 oracle B_{cop}의 역할을 대신할 수 있으므로 이로부터 A의 해를 계산할 수 있고 따라서 현재 세션에서 COI attack을 행하면 쉽게 사용자 i와의 공유키이 $g^{R_I R_a + X_I X_J}$를 계산할 수 있다. ///

다음의 시스템은 일방향 키이분배 프로토콜로 역시 KKI attack하에서 파라독스가 성립하는 예이다. 여기서 문제 A'는 g^{X_I}, g^{X_J}로부터 $g^{X_I X_I'}$를 구하는 문제로 정의하면 이는 역시 CDH를 깨는 문제 A와 마찬가지로 어렵다고 할 수 있다.

[예 4.3.1] [설명용 설계 예]

- 사용자 i는 랜덤수 R_I를 발생시켜 $Z_I \equiv_m M \cdot P_I^{R_I}$를 계산하여 j에게 전송한다. 여기서 $M \equiv_m g^{S_I S_J}$이다.
- 사용자 i와 j는 각각 세션키이 $K_I \equiv_m M^{R_I} = K_J \equiv_m (Z_I \cdot M^{-1})^{S_J} \equiv_m g^{S_I S_J R_I}$를 계산한다.

i) 이 시스템은 $A' \propto B_{cop}$이 성립한다.

(증명) $g^{S_I} = g^{X_I}$, $g^{S_J} = g^{X_J}$로 두고 임의의 R을 선택하여 $Z_I \equiv_m g^R$을 oracle의 입력으로 주면 그 출력 $g^{RX_J + X_I X_I'}$으로부터 A'의 해 $g^{X_I X_I'}$을 쉽게 구할 수 있다.

ii) 이 시스템은 KKI attack하에서 파라독스가 성립한다.

(증명) 위의 [예 3.3.1]의 파라독스에서와 마찬가지로 과거 세션에서 impersonation후의 세션키이 $g^{R'S_J + S_I S_I'}$를 알고 있는 공격자는 이로부터 $g^{S_I S_I'}$을 계산할 수 있고 따라서 현재 세션에서 impersonation attack을 시도하면 쉽게 사용자 j와 세션키이를 공유할 수 있다. ///

이상에서 살펴본 키이분배 방식에서의 파라독스를 정리하면 다음과 같다.

◆만일 $B_{kkp}(B_{kkI}) \propto A$와 $A \propto B_{cop}$가 성립하고 두 reduction이 모든 가능한 전송정보에 대해서 성립한다면 이 시스템은 known-key attack하에서 쉽게 깨어진다. $A \propto B_{cop}$가 모든 가능한 전송정보에 대하여 성립한다는 말은 곧 B_{kkp}나 B_{kkI}가 입력정보로 가지고 있는 특정한 과거의 세션키이 및 관련 전송정보도 oracle B_{cop}의 역할을 대신할 수 있다는 의미이고 두 reduction에서 $B_{kkp}(B_{kkI}) \propto B_{cop}$이 성립하므로 B_{cop} 없어도 KKP나 KKI attack이 성공할 수 있다는 것은 당연하다.

◆만일 그 시스템의 전송함수 $Z_I = F_I(P, S_I, R_I)$가 사용자 i의 랜덤수 R_I에 대하여 one-way 함수이면 단순히 전송정보의 관측만으로 A의 해 $g^{S_I S_J}$를 구할 수 없는 경우가 대부분이므로

이때는 KKI attack하에서 파라독스가 성립한다. 물론 두 reduction이 모두 성립하는 경우에 한해서이다. ([예 3.3.2]과 [예 4.3.1])

◆일반적으로 전송함수 Z_I가 사용자의 비밀키 S_I에 의존하는 경우(주로 $g^{S_I S_J}$의 형태)나 전송간에 서로 독립이 아닌 경우는 reduction이 불가능한 때가 많으므로 파라독스를 피할 수 있는 한 방법이 될 수 있다(4.5절의 예 대부분).

◆일반적으로 세션키이 계산함수 $K_I=H_I(P, S_I, R_I, Z_J)$가 고정된 공개정보 및 비밀키이 만으로 이루어진 항(term)을 포함하고 있지 않다면 두 reduction이 모두 공개정보를 보존하는 것은 불가능하므로 파라독스는 성립하지 않을 것이다 ([예 4.2.1]). 따라서 세션키이가 전적으로 서로 교환되는 랜덤수에 의존하는 one-way 함수만으로 구성되도록 선택하는 것이 안전한 키이분배 방식을 설계하는 한 방법이 될 것이다.

한편 파라독스가 성립하는 시스템은 대부분이 $g^{S_I S_J}$ 형태의 고정된 비밀정보를 세션키이의 일부로 포함하고 있는 경우로 제한되어 있고 또한 이를 피하는 것 역시 쉬운 반면 파라독스를 피할 수 있도록 설계된 시스템은 그 안전성을 증명하기가 어려운 것이 일반적이다. 즉 키이분배 방식에서의 reducibility test는 그 안전성 검증이나 안전한 시스템을 설계하는데 있어서 많은 도움을 주는 것은 사실이나 이것이 일반적으로 모든 시스템에 적용되는 체계적인 증명도구로서의 효용에는 한계가 있다는 것을 알 수 있다.

