수신 보장성이 향상된 공평한 배달 증명 전자 메일 프로토콜

(A Fair Certified Electronic Mail Protocol that Enhances Guarantee of Reception)

박 용 수 † 조 유 근 ††

(Yongsu Park) (Yookun Cho)

요 약 배달 증명 전자 메일(certified e-mail)은 수신자가 메일을 받았을 때 그리고 그 때만 송신자가 영수증을 받는 시스템이다. 최근에 발표된 낙관적(optimistic) 기법에서는 이전 방법과는 달리 메일이 언제든지 수신자에게 전달될 수 있는 상태가 되면 수신 영수증이 발행된다. 따라서, 송신자는 수신 영수증을 확보하였음에도 불구하고 수신자가 실제로 메일을 받았는지 여부를 알 수 없다. 본 논문에서는 최근에 발표된 낙관적 기법의 문제점을 지적한 후, 수신 의무 시간 개념을 도입하여 기존 프로토콜을 보완한 내용을 제시한다. 수정된 프로토콜에서는 영수증의 의미를 수신 약속으로 변경시켜 위에서 언급한 문제점을 해결했다. 제시된 기법은 대부분의 낙관적 기법에 적용 가능하며 프로토콜 수행에 필요한 메시지 수를 더 이상 늘리지 않아 기존 기법에 비해 추가되는 부담이 거의 없다.

키워드 : 전자 서명, 전자 메일, 보안

Abstract Certified e-mail system guarantees that recipient will get mail content if and only if mail originator receives a receipt. Unlike previous schemes, Optimistic protocols recently published generate a receipt when it meets the condition that the mail content can be accessed by receiver at any time. So originator cannot assure the delivery of e-mail although he can get a receipt. In this paper, we show some flaws in optimistic protocols and propose improved schemes using delivery deadline. Modified protocols guarantee proof-of-receipt and eliminate the problem mentioned above. Furthermore, proposed modification technique can be applicable to most optimistic protocols and is efficient in the sense that modified schemes do not increase the number of messages.

Key words : digital signature, electronic mail, security

1. 서 론

최근 인터넷이 활성화됨에 따라, 전자 메일은 일상 생활에서 아주 중요한 통신 수단이 되었다. 기존 전자 메일 시스템이 상업적인 용도로 사용되기 위해서는 여러 가지 부가 기능이 필요하며, 그 중 하나가 메일의 배달을 송신자가 확인할 수 있는 배달 증명 전자 메일 기능이다[1]. 배달 증명 전자 메일 시스템에서 메일의 공평한 교환이란 수신자가 메일을 배달받았으면 (그리고 그럴때만) 송신자가 영수증을 받는 것이며[1,2], 본 논문에서 메일 배달 혹은 전달은 수신자가 메일 내용 전부를 전송받았음을 의미한다[1,3].

대부분의 실용적인 배달 증명 메일 시스템은 TTP (Trusted Third Party)를 사용하여 본 문제를 해결한다[1,2,3,4,5,6,7]. TTP가 항상 참여하는 초기 프로토콜[1,2,3]은 메일 수신에 대한 정의를 메일 배달과 동일한 의미로 두고 있어서 공평한 교환을 보장하지만, TTP가 보낸 데이타가 송/수신자에게 일정 시간 내 항상 전달된다는 비현실적인 가정을 하고 있다.

최근 발표된 낙관적 기법[4,5,6,7]은 TTP를 예외적인 경우에만 참여시켜서 TTP의 부하를 감소시켰을 뿐만 아니라, 이전 기법의 비현실적인 가정을 제거하였다. 그

† 비 회 원 : 서울대학교 전기컴퓨터공학부
yspark@ssrnet.snu.ac.kr
†† 종신회원 : 서울대학교 전기컴퓨터공학부 교수
cho@ssrnet.snu.ac.kr
논문접수 : 2002년 2월 18일
심사완료 : 2002년 11월 12일

러나, 낙관적 기법은 메일이 언제든지 수신자에게 전달될 수 있는 상태가 된 때를[1] 메일 수신이라 정의하고 이 때 영수증을 발행한다. 따라서, 송신자는 영수증을 확보했음에도 메일이 실제로 수신자에게 전달되었는지 여부를 알 수 없으며, 수신자는 영수증을 발행했음에도 메일의 배달을 무한정 늦출 수 있어 공평한 교환이 이루어지지 않는다.

위의 문제점에도 불구하고 몇몇 낙관적 기법은 영수증의 의미를 수신자가 메일을 전달받았음으로 간주한다[6,7]. 이 기법에서 송신자가 영수증을 확보하면, 수신자는 비록 메일을 받은 상태가 아니지만 받은 것으로 간주된다. 따라서, 수신자는 언제 생길지 모르는 분쟁에 대비하기 위하여 되도록 빨리 메일을 받아야하는 의무감을 가지게 된다.