4.4 엔트로피의 개념 및 그 응용

어떤 한 사건이 지닌 정보는 그 사건이 일어날 확률에 반비례하는 것은 당연하다. 가령 내일 태풍이 불 것이라는 기상예보는 가끔 구름이 끼는 날씨가 될 것이라는 예보보다 훨씬 많은 정보량을 지니는 것은 그것이 일어날 확률이 훨씬 작기 때문이다. 따라서 확률의 역수의 로그(logarithm)를 취한 값이 그 사건의 정보량이 된다.(통상 정보량의 단위로 비트(bit)를 사용하므로 이하에서 로그의 base는 2이다.) 이와같이 어떤 한 랜덤변수(random variable) X가 일어날 확률의 역수의 로그를 취한 값의 평균을 X의 엔트로피(entropy) H(X)라 하며 $H(X)=\sum_x P(x)\log\frac{1}{P(x)}$와 같이 정의된다. 이는 랜덤변수 X에 대한 평균적인 불확실성(average uncertainty)의 정도 혹은 X의 랜덤화된 정도(randomness)를 나타내는 척도라 할 수 있다. 따라서 일어날 수 있는 모든 경우에 대하여 그 가능성이 확률 $p(x)=1/r$로 동일할때 $H(X)=\log r$로 최대가 된다. 한편 어떤 랜덤변수 Y가 주어졌을때의 랜덤변수 X의 남아있는 정보량을 나타내는 것으로 조건부 엔트로피(conditional entropy) H(X|Y)를 생각할 수 있다. 이는 Y가 관측되었을때 X에 남아 있는 불확실성의 정도를 나타내 주는 것으로 $H(X|Y)=\sum_{x,y} P(x,y)\log\frac{1}{P(x|y)}$로 정의되며 관측 Y가 X에 대한 아무런 정보를 제공해 주지 않을때, 즉 두 랜덤 변수 X와 Y가 독립적(independent)일때 최대값 $H(X|Y)=H(X)$를 갖는다. 여기서 Y의 관측에 의한 X의 정보량의 손실 즉 $I(X;Y)=H(X)-H(X|Y)$를 X와 Y 사이의 상호정보량(mutual information)이라 부른다. $I(X;Y)=0$인 경우는 Y가 관측되더라도 X에 대한 정보가 전혀 누출되지 않는, 정보이론상으로 X에 대해 완전 비밀(perfect secrecy)이 보장되는 경우로 키이분배 방식에서 이런 성질을 갖는 전송함수 및 세션키이 계산함수를 선택하는 것이 가장 이상적인 경우라고 할 수 있다.

그러나 이와같은 정보이론적 접근법(information theoretic approach)에 따른 암호시스템의 설계는 거의 불가능하므로(메세지의 길이와 같은 길이의 랜덤 키이스트림(random key-stream)을 이용하는 one-time pad를 제외하고는 perfect secrecy를 제공하는 암호시스템은 아직까지는 존재하지 않는다) 실제로는 공격자의 계산능력이 한정되어 있다는 가정하에 계산상 어려운 문제를 바탕으로 암호시스템을 설계하는 계산이론적 접근법(complexity-theoretic approach)이 주류를 이루고 있다. 예를 들면 $y\equiv_m g^x$과 같은 모듈라 멱승의 경우, y가 주어지면 x를 계산할

수 있으므로 정보이론상으로는 $H(X|Y)=0$이 되나 이 경우는 유한한 계산능력으로는 이산대수 문제를 한정된 시간안에 푸는 것이 어렵다는 계산이론적 가정에 바탕을 둔 것으로 비록 Y가 X의 모든 정보를 포함하고 있지만 이는 계산상 얻기 힘든 정보(computationally inaccessible information)이므로 이같은 함수도 암호시스템에 안전하게(computationally secure) 사용할 수 있는 것이다. 이는 정보이론적 접근법에서 공격자의 계산능력이 무한하다는 지나친 가정을 완화한 것으로 실제로 대부분의 암호시스템을 설계하는 접근법이 되고 있다.

이제 위에서 살펴본 기초적인 정보이론을 바탕으로 몇가지 간단한 연산의 전송함수나 세션키이 계산함수로의 타당성 여부를 전송도중(혹은 노출시)의 정보누출 정도를 통해 알아 보기로 하자. 먼저 [YS89]에서 사용한 비밀키이 S와 랜덤수 R의 합 $Z=S+R$을 전송하는 경우를 생각해 보자. 여기서 S와 R은 모두 $[0, m)$에서 균일한 확률분포를 갖는다고 가정한다. 그러면 Z는 $[0, 2m-1)$에서 삼각분포를 갖게 될 것이다. 우선 $Z=z$가 $[0, m)$에 있을 때는 $R=r$가 취할 수 있는 경우의 수(선택의 자유도)는 $[0, z]$구간의 임의의 값을 가질 수 있으므로 z개 이고(즉 $1/z$의 균일한 확률분포를 가지고) 일단 r이 주어지면 $S=s$는 선택의 여지가 없으므로 r과 s의 결합된 불확실성의 정도는 z가 될 것이다. 한편 z가 $[m, 2m-1)$의 구간에 속할 때는 r과 s는 각각 $[z-m, m)$에서 임의의 값을 취하되 역시 선택의 폭은 $m-(z-m)=2m-z$로 제한되어 있다. 따라서 각 구간에서 이들의 평균을 구하면 이것이 곧 z가 주어졌을때의 r과 s의 불확실성의 평균적인 정도를 나타내는 $H(R, S|Z)$의 값이 될 것이고, 이것의 최대값 즉 $\log_2(2m)=\log_2(m)+1$과의 차이가 전송중에 누출된 정보량이 될 것이다. 그런데 $Z=z$역시 위의 (R, S)의 선택의 자유도를 나타내는 함수와 마찬가지로 삼각분포를 이루므로 $Z=z$가 주어졌을때 (R, S)쌍의 평균 정보량은 대략 $H(R, S|Z)=2\cdot 1/m^2\cdot\int_0^m z\cdot\log_2(z)=\log_2(m)-1/2\cdot\log_2(e)<\log_2(m)-0.7$로 최대값에 비해 2 bit 이하의 차이를 준다. 따라서 전송중의 정보누출은 2 bit 이하라고 할 수 있다.