본 논문에서는 위의 문제를 해결하기 위하여 "수신 의무 시간"을 도입하였다. 수신 의무 시간이란 수신자가 공용 DB(Database)나 TTP에 연결하여 메일 배달을 완료해야 할 마감 시간(deadline)을 의미한다. 수신 의무 시간을 도입하면, 영수증의 의미는 수신자의 수신 의무 시간 이내 수신 의무 이행에 대한 약속이 된다. 따라서, 송신자는 영수증을 확보하면, 시간 내 메일 배달을 기대할 수 있으며, 추후 분쟁 발생 시 불이행에 대한 책임을 물을 수 있다. 또한, 수신자는 당장 메일을 받을 필요가 없고 수신 의무 시간 내에 받으면 되어 의무감을 덜어낼 수 있다. 결국, 메일 수신의 의미가 수신자가 의무시간 내 메일 내용 전부를 전송받음으로 그 뜻이 환원되며, 수신 의무 시간 이후에는 수신자가 메일을 수신하면 그리고 수신할 때만 송신자가 영수증을 수신하게 되어 양자간 공평한 교환이 이루어진다.

본 논문의 전개순서는 다음과 같다. 먼저, 2장에서 기존 배달 증명 전자 메일 기법에 대해 분석하고, 3장에서 TRICERT와 Schneier가 만든 기법의 문제점을 지적한다. 4장에서 수신 의무 시간과 이를 이용한 개선된 기법을 설명한 후, 개선된 기법을 분석한다. 마지막으로 5장에서 결론을 내린다.

2. 요구 사항 및 관련 연구

먼저, 배달 증명 전자 메일이 갖추어야할 요건을 설명한 후, 2.1 절에서 기존 배달 증명 전자 메일 기법에 대해 분석한다. 배달 증명 전자 메일이 갖추어야 할 요건은 메일 내용의 기밀성, 프로토콜의 효율성 등 여러 가지가 있지만, 배달 증명 전자 메일의 기본 기능에 대하여 집중하며 다음과 같이 요약된다.

• 수신자와 송신자 사이 전자 메일과 영수증의 공평한 교환

이상적으로 송/수신자는 메일과 영수증 수신이 "동시에" 이루어지길 원하지만, 실제적으로 이를 구현하기는 거의 불가능하다. 결국, 한 사건이 종료 후, 다른 사건의 종료까지 시간 차이가 생기며, 이를 수식으로 표현하면 다음과 같다(단, event A는 사건 A를 의미하며, 명제 Fact(B)는 사건 B의 발생 여부를 나타낸다. 그리고, Time(C)는 명제 C가 참이 된 시간을 의미한다).

event A: 수신자가 송신자의 메일을 전달 받음

event B: 송신자는 수신자가 작성한 영수증을 확보

$$Fact(A) \Leftrightarrow Fact(B) \quad (1)$$

$$|Time(Fact(A)) - Time(Fact(B))| \leq t \quad (2)$$

위 식에서 두 사건이 일어난 시간 차이는 최대 t이며, 이 시간이 짧으면 짧을수록 이상적인 공평한 교환에 가까워진다[2].

• 현실성 있는 통신 채널

실용적인 배달 증명 전자 메일 프로토콜에서 TTP와 송/수신자 사이 통신 채널은 데이타가 일정 시간 내 항상 전달된다고 가정하고 있으며, [1]에서처럼 다수의 TTP를 분산시켜두고, 통신 실패 시 재접속 등의 방법을 이용하여 구현하면 된다고 언급하고 있다. 하지만, 일반적으로 전자 메일 환경에서 송/수신자는 항상 온라인 상태가 아니며, TTP로부터 데이타를 받았음에도 받지 않았다고 부인하는 등 프로토콜에 비협조적일 수 있음을 고려해 보면, TTP가 송신한 데이타가 일정 시간 내 수신자에게 전달된다는 가정은 매우 비현실적이며 구현하기가 힘들다.

다만, 메일 송/수신자가 TTP에 접속하여 데이타를 전송 시, TTP는 항상 올바르게 동작하고 여러 곳에 분산 되어있으므로 송/수신자가 통신 실패 시 다른 서버를 선택하여 재전송 할 의지만 있으면, 송/수신자로부터 TTP로의 데이타 전송은 일정 시간 내 항상 전달 가능하다고 가정할 수 있다.

2.1 관련 연구

관련 연구는 "공평한 교환(fair exchange)" 기법을 써서 송/수신자가 직접 정보를 교환하는 방법과 TTP (Trusted Third Party)를 이용하는 보다 실용적인 방법으로 나뉜다[2]. 전자의 경우, 이론적으로 여러 논문이 발

1) 일례로, 공용 DB (Database)에 암호화된 메일 내용을 풀 수 있는 키를 올려놓거나, 수신자에게 TTP의 비밀키로 암호화된 메일을 보내고 응답을 얻는 방법 등이 있다.

2) 최근 공평한 교환 프로토콜에서도 송수신자만 개입하는 것이

표되고 있지만[11,12], 통신 비용이 너무 높아 비현실적이다.

TTP를 사용하는 기법은 TTP를 항상 사용하는 기법[1,2,3]과 TTP를 낙관적으로 사용하는 기법[4,5,6,7]으로 나뉜다. TTP를 항상 사용하는 기법은 모두 TTP가 보낸 데이타가 송/수신자에게 일정 시간 내 항상 전달된다는 비현실적인 가정을 하고 있지만 식 (2)에서 t 값이 최대 전송 지연시간으로 한정된다.