다음은 $[0, m)$구간의 두 랜덤수 R_1과 R_2의 비트(bit) 단위의 EXclusive OR(modulo-2 addition)에 의한 전송함수의 경우를 살펴보자.($Z=R_1\oplus R_2$를 전송할 경우(R_1, R_2)에 대해 누출되는 정보량) 이 경우는 쉽게 예측할 수 있듯이 $[0, m)$구간의 임의의 Z값에 대해 R_1이나 R_2가 취할 수 있는 선택의 폭에 전혀 제한을 받지 않으므로 정보손실이 전혀 없는 경우이다. 이는 one-time pad(혹은 Vernam cipher)가 perfect security를 제공할 수 있는 근거가 되는 것으로 ciphertext의 확률분포가 plaintext의 확률분포와 통계적으로 완전히 독립임을 의미하기 때문이다. 이 예에서도 마찬가지로 $Z=z$가 주어졌을때 R_1은 $[0, m)$에서 임의의 값을 취할 수 있으므로 (R_1이 주어지면 R_2에 대해서는 선택의 여지가 없다.) $Pr(r_1|Z=z)=Pr(r_1)=1/m$이 되어 $H(R_1|Z)=H(R_1)=\log_2 m$이 된다. 즉 R_1에 대해 정보손실이 전혀 없다. 따라서 이는 두 사용자간에 세션키이를 계산하기 위해 랜덤수 R을 그들간의 고정된 공유키이 $M\equiv_m g^{S_iS_j}$와 EXclusive ORing하여 안전하게 전송할 수 있음을 의미한다. 물론 이때는 R의 노출은 곧 M의 노출을 의미하므로 known-key attack을 막을 수 있도록 세션키이 계산함수는 이 R에 대해 one-way 함수이어야 한다.

마지막 예로 두 랜덤수의 modular multiplication 역시 정보누출이 거의 없이 같은 목적으로 사용될 수 있다.($Z\equiv_m R_1\cdot R_2$를 전송할 경우(R_1, R_2)에 대해 누출되는 정보량) 소수 p를 modulus로 하는 연산에서는 구간 $[1, p-1]$의 모든 수들은 p와 서로소(relatively prime)이므로 R_1이나 R_2가 될 가능성이 있고 따라서 이 경우는 Z가 주어진다 하더라도 EXclusive OR와 마찬가지로 R_1이나 R_2에 대해 정보누출이 전혀 없는 경우이다. 합성수 m을 modulus로 사용할 때에도 R_1이나 R_2가 m의 인수인 p나 q를 포함하는 경우(이 가능성은 무시할 수 있을 정도이다)를 제외

하고는 Z가 주어진다 하더라도 이들에 대한 아무런 정보를 제공해 주지 않는다. 따라서 이 연산 역시 전송함수나 세션키의 계산함수로 안전하게 이용할 수 있을 것이다.

이상에서 살펴본 것처럼 전송함수(혹은 세션키의 계산함수)의 계산에 안전하게 이용할 수 있는 연산은 일반적으로 흔히 쓰이는 계산상 어려움에 바탕을 둔 모듈라 멱승 이외에도 산재하면서도 정보이론상으로 정보누출이 거의 없는 여러가지 연산이 있다는 것을 알 수 있다. 따라서 시간이 많이 걸리는 모듈라 멱승 대신 이들 연산을 이용한다 하더라도 안전한 키이분배 방식을 설계할 수 있을 것이다.

4.5 프로토콜 분석에 의한 안전성 고찰

4.2절과 4.3절에서 살펴본 것처럼 reduction에 의한 증명은 다양한 공격방법과, 키이분배 방식에서의 다양한 전송함수 및 세션키의 계산함수 등에 따라 그 적용이 매우 제한되어 있으므로 결국은 시스템의 안전성을 분석하는데는 각 시스템에 대해 가능한 모든 경우를 고려하여 그 안전성을 검토하는 것이 필요하다. 일반적으로 ciphertext-only attack은 전송함수 및 세션키의 계산함수가 사용자의 비밀키이나 랜덤수에 대해서 one-way 함수의 성질을 갖도록 설계함으로써 막을 수 있는 경우가 많으나 impersonation attack의 경우는 공격자의 모든 가능한 impersonation에 대해서 그 안전성을 고려해야 하므로 일반적으로 안전성의 증명이 힘들뿐더러 그 공격양상을 완전히 파악하기도 어렵다. 하지만 impersonation attack의 경우도 역시 세션키이의 불법계산에 유효한 impersonation이 불가능 하도록 전송함수를 적절히 선택하고 또한 세션키이 계산함수도 공격자의 영향이 전혀 미칠 수 없는 영역을 포함하도록 설계함으로써 대부분의 경우 증명은 어려우나 이를 효과적으로 막을 수 있는 시스템을 설계할 수 있다. 안전한 시스템을 설계하는데 도움이 될 수 있는 전송함수 및 세션키이 계산함수들의 성질을 예를들어 살펴보면 다음과 같다.