최초의 배달 증명 메일 논문은 Bahreman과 Tygar가 발표하였다[2]. 이 논문에서는 기존 메일 시스템에 대한 문제점, 그리고 배달 증명 메일 시스템이 갖추어야할 요소를 밝혔으며, TTP를 사용하는 기법과 사용하지 않는 기법을 제시하였다. TTP를 사용하는 기법에서, 송신자는 TTP에게 메일을 전달하고 송신 증명 영수증을 받는다. TTP는 메일을 암호화하여 수신자에게 전달한 후, 수신자로부터 암호화된 메일에 대한 전자서명을 받는다. 그 후, TTP는 송신자에게 영수증을 전달하며 동시에 수신자에게 암호화 된 메일을 풀 수 있는 키를 전달한다. 이 기법은 식 (2)에서 t 값이 최대 전송 지연시간으로 한정된다. 또한, 송신자는 전자 서명된 메일을 보내지 않아 수신자의 위조 혹은 제 3자의 위조에 대한 위험을 안고 있다. 부가적으로, 이 프로토콜은 6개의 메시지를 주고받으며, 메일 내용에 대한 비밀은 보장되지 않는다.

Zhou와 Gollman가 제안한 기법은 TTP를 두어서 각 TTP가 MTA(Mail Transfer Agent)의 역할을 대신하게끔 하였다[3]. TTP는 항상 제대로 동작한다고 가정하므로, 전자 메일을 중간에 유실없이 확실하게 보낸다. 하지만, 여러 개의 TTP들을 묶어서 1 개로 볼 때, [2]의 TTP를 사용하는 프로토콜과 거의 동일하다. 따라서, 이 기법도 t 값이 최대 전송 지연 시간으로 한정된다. 그 이유는 비록 송신자는 영수증을 받지 못했지만, TTP가 가지고 있으므로 분쟁시 효력을 발생하기 때문이다. 또한, 수신자의 위조 혹은 제 3자의 위조에 대한 위험을 안고 있고, 메시지 수가 많으며, 메일 내용에 대한 비밀은 보장되지 않는다.

TTP를 항상 사용하는 보안 메일 프로토콜은 Deng 등이 고안한 기법[1]이 제일 발전된 형태이다. 먼저 송신자는 수신자에게 TTP만 볼 수 있게끔 암호화한 메일을 주며, 수신자는 메일을 받아 TTP에게 전달한다. TTP는 전자서명 확인을 통하여 메일 송신자를 확인 후, 수신자에게는 메일을 수신자의 공개키로 암호화하여 주고, 송신자에게는 메일에 대하여 수신자와 TTP가 서명한 영수증을 전달한다. 이 프로토콜 역시 t 값이 최대 전송 지연시간으로 한정된다. 송신자는 메일을 전자서명해서 보내기 때문에 수신자의 위조 혹은 제3자의 위조에 대한 위험은 없지만, 서명을 수신자가 확인할 수 없기 때문에 수신자는 DoS(Denial of Service) 공격에 당할 수 있다. 이 프로토콜 역시 메일 내용에 대한 비밀은 보장되지 않는다.

최근의 논문에서는 항상 TTP가 참여하는 것이 아니라 예외적인 경우에만 참여하는 낙관적인 프로토콜이 연구되었다[4,5,6,7]. 이들 프로토콜은 TTP의 부하를 감소시켰을 뿐만아니라, 위에서 언급한 TTP와 송/수신자 사이에 데이타가 일정 시간 내 항상 전달된다는 비현실적인 가정을 없앴다. 낙관적인 프로토콜은 크게 2가지로 나뉜다. 첫째는 송신자가 TTP의 공개키로 암호화한 메일을 보내고 수신자는 이것을 서명하여 송신자에게 줌으로써 송신자가 event B를 확보하는 방식이며, 둘째는 예외 발생 시 송신자가 TTP나 DB에 메일을 볼 수 있는 수단을 올려놓음으로써 영수증을 발생시키는 방법이다.

최초의 낙관적인 프로토콜은 Micali의 논문이다[6]. TRICERT[7]와 Schneier가 고안한 기법[4]은 가장 최근 발표된 낙관적인 프로토콜이며, 이들 기법에 대한 설명 및 문제점은 3 .1 절 및 3.2 절에서 설명한다. Puigserver의 방법[5]은 Schneier 기법의 변형이며, Schneier 기법의 문제점을 그대로 안고 있다. 따라서, 이 기법의 문제점은 본 논문에서 따로 언급하지 않는다.

3. TRICERT와 Schneier의 기법 분석

3.1 절과 3.2 절에서 기존 배달 증명 메일 시스템 중 최근에 발표된 낙관적 기법인 TRICERT와 Schneier의 기법을 분석하며 그 문제점을 살펴본다. 본 논문에서 사용하는 용어의 의미는 다음과 같다. PH는 프로토콜 헤더로써, 송/수신자 ID, 타임스탬프 그리고, nonce 값을 가지고 있어서 재전송 공격을 방지한다. PA(B)는 문서 B를 A의 공개키로 암호화한 암호문이며, SigC(D)는 문서 D를 C의 개인키로 전자서명한 서명문이다.