◆Known key attack하에서 파라독스를 피할 수 있도록 세션키이의 계산에 매 세션키이마다 보존되는 고정된 비밀정보가 포함되지 않게 한다.

◆Impersonation에 의한 공격 가능성을 줄일 수 있도록 전송함수가 공격자에 의해 계산 불가능한(사용자의 비밀키이에 의존하는) one-way 함수가 되도록 하거나, 세션키이의 계산함수에 사용자가 발생시킨 랜덤수에 의존하면서 전송함수와는 무관한 부분이 포함되도록 선택한다. 즉 공격자가 어떤 방법으로 impersonation을 하더라도 계산 불가능한 항을 포함하도록 하되 이 항이 해당 세션에서 발생시킨 랜덤수에 의존하여 세션키이마다 랜덤하게 변하도록하여 세션키이가 노출되더라도 아무런 정보를 제공하지 않도록 한다. 예를들면 [예 4.2.1]의 세션키이 계산 $K_i \equiv_m P_j^{R_i} \cdot Z_j^{x_i}$에서 $P_j^{R_i}$ 부분은 공격자의 어떤 impersonation Z_j에 의해서도 영향을 받지 않으며(R_i를 모르는 한) 이를 계산하는데 이용할 수 있는 정보는 공개정보 $P_j \equiv_m g^{x_j}$와 전송함수 g의 뿐이므로 이는 CDH를 깨는 문제 A만큼 어려울 것이고 또한 이 부분은 사용자가 발생시킨 랜덤수 R_i에 의존하므로 매 세션마다 랜덤하게 변할 것이므로 알려진 세션키이가 아무런 정보를 주지 않는다. 따라서 이 시스템은 impersonation attack에 대해서도 안전한 것으로 생각할 수 있다.

다음의 두 예는 전송함수로서 두 사용자의 고정된 공유키이(fixed common key) $M \equiv_m g^{x_i x_j}$을 이용하고 또한 세션키이 계산함수가 단일항이면서 랜덤수에 전적으로 의존하는 시스템으로서 수학적인 증명은 어려우나 안전한 시스템인 것으로 생각된다.

[예 4.5.1] [YA83]

- 사용자 i는 공개정보를 이용하여 $M \equiv_m P_j^{x_i} \equiv_m g^{x_i x_j}$을 계산한 후 랜덤수 R_i를 선택하여 전송정보 $Z_i \equiv_m M^{R_i}$을 계산한 다음 이를 j에게 보낸다.
- 사용자 i는 j로부터 받는 전송정보 Z_j를 이용하여 세션키이 $K_i \equiv_m Z_j^{R_i} \equiv_m M^{R_i R_j}$를 계산한다.

우선 이 시스템은 COP와 KKP attack하에서 안전하다는 것은 명백하다. 이는 설령 M을 알고 있다고 하더라도 COP attack하에서 이 시스템을 깨는 것은 CDH를 깨는 문제 A 만큼 어렵고 또한(두 사용자의 랜덤수에 전적으로 의존하는) 세션키이가 알려진다 하더라도 공격자에게 제공하는 새로운 정보는 전혀 없기 때문이다. COI attack의 경우 공격자가 어떤 형태의 $\underline{Z_j}$를 전송한다고 해도 세션키이는 $\underline{Z_j}^{R_i}$의 형태이므로 결국 전송정보 $Z_i \equiv_m M^{R_i}$로 부터 R_i를 알아내야 한다. 이는 M을 모르는 상태이므로 이산대수문제를 푸는 것보다 어렵다고 할 수 있다. 한편 KKI attack의 경우도 KKP attack과 마찬가지로 알려진 세션키이는 새로운 세션의 공격에 아무런 정보를 제공하지 않으므로 COI attack과 다를 것이 없다. 결국 이 시스템에서는 세션키이가 사용자가 선택한 랜덤수에 전적으로 의존하므로 공격자가 합법적인 사용자와 같은 전송정보를 발생시킬 수가 없는 한(즉 사용자가 발생시킨 랜덤수나 고정된 공유키이 M을 모르는한) 어떤 공격에 의해서도 세션키이를 알아내는 것은 불가능하다고 할 수 있다. ///

[예 4.5.2] [ML90]

- 사용자 i는 공개정보를 이용하여 $M \equiv_m P_j^{S_i} \equiv_m g^{S_i S_j}$을 계산하고 랜덤수 R_i를 선택하여 전송정보 $Z_i \equiv_m M \cdot g^{R_i}$을 계산한 후 j에게 보낸다.
- 사용자 i는 modulo m으로 M의 역원(multiplicative inverse) M^{-1}과 j로부터 받은 전송정보 Z_j를 이용하여 세션키이 $K_i \equiv_m (Z_j \cdot M^{-1})^{R_i} \equiv_m g^{R_i R_j}$를 계산한다.