3.1 TRICERT 분석

그림 1은 TRICERT 프로토콜을 보여준다. 그림에서 PA는 전자 우편 배달 디몬(daemon) 프로세스로, PA와 연결은 단절될 수 있고, PA는 오동작할 수 있으며, 수

아니라 TTP를 이용하는 방법이 나오고 있다[9,10]. 하지만, 이들 방법은 cut and choose 방법을 사용하여 매우 복잡하거나, TTP가 프로토콜 도중 몇몇 메시지를 취소(revoke)해야하는 등 실용적인 배달 증명 메일 시스템에 사용하기에는 부적합하다.

신자와 결탁할 수 없다고 가정한다(송신자가 여러 PA 중 하나를 선택한다).

1. 먼저, 송신자는 메일 M을 수신자의 공개키로 암호화한 후, TTP의 공개키로 암호화하여 C를 만들고 서명 메시지 S를 만들어 PA에게 전송한다.
2. PA는 전자서명 S를 확인한 후, S와 SigPA(PH'||C)를 수신자에게 전달한다.
3. 수신자는 영수증 R을 PA에 보낸다.
4. PA는 영수증 R을 송신자에게 전달함과 동시에 (4번째 메시지) 수신자의 공개키로 암호화된 메일을 수신자에게 보내준다(5번째 메시지).
5. 수신자는 PA로부터 암호화된 메일을 받지 못한 경우, TTP에 접속하여 S와 SigPA(PH'||C)를 보내며 (6번째 메시지), TTP는 서명을 검증한 후 C를 복호화하여 수신자에게 되돌려준다(7번째 메시지).

이 프로토콜의 문제점은 세 번째 메시지에서 PA가 영수증을 확보했을 때, 수신자는 비록 메일을 수신한 상태가 아니지만, 수신 영수증을 발행한 상태가 된다는 점이다. PA는 오동작할 수 있고, 송신자가 단독으로 지정하므로 송신자와 결탁하여 영수증을 받고 메일을 주지 않을 수 있다. 따라서 수신자는 언제 생길지 모르는 분쟁에 대비하기 위하여, TTP에 접속하여 되도록 빨리 메일을 수신해야하는 의무감이 생긴다. 결과적으로 TRICERT는 수신자에게 불리한 프로토콜이다.

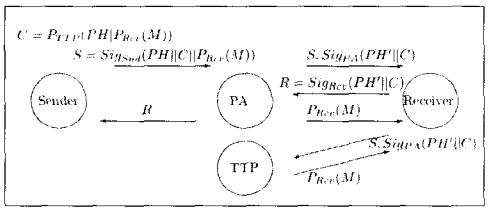


그림 1 TRICERT

3.2 Schneier의 기법 분석

그림 2는 Schneier의 기법을 보여준다. 프로토콜은 다음과 같다.

1. 먼저 송신자는 대칭 키 K를 생성한 후 메일을 암호화하여 수신자에게 보낸다.
2. 수신자는 받은 메시지를 전자서명하여 송신자에게 보낸다. 이 때, 송신자가 키 값을 올려놓을 DB의 위치 X와 송신 의무시간 T'를 같이 보내준다.
3. 송신자는 키 K를 수신자에게 보낸다.
4. 수신자는 받은 키 값을 전자서명한 영수증을 보낸다.
5. 송신자는 영수증을 못 받게 된 경우, 송신 의무시간 T' 내에 공용 DB X에 키를 올려놓는다.

프로토콜의 마지막 5번째 메시지가 DB로 전송되었을 때 영수증이 생긴다. 하지만, 이 영수증은 송신자가 암호화된 메일과 키를 보냈다는 것밖에 확인할 수 없으며, 수신자는 메일의 수신을 무한정 늦출 수 있어 형평성이 크게 어긋난다. 따라서, 송신자는 메일의 수신 여부를 확인하기 위하여 부가의 작업(일례로, 전화로 확인하거나 팩스를 보내는 등)을 해야 한다.

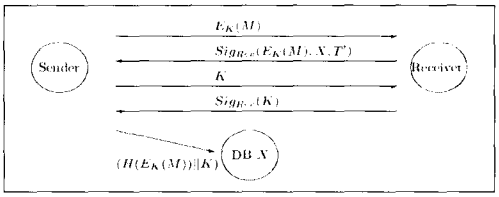


그림 2 Schneier의 기법

4. 수신 의무 시간을 가지는 개선된 기법

4.1 절에서 수신 의무 시간에 대해 설명하고, 수신 의무 시간을 정하는 프로토콜을 제안하며, 4.2 절과 4.3 절에서 이를 이용한 개선된 TRICERT와 개선된 Schneier의 기법을 제안한다. 그 후, 4.4 절에서 개선된 기법을 분석한다. 본 논문에서 사용되는 모든 시간은 TTP의 시간으로 정하며 이는 Schneier 논문의 송신 의무 시간과 동일한 기준이다.