이 시스템 역시 위의 예에서와 유사한 분석을 통해 지금까지 알려진 가능한 어떤 공격에 대해서도 이를 깰 수 있는 방법이 없다는 점에서 안전한 시스템일 것으로 생각된다. ///

위에서 예로든 두 시스템 모두 공격자가 모방(simulation)할 수 없는 전송함수를 선택하고, 또한 세션키이 계산함수로 사용자가 발생시킨 랜덤수와 전송정보에 의존하는 함수를 사용함으로써 어떤 impersonation에 의해서도 유효한 세션키이를 얻는 것을 막을 수 있었다. 즉 시스템의 안전성은 전적으로 공격자가 유효한 전송함수를 발생시킬 수 없다는데 의존한다고 할 수 있다.

다음의 예는 누구나 유효한 전송정보를 발생시킬 수 있으나 세션키이 계산함수에 사용자가 발생시킨 랜덤수의 멱승으로된 단일 항을 포함시켜 어떤 형태의 전송에 대해서도 이를 상쇄시키거나 대체시킬 수 없도록 함으로써 세션키이의 불법 계산을 효과적으로 막도록 하였다.

[예 4.5.3] [ON84]

- 사용자 i는 랜덤수 R_i를 선택하여 전송정보 $Z_i \equiv_m g^{R_i S_i}$을 계산한 후 j에게 전송한다.
- 사용자 i는 j로부터 받은 전송정보 Z_j를 이용하여 세션키이 $K_i \equiv_m (Z_j \cdot P_j^{R_i})^{S_i} \equiv_m g^{S_i S_j (R_i + R_j)}$을 계산한다.

이 시스템은 COP attack하에서는 다음과 같이 reduction에 의해 안전성을 증명할 수 있다. 즉 임의의 전송정보에 대하여도 R_i와 S_j를 임의로 선택하고 $g^{S_i} = g^{X_i} (g^{R_j S_i} = g^{R_j X_i})$, $g^{R_j S_j} = g^{X_j}$로 두어 oracle의 입력으로 주면 oracle은 그 출력으로 $g^{X_i X_j} \cdot g^{R_j X_j S_i}$을 줄 것이므로 이로부터 $g^{X_i X_j}$을 계산할 수 있다. (두번째 항은 S_i와 $g^{R_j X_j}$을 알고 있으므로 쉽게 계산할 수 있다.) 한편 COI attack의 경우는 reduction이 불가능하므로 프로토콜 분석을 통해 그 공격이 성공할 수 없음을 보일 수 있다. 즉 impersonation attack이 성공하기 위해서는 위의 세션키이 계산에서 볼 수 있듯이 공격자가 전송하는 $\underline{Z_j}$가 세션키이 계산에서 괄호안의 두번째 항인 $P_j^{R_i} \equiv_m g^{R_i S_j}$을 상쇄시켜 없어지게 할 수 있어야 한다. 그렇지 않을 경우 세션키이에 $g^{R_i S_j}$이 포함되어 공개정보와 전송정보로부터 이를 계산하는 것은 어렵기 때문이다. 그러나 공격자가 사용자에 의해 발생된 랜덤수 R_i가 포함된 그와 같은 전송정보를 보내는 것은 불가능하므로 COI attack하에서도 이 시스템을 깨는 것은 불가능하다. Known-key attack의 경우 알려진 세션키이는 앞의 두 예에서와 마찬가지로 어떤 공격에 대해서도 공격자가 다른 세션을 공격하는데 새로운 정보를 제공해 줄 수 없다는

것을 알 수 있다. 따라서 알려진 어떤 방법으로도 이 시스템을 깰 수는 없으므로 이 시스템은 최소한 그러한 공격하에서는 안전하다고 할 수 있다. ///

마지막으로 다음의 예는 GF(p)상에서 사용자의 비밀키이의 역원 S_I^{-1}을 이용하여 전송함수에 포함된 비밀키이 S_I를 제거하여 세션키이가 두 사용자가 각기 발생시킨 랜덤수에 전적으로 의존하도록 함으로써 어떤 impersonation attack에도 저항할 수 있도록 설계된 시스템이다. 이 시스템을 자세히 분석해 보면 전송함수 $g^{S_I K_I}$로부터 g^{K_I}을 구할 수 없는 한 어떤 공격에 의해서도 이를 깰 수 없다는 것을 알 수 있다. 물론 이 시스템에서는 각 사용자들이 자신의 비밀키이에 대한 역원을 구할 수 있어야 하므로 합성수 m을 공통 modulus로 사용하는 것은 불가능하며 또한 비밀키이 S_I의 역원이 존재하기 위해서는 이 비밀키이가 p-1과 서로소가 되도록(즉 gcd $(S_I, p-1)=1$) 선택해야 할 것이다.

[예 4.5.4] [MTI86]

- 사용자 i는 공개정보와 자신이 선택한 랜덤수 R_I를 이용하여 전송정보 $Z_I \equiv_p P_j^{R_I} \equiv_p g^{S_I K_I}$을 계산한 후 j에게 보낸다.
- 사용자 i는 modulo p로 S_I의 역원(multiplicative inverse) S_I^{-1}과 j로부터 받은 전송정보 Z_j를 이용하여 세션키이 $K_I \equiv_p Z_j^{S_I^{-1}} \cdot g^{R_I} \equiv_p g^{K_I + R_I}$ 혹은 $K_I \equiv_p Z_j^{R_I S_I^{-1}} \equiv_p g^{K_I R_I}$을 계산한다.

이상에서 살펴본 키이분배 방식들은 알려진 어떤 공격에 의해서도 이들을 깰 수 있는 방법이 없다는 의미에서 실제적인 면에서 안전한(practically secure) 시스템들로 볼 수 있을 것이다. 그러나 이들 방식들이 모두 시간이 많이 걸리는 모듈라 멱승을 세 번씩은 계산해야 한다는 점에서 실제로 구현하기에는 그 효율성이 문제가 되지 않을 수 없다. 모든 암호시스템들이 그러하듯이 우선은 안전성이 보장되어야 하겠지만 가능하면 간단하고 효율적이면서도 안전성을 잃지 않는 시스템을 설계하는 것이 모든 설계자들의 목표일 것이다. 이런 점에서 4.4절에서 살펴본 정보이론적 접근법을 계산상 어려움에 바탕을 두는 계산이론적 접근법과 결합한다면 보다 간단하면서도 안전한 시스템을 설계할 수 있을 것이다.

일반적으로 키이분배 방식에서 안전성 여부는 증명이 불가능한 경우가 대부분이므로 위에서 살펴본 바와 같은 모든 가능한 공격들에 대한 프로토콜분석은 안전한 시스템을 설계하는데 있어서 거쳐야 할 필수적인 단계일 것으로 생각된다. 그러나 이 프로토콜 분석법에 의한 안전성 고찰의 한계는 공격자가 이용할 수 있는 모든 가능한 공격방법이 시스템 설계자에게 알려져 있는가 하는 점이다. 즉 시스템 설계자에게 알려지지 않은 공격방법이 있을 수 있으며 또한 연구가 진척됨에 따라 예측치 못했던 새로운 공격방법이 개발될 수도 있을 것이다. 물론 이와 유사한 문제점은 대부분의 암호시스템이 바탕으로 하는 intractability assumption에도 존재하지만 (즉 이론과 기술이 발전함에 따라 어려운 것으로 알려졌던 문제들이 쉽게 풀릴 수 있다면 이를 바탕으로 하는 모든 암호시스템은 쉽게 깨어질 것이므로)위에서 언급한 문제점들은 좀더 시간을 두고 학계나 산업계의 시험을 받아야 할 것으로 보인다.

5. 결 론

본 논문에서는 키이분배 방식에 있어서 지금까지 알려진 가능한 모든 공격방법들을 분석하고 각 공격에 의해 쉽게 깨어질 수 있는 시스템 및 이를 깰 수 있는 시스템들을 예를 통해 살펴 봄으로써 새로운 키이분배 방식의 설계시 고려해야할 전송함수나 세션키이 계산함수들의 성질들을 알아보았다. 또한 키이분배 방식의 안전성을 증명할 수있는 도구로서 비록 제한된 적용범위를 가지나 안전성 검정에 있어서 매우 효과적인 reducibility test와, 보다 간단하고 안전한 전송함수나 세션키이 계산함수를 선택하는데 있어서 고려해야할 정보누출 여부를 점검할 수 있는 엔트로피(entropy) 함수, 그리고 다양한

공격방법들에 대해 주어진 시스템의 안전성을 프르토콜의 분석을 통해 점검하는 방법 등을 제시하였다. 또한 이미 제안된 방식들이나 설계예를 통해 그들의 안전성을 분석하는 과정에서 쉽게 깨어지지 않는 시스템이 가져야 할 성질들을 조사하여 새로운 키이분배 접근법을 제시하였다. 이와같은 기존방식들의 안전성 분석은 보다 간단하고 계산량이 적으면서도 안전한 시스템을 설계하는데 있어서 많은 도움이 될 것이며, 또한 모든 키이분배 방식에 일률적으로 적용될 수 있는 체계적인 증명법이 없는 상황에서 이러한 다양한 시도는 안전한 시스템의 설계나 분석시에 꼭 필요한 과정일 것으로 생각된다.

참 고 문 헌

[A79] L.Adleman, "A subexponential algorithm for the discrete logarithm problem with applications with cryptography", Proc. of 20th IEEE Symp. on Foundations of Computer Science(FOCS), 1979, pp. 55-60.

[Be70] E.R.Berlekamp, "Factoring polynomials over large finite fields", Math. comput., vol.24, 1970, pp. 713-735.

[B182] R.Blom, "Non-public key distribution", Proc. Crypto'82 : Advances in Cryptology, Plenum Press, 1983, pp. 231-236.

[B184] R.Blom, "An optimal class of symmetric key generation systems", Proc. Eurocrypt'84 : Advances in Cryptology(Lecture notes in Computer Science 209), Springer-Verlag, 1985, pp. 335-338.

[BCGL89] S.Ben-David, B.Chor, O.Goldreich, and M.Luby, "On the theory of average case complexity", Proc. 21rd ACM Symp. on Theory of Computing (STOC), 1989, pp. 204-216.