4.1 수신 의무 시간

본 논문에서 제시하는 수신 의무 시간은 수신자가 이 시간 내 메일을 수신해야 하는 마감 시간을 의미한다. 수신 의무 시간(=T)을 도입하면, 식 (2)에서 t 값이 다음과 같이 한정된다.

$$| \text{Time(Fact(A))} - \text{Time(Fact(B))} | \leq t$$

$$t = T - \text{Min}(\text{Time(Fact(A))}, \text{Time(Fact(B))})$$

송신자는 수신 의무 시간 T 값을 되도록 작게 정하고 싶어 할 것이며, 수신자는 되도록 크게 정하고 싶어하기 때문에, 송신자 혹은 수신자가 일방적으로 결정하면 안되며 상호 협의를 거쳐야 한다. 본 논문에서 제안하는 수신 의무 시간 협의 프로토콜은 다음과 같다.

1. 송신자는 수신 의무 시간의 최소값과 최대값을 정하여 수신자에게 전달
2. 수신자는 최소값과 최대값 사이로 시간을 정하여 수신자에게 전달, 혹은 수신 거부 메시지를 수신자에게 전달

한편, 다음과 같은 약식 수신 의무 시간 협의 프로토콜을 사용할 수 있다.

1. 송신자는 수신 의무시간을 정하여 수신자에게 전달
2. 수신자는 이 값에 동의/거부 여부를 수신자에게 전달

약식 프로토콜에서는 송신자가 일방적으로 수신 의무 시간을 제시하며 수신자는 동의/거부만 결정할 수 있어 식 (2)에서 t 값이 수신자에 의해 줄어들 수 있는 여지가 없어진다. 하지만, 이 방법을 기존 낙관적인 프로토콜에 적용시 대부분의 경우 메시지 수가 늘지 않는 장점이 있다(개선된 TRICERT에서 약식 프로토콜을 사용하였다).

수신 의무 시간 협의 프로토콜이 성공적으로 완료되면, 송/수신자는 다음과 같은 사항에 대해 동의한 것이다.

1. 송/수신자는 쌍방이 정한 (혹은 송신자가 제시한) 수신 의무 시간에 대해 동의한다.
2. 송신자는 수신 의무 시간 후에 제 3자에게 분쟁을 제기할 수 있다. 즉, 수신 의무 시간 전에 분쟁 제기는 무효가 된다.
3. 수신자는 수신 의무 시간 내에 수신을 약속하며, 수신 의무 시간 이후 메일을 수신하지 않는데 대한 책임을 진다.

수신자는 송신자가 제시한 수신 의무 시간이 무리한 조건이라고 생각되면, 프로토콜을 중단하여 수신을 거부한다. 프로토콜이 중단되면, 메일 송신과 영수증 확보가 되지 않으므로, 2장에서 언급한 사건 A, B가 발생하지 않아서 공평함에 위배되지 않는다.

4.2 개선된 TRICERT

우선, TRICERT를 수행하기 이전에 수신 의무 시간 협의 프로토콜을 수행하는 방식을 생각해볼 수 있다. 하지만, 이 방법은 메시지 수가 너무 많고, 송신자가 협의 프로토콜 이후 TRICERT를 수행하지 않을 수 있다. 본 논문에서는 TRICERT의 메시지 내 수신 의무 시간 협의 메시지를 넣어 프로토콜을 최적화하였다. 그림 3은 수정된 프로토콜을 보여주고 있다.

1. 송신자는 먼저 메일 M을 수신자의 공개키로 암호화 한 후, TTP의 공개키로 암호화하여 C를 만들고 메시지 S를 만들어 PA에 보낸다. 이 때, 수신 의무 시간 T를 같이 넣어서 보낸다.
2. PA는 S 내 전자서명을 확인 후, S와 SigPA (PH'||C||T)를 수신자에게 전달한다.
3. 수신자는 영수증 R을 PA에 보낸다. 이 때, R을 보내는 의미는 수신 의무 시간에 동의함을 나타낸다.
4. PA는 영수증 R을 송신자에게 전달함과 동시에 (4번째 메시지) 수신자의 공개키로 암호화된 메일을 수신자에게 보내준다(5 번째 메시지).
5. 수신자는 PA로부터 암호화된 메일을 받지 못한 경우, TTP에 접속하여 S와 SigPA(PH'||C||T)를 보내며(6 번째 메시지), TTP는 서명을 검증하고 수신 의무 시간 내 요청이 왔는지 검사 후, C를 복호화하여 수신자에게 되돌려준다(7 번째 메시지).

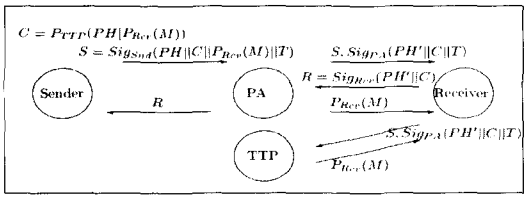


그림 3 개선된 TRICERT

TRICERT에서 PA와 송신자는 서로 신뢰한다고 가정하였으므로, 두 개체를 동일하게 볼 수 있고 첫 번째 메시지와 네 번째 메시지는 내부 메시지로 간주할 수 있다. 그러면 수정된 TRICERT를 기존 기법과 비교해 볼 때, 두 번째와 여섯 번째 메시지에 수신 의무 시간이 들어간 점이 차이가 난다. 개선된 TRICERT에서 두 번째 메시지를 수신자가 받은 경우, 수신자는 TTP의 공개키로 암호화된 메일과 함께 수신 의무 시간을 전달받는다. 메일을 받고 싶으면, 영수증(세 번째 메시지)을 PA에 돌려주며 이 때, 기존 방법에서는 영수증이 메일을 전달받음을 의미하였지만, 수정된 방법에서는 수신 의무 시간 내에 메일을 받겠다는 약속을 의미한다. 따라서, 분쟁이 일어날 수 있는 시점이 수신 의무 시간 이후로 미루어지며 다음과 같은 문제점이 해결된다. 첫째, 송신자가 영수증을 확보하자마자 클레임을 거는 문제점이 해결되고, 둘째, 수신자는 이런 문제점을 해결하기 위하여 영수증을 보내자마자 TTP에 접속하여 메일을 수신해야하는 조급함으로부터 해방된다. 결과적으로, 식 (2)에서 t 값이 기존 기법은 무한대였음에 반해 수정된 기법은 |Time(Fact(세번째 메시지가 도착))-수신 의무 시간|으로 한정된다.

4.3 개선된 Schneier의 기법

그림 4는 개선된 Schener의 기법을 보여주고 있으며, 이에 대한 설명은 다음과 같다.

1. 송신자는 먼저 대칭 키 K를 생성한 후 메일을 암호화하여 수신자에게 보낸다. 이 때, 수신 의무 시간의 최대/최소값을 보낸다.
2. 수신자는 암호화된 메시지, 송신자가 키 값을 올려놓을 DB의 위치 X와 송신

의무 시간 T'를 전자 서명하여 송신자에게 보낸다. 이 때, 송신자가 제시한 수신 의무 시간의 최대/최소값 내에서 수신 의무 시간을 결정하여 이를 전자 서명하는 메시지에 추가한다.

3. 송신자는 키 K를 직접 수신자에게 보낸다.
4. 수신자는 받은 키 값을 전자 서명한 영수증을 보낸다.
5. 송신자는 영수증을 못 받게 된 경우, 송신 의무시간 T' 내에 공용 DB X에 키를 올려놓는다.

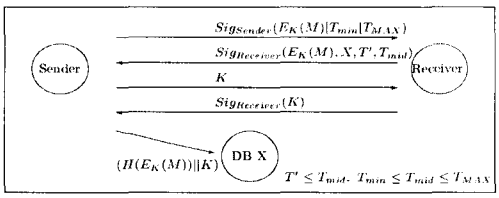


그림 4 개선된 Schneier의 기법

수정된 Schneier의 프로토콜을 기존 프로토콜과 비교해보면, 첫 번째 메시지에서 수신 의무 시간의 최대/최소값이 주어지며, 두 번째 메시지에서 송신자가 제시한 수신 의무 시간의 최대/최소값 내에서 수신 의무 시간을 결정한 값이 들어있다. 이 때, 기존 프로토콜에서는 송신 의무 시간만을 포함하는데, 이 때문에 송신자만 제한 시간 내에 메일을 공용 DB에 올려놓아야 하는 책임을 가지며, 수신자는 메일 수신을 무한정 연기할 수 있게 되어 식 (2)에서 t 값이 무한대가 된다. 이에 반해 수정된 기법은 두 번째 메시지에서 송신 의무시간과 수신 의무 시간을 함께 정하여 식 (2)에서 t 값이 |Time (Fact(두번째 메시지가 도착))-수신 의무 시간|으로 한정된다.

4.4 분 석

1980년 후반 이래 BAN 로직, CSP 등의 기법을 이용하여 인증 프로토콜의 안전성 분석에 대한 연구가 진행되어 왔다. 하지만, 공평한 교환 프로토콜 및 배달 증명 메일 프로토콜에 대한 분석은 인증 프로토콜 분석과는 매우 다르며 최근 시도되고 있다[13,14,15,16]. 이들 논문에서 사용하는 분석툴이 BAN 로직, SVO, CSP, Murphi 등 각 논문마다 전부 다르며, 분석 또한 부인방지에 대한 분석을 주로 다루고 있고, 공평함에 대한 분석은 가장 최근 논문인 [16]에서만 다루고 있다.

본 논문에서는 [16]에서 사용한 방법인 ATS (Alternating Transition System)를 이용하여 TRI CERT, Schneier의 프로토콜, 그리고, 개선된 프로토콜을 모델링하며, MOCHA 시스템 검증기를 이용하여 각 기법을 분석한다. 자세한 모델링 작업 절차는 [16]에 기술되어 있다. 본 논문에서는 TRICERT와 Schneier의 프로토콜이 본 논문에서 지적한 결점을 제외하고는 원활히 수행된다는 가정 아래, 지적된 결점이 어떻게 해결되었는지에 대해 집중한다.