[BM88] M. Bellare and S.Micali, "How to sign any given trapdoor function", STOC, 1988, pp. 32-34.

[BK89] F.Bauspiess and H.Knobloch, "How to keep authenticity alive in a computer network", Proc. Eurocrypt'89.

[BW88] J.Buchmann and H.C.Williams, "A key-exchange system based on imaginary quadratic fields", J. Cryptology 1, 1988, pp. 107-118.

[C71] S.A.Cook, "The complexity of theorem proving procedures", STOC, 1971, pp. 151-158.

[D83] D.E.Denning, "Protecting public keys and signature keys", IEEE Computer, 16, 2, Feb. 1983, pp. 27-35.

[DH76] W. Diffie and M.E.Hellman, "New direction in cryptography", IEEE Trans. Inform. Theory, IT-22, 6, 1976, pp. 644-654.

[DS81] D.E.Denning and G.M.Sacco, "Timestamps in key distribution protocols", Commun. ACM, Vol. 24, No.8, 1981, pp. 533-536.

[E85] T.Elgamal, "A public key cryptosystem and a signature scheme based on discrete logarithm", IEEE Trans. Inform. Theory, IT-31, 1985, pp.469-472.

[FS86] A.Fiat and A.Shamir, "How to prove yourself : Practical solutions to identification and signature problems", Proc. Crypto'86 ; A dvances in Cryptology(Lecture notes in Computer Science 263), Springer-Verlag, 1987, pp. 186-194

[Go88] O.Goldreich, "Toward a theory of average case complexity(a survey)", TR-531, Computer Science Dept., Technion, Haifa, Israel, Mar. 1988.

[Gu87] Y.Gurevich, "Complete and incomplete randomized NP problems", FOCS, 1987, pp. 111 117.

[GJ79] M.R.Garey and D.S.Johnson, "Computer and intractability : A guide to the theory of NP-Complete", Freeman, San Francisco, 1979.

[GMRa85] S.Goldwasser, S.Micali and C.Rackoff, "The Knowledge complexity of interactive proof systems", STOC, 1985, pp. 291-304 and SIAM J. Comput., Vol.18, No.1, 1989, pp.186-208.

[GMRi88] S.Goldwasser, S.Micali and R.L.Rivest, "A digital signature scheme secure against chosen message attacks", SIAM J.Comput., Vol.17, No. 2, 1988, pp.281 308.

[HK89] L.Harn and T.Kiesler, "Improved Rabin's scheme with high efficiency", Elect. Letters, Vol. 25, No.11, 1989, pp. 726-728.

[IBM78] IBM Systems J., 17, No.2. 1978, pp. 106-150.

[ISO88] "Banking-key management(wholesale)", Int-

ernational Standard ISO8732, International Organization for Standardization, Geneva, 1988.

[ITW82] L.Ingemarsson, D.T.Tang, and C.K.Wong, "A conference key distribution system", IEEE Trans. Inform. Theory, IT-28, 1982, pp. 714-720.

[J84] D.S.Johnson, "The NP Complete column : an ongoing guide", J. Algorithms, Vol. 5, 1984, pp. 284-299 and pp. 433-447.

[Ka72] R.M.Karp, "Reducibility among combinatorial problems", Complexity of Computer Computations, R.E.Miller and J. W. Thatcher(eds.), Plenum Press, 1972, pp. 85-103.

[Ko87] N.Koblitz, "Elliptic curve cryptosystems", Math. Comput. 48, 1987, pp. 203-209.

[KIT87] K.Kurosawa, T.Ito, and M.Takeuchi, "Public key cryptosystem using reciprocal number with same intractability as factoring a large number", Elect. Letters, Vol.23, No.15, 1987, pp. 809-810.

[KW90] 권창영, 원동호, "공개키 분배에 관한 연구", 한국통신학회 논문지, pp. 981-898, 12/90.

[L84] L.A.Levin, "Average case complete problems", SIAM J. comput., 1986, Vol. 15, No.1, pp. 285-286. Extended abstract appeared in Proc. STOC, 1984, p.465.

[L88] L.A.Levin, "Homogenous measures and polynomial time invariants", FOCS, 1988, pp.36-41.

[LL90] C.S.Laih and J.Y.Lee, "Efficient probabilistic public key cryptosystem based on the Diffie-Hellman problem", Elect. Letters, Vol.26, No. 5, 1990, pp.326-327.

[Mi85] V.Miller, "Use of elliptic curves in cryptography", Proc. Crypto'85 : Advances in Cryptology (Lecture notes in Computer Science 218), Springer Verlag, 1986, pp.417-426.

[Mc88] K.S.McCurley, "A Key distribution system equivalent to factoring", J.Cryptology, Vol.1, No.2, 1988, pp. 95-106.

[Mc90] K.S.McCurley, "The discrete logarithm problem", Proc. Symp. in Applied Math., Vol. 42, 1990, pp. 49-74.

[MI87] T.Matsumoto and H.Imai, "On key predistribution system", Proc. Crypto'87 : Advances in Cryptology(Lecture notes in Computer Science 293), Springer Verlag, 1988, pp. 185-193.