ATS는 여러 선수(player)가 서로 게임을 하는 시스템이다. 각 선수는 매 스텝(step)마다 현재 상태(state)에서 가능한 선택을 하고, 모든 선수의 선택으로 다음 스텝의 상태를 결정한다(명확한 정의는 [16]에 있음). 본 논문에서 사용한 ATS에는 다음과 같은 선수가 있다: "Originator"(송신자), "Recipient"(수신자), "TTP", "Reliable_comm", "Unreliable_comm", "Judge"(재판관), 그리고, "PA"가 있다. [16]에서 설명되었듯이, Reliable_ comm은 데이타가 일정 시간 내 항상 전달되는 통신을 모델링한 것이며, Unreliable_comm은 일반적인 통신을 모델링한 것이다. Judge는 송/수신자 사이에 분쟁을 해결하는 재판관 역할을 하며, TTP는 TRICERT의 TTP 혹은 Schneier 기법의 DB를 모델링한 것이다. 또한, TTP는 전역 시간(global clock)을 관리하고 모든 선수는 현재 시간을 얻기 위하여 이 값을 읽을 수 있다. Judge와 TTP는 Reliable_comm과만 데이타를 교환한다(즉, 송/수신자와 재판관 혹은 TTP 사이의 통신은 신뢰할 수 있음을 가정한다).

각 선수들은 매 스텝마다 가능한 전략 중 하나를 선택한다(예를 들어, 멈추거나, 가만히 있거나, 특정 변수 값을 변경한다). 단, 선수들 중, TTP, Reliable_comm, Judge는 매 상태에서 단일 전략(올바른 동작)을 선택한다. 우리는 보호 명령(guarded command)을 이용하여 각 선수의 가능한 전략을 기술하였다. 일례로, 아래 식은 Schneier의 기법에서 사용된 보호 명령의 일부를 보여준다.

```
...
[] ~STOPoriginator & Originator_gets_Key & Originator_gets_
   Sig(EK(M),X,T') & ~Send_MSG3
    -> Send_MSG3':=true
...
[] Send_MSG3 & ~Receive_MSG3 -> Receive_MSG3':=true
[] true ->
...
[] ~STOPrecipient & Receive_MSG3 & ~Recipient_gets_Key
   -> Recipient_gets_Key':=true
[] ~STOPrecipient & ~Recipient_gets_mail & Recipient
   _gets_encrypted_mail
   & Receipient_gets_Key -> Recipient_gets_mail':=true
...
```

위 식에서, STOPoriginator, Originator_gets_Key, Receive_MSG3, ... 등은 Boolean 변수이다. 보호 명령은 '[]E->F'의 형식으로 표시되며, E는 조건을 의미하고, F는 E 조건을 만족시킬 때 변수에 대한 갱신 내역을 의미한다. 각 선수는 보호 명령의 집합을 가지고 있어서, 매 스텝마다 보호 명령들의 조건 중, 참인 조건 중 하나를 선택하여, 해당 변수를 갱신한다. 위 식에서, 첫째 보호 명령은 송신자에 해당되며, 송신자가 멈추지 않았고(~STOPoriginator), 송신자가 키 값을 가지고 있으며(Originator_gets_Key), 송신자가 수신자가 보낸 두 번째 메시지를 가졌고(Originator_gets_Sig(EK(M),X,T'), 송신자가 세 번째 메시지를 송신하지 않았으면(~Send MSG3), 세 번째 메시지를 송신한다는(Send_ MSG3' := true) 뜻이다. 두 번째와 세 번째 보호 명령은 Unreliable_comm에 해당되며, 송신자가 세 번째 메시지를 송신하고 수신자가 세 번째 메시지를 받지 않았으면 수신자에게 세 번째 메시지를 전달하거나, 아무 일도 하지 않는다([]true->). 네 번째와 다섯 번째 보호 명령은 수신자측에 해당되며, 세 번째 메시지를 받은 경우 키 값을 얻어 암호화된 메일을 해독하는 부분에 해당된다.

우리는 ATL(Alternating-time Temporal Logic)을 사용하여, 기존 기법의 문제점을 검사하고 개선된 기법에서 지적된 문제가 발생하는지 검사하였다. 본 논문에서 사용하는 ATL은 <<G>> ◇ H의 형식만을 사용하며, 그 의미는 선수들이 집합 G와 집합 Gc의 두 편으로 나뉘어 경쟁할 때, G가 목적 명제 H를 결과적으로 참이 되게 하는 (eventually true) 전략이 있는지 여부를 나타낸다. MOCHA는 ATS와 ATL을 지원하는 시스템 모의 실험 및 검증 도구이며, 이를 이용하여 기존 기법과 개선된 기법을 모델링하고 검사하였다.