[MI90] T.Matsumoto and H.Imai, "On the security of some key sharing schemes", The 1990 Symp. Cryptography and Inform. Security, Japan, 1990, pp.1-6(in Japanese).

[ML90] 문상재 : 이경종, "키이분배 프로토콜의 해약", 제 2회 정보보호 암호에 관한 workshop 논문집, pp.117-124, 6월, 9/90.

[MTI86] T.Matsumoto, Y.Takashima, and H.Imai, "On seeking smart public key distribution systems", Trans. IECE Japan, Vol.E.69, No.2, 1986, pp. 99-106.

[N86] R.Nobauer,"Key distribution systems based on polynomial functions and Redejfunctions", Probl. of Control and Inform. 1986, 15, pp.91-96.

[NY89] M.Naor and M.Yung, "Universal one way hash functions and their cryptographic applications", STOC, 1989, pp.33-34.

[O84] A.M.Odlyzko, "Discrete logarithms in finite fields and their cryptographic significance", Proc. Eurocrypt'84 : Advances in Cryptology (Lecture notes in Computer Science 209), Springer Verlag, 1985, pp.224-314.

[ON84] E. Okamoto and K. Nakamura, "A note on public key distribution system", 1984 Natl. Conf. Rec. on Comm. IECE Japan, 15, Oct. 1984.

[OO90] T.Okamoto and K.Ohta, "How to utilize the randomness of zero knowledge proofs", Proc. Cryto '90, pp. 437-456.

[OT89] E.Okamoto and K.Tanaka, "Key distribution system based on identification information", IEEE J. Select. areas commun., Vol.17, No.4, 1989,pp. 481-485.

[OVS84] R.W.K.Odoni, V.Varadharajan, and P.W.Sanders, "Public key distribution in matrix rings", Elect. Letters 20, 1984, pp. 386-387.

[PH78] S.C.Pohlig and M.E.Hellman, "An improved algorithm for computing logarithms over GF(p) and its crytographic significance", Trans. IEEE Inform, Theory, IT-24, No.1, pp.106-110, Jan.1978.

[R79] M.O.Rabin, "Digital signature and public key functions as intractable as factoring", TM-21

2, Lab. Computer Science, MIT, 1979.

[R80] M.O.Rabin, "Probabilistic algorithms in fintie fields", SIAM J. Comput., Vol.9, No.2, 1980, pp. 273-280.

[Sha79] A.Shamir, "How to share a secret, "Commun. ACM 22, 1979, pp. 612-613.

[Sha84] A.Shamir, "Identity-based cryptosystems and signature schemes", Proc. Eurocrypt'84, : Advances in Cryptology(Lecture notes in Computer Science 209), Springer-Verlas, 1985, pp. 47-53.

[Shm85] Z.Shmuely, "Composite Diffie-Hellman public key generating systems are hard to break", TR. No.356, Computer Science Dept., Technion-Istaei Institute of Technology, Feb. 1985.

[T87] H.Tanaka, "A realization scheme for the identity-based cryptosystem", Proc. Eurocrypt'87 : Advances in Crytology(Lecture notes in Computer Science 304). Springer-Verlag, 198 8, pp. 340-349.

[TI89] S.Tsujii and T.Itoh, "An ID-based cryptosystem based on the discrete logarithm problem", IEEE J. Select. areas commun., Vol.7, No. 4, 1989, pp. 467-473.

[V87] V.Varadharajan, "Cryptosystems based on permutation polynomials", Int. J. Computer Math., Vol.23, 1987, pp.237-250.

[W87] H.Woll, "Reductions among number theoretic problems", Information and Computation 72, 1987, pp. 167-179.

[Y90] Y.Yacobi, "A key distribution paradox", Proc. Crypto'90, pp. 245-255.

[YA83] T.Yamamoto and R.Akiyama, "A data encryption device incorporating fast PKDS", Proc. of IEEE Global Telecom. Conf., Nov.-Dec. 1983, pp. 1085-1090.

[YS89] Y.Yacobi and Z.Shmuely, "On key distribution systems", Proc. Crypto'89, pp. 344-355.

李 弼 中 (Pil Joong LEE) 正會員
1951年 12月 30日生
1974年 2月 : 서울大學校 電子工學科 學士
1977年 2月 : 서울大學校 電子工學科 碩士
1982年 6月 : U.C.L.A. System Science, Engineer
1985年 6月 : U.C.L.A. Electrical Engineering, Ph.D.
1976年 5月 ~ 1977年 2月 : 서울大學校 電子工學科 有給助教
1979年 5月 ~ 1980年 6月 : U.C.L.A. Post Graduate Research Engineer
1980年 6月 ~ 1985年 8月 : Jet Propulsion Laboratory, Senior Engineer
1985年 8月 ~ 1990年 2月 : Bell Communications Research, M.T.S.
1990年 2月 ~ 現在 : 浦項工科大學 電子電氣工學科 副教授

林 采 薰 (Chae Hoon LIM) 正會員
1963年 3月 11日生
1989年 2月 : 서울大學校 電子工學科 學士
1989年 1月 ~ 1990年 1月 : 韓國 데이타通信(株) 技術本部 研究員
1990年 2月 ~ 現在 : 浦項工科大學 電子電氣工學科 碩士過程 在學中
(Deok Hwan AN) (Tae Kyu YANG) (Sang Hyo LEE)