프로토콜 검증을 단순화하기 위하여, Originator와 Recipient는 두 사건 (event A, B) 중 어느 하나가 발생할 때까지는 프로토콜을 중단하거나 연기할 수 없다고 가정한다. 그 동안에는 공평함이 보장되므로 이 가정은 합리적이다. 아래 식은 수신자 측에서 이 가정을 보호 명령으로 표현한 것이다. 첫째 식은 수신자가 아직 멈추지 않고 (~STOPrecipient) 송신자가 영수증을 확보하거나 수신자가 메일을 수신하면 수신자는 멈출 수 있다는 (즉, 프로토콜에 더 이상 참가하지 않음) 뜻이며, 둘째 식은 동일 조건에서 수신자는 한 스텝 쉴 수 있다는 뜻이다. 송신자도 유사한 방법으로 표현하였다. 본 절에서 TTP가 생성한 영수증 혹은 송신자가 DB에 키 값을 올려놓음으로써 생기는 영수증을 "Affidavit"이라고 부른다.

```
[] ~STOPrecipient & (Affidavit|Receipt|Recipient_gets_mail)
    -> STOPrecipient':= true
[] ~STOPrecipient & (Affidavit|Receipt|Recipient_gets_mail) ->
```

4.4.1 TRICERT와 Schneier의 기법 분석

TRICERT의 결점을 검사하기 위하여 다음과 같은 보호 명령을 설정하였다:

```
[] Judge_receives_Receipt_or_Affidavit_from_Originator &
   ~Recipient_gets_mail
  => Unfair_case':= true
```

이 식은 Judge가 Originator로부터 영수증을 받고 Recipient가 메일을 수신하지 못하면 Unfair_case는 참이 된다는 의미이며, 이는 3.1 절에서 지적된 문제점이다. 우리는 MOCHA를 사용하여 다음과 같은 ATL 식이 참임을 확인하였다:

<<Originator, PA, Unreliable_comm>>
◇ (Unfair_case) (3)

즉, TRICERT에서 Originator, PA, 그리고 Unreliable_comm이 결탁하면 Unfair_case가 발생할 수 있으며, Fact(B)⇏Fact(A)임이 확인된다.

Schneider의 기법의 결점을 검사하기 위하여 다음과 같은 ATL 식을 검사하였다:

<<Recipient, Unreliable_comm>>
◇ (Receipt ∨ Affidavit) (4)

이 식은 Recipient와 Unreliable_comm이 결탁하면 언제든지 Originator가 영수증을 확보할 전략이 있음을 의미한다. 이는 두 사건 중 하나가 일어나기 전에는 송/수신자 모두 프로토콜을 준수한다는 가정에 의한 것이다.

<<Recipient, Unreliable_comm>> ◇ ((Receipt ∨ Affidavit) ∧ ¬ Recipient_gets_mail) (5)

위 식은 Recipient와 Unreliable_comm이 결탁하면 Originator가 영수증을 확보하지만 Recipient는 메일을 수신을 무한정 지연할 수 있음을 뜻하며 이는 3.2 절에서 지적된 문제점이다. MOCHA는 식 (4), (5)가 참임을 보였으며, 따라서, Schneier의 기법은 Fact(B)⇏Fact(A)임이 확인되었다.

4.4.2 개선된 기법 분석

먼저 개선된 Schneier의 기법에서 수신 의무 시간이 결정되면, Recipient는 수신 의무 시간 이전에 메일을 수신해야하는 의무를 가진다. Recipient가 수신 의무 시간 이전에 행동을 멈추거나 지연시킴을 방지하기 위해

다음과 같이 수신자의 보호 명령을 변경하였다. Deli very_deadline은 초기에 무한대로 설정되며, 프로토콜 중에 협의된 값으로 설정된다.

```
[] ~STOPrecipient & (Affidavit|Receipt|Recipient_gets_mail)
   & (clock > Delivery_deadline)
   -> STOPrecipient':= true
[] ~STOPrecipient & (Affidavit|Receipt|Recipient_gets_mail)
   & (clock > Delivery_deadline)
   ->
```

위와 같이 개선된 Schneier의 기법에 맞춰 보호 명령을 변경한 후, MOCHA에서 ATL 식 (5)를 다시 검사한 결과, 거짓임을 확인하였다. 더 나아가, 다양한 조건의 ATL 식을 검사하였으며, 개선된 기법에서 이전 기법과 같은 결점을 발견할 수 없었다.[3]

개선된 TRICERT에서도 위와 같이 보호 명령을 변경하고, 다음과 같이 Judge가 분쟁을 해결하는 시점을 수신 의무 시간 이후로 미루었다:

```
[] Judge_receives_Receipt_from_Originator & ~Recipient_gets
   _mail
   & (clock > Delivery_deadline) => Unfair_case':= true
```

MOCHA를 이용하여, ATL 식 (3)을 다시 검사한 결과, 거짓임을 확인하였다. 더 나아가, 다양한 조건의 ATL 식을 검사한 결과, 개선된 기법에서 이전 기법과 같은 결점을 발견할 수 없었다.

5. 결 론

본 논문에서는 최근 낙관적 배달 증명 전자 메일 프로토콜의 문제점을 지적하였으며, 이를 해결하기 위하여 수신 의무 시간의 개념을 도입하여 기존 프로토콜을 수정, 보완하는 내용을 제시하였다. 기존 프로토콜에 수신 의무 시간 협의 과정이 추가됨으로써 메일의 수신과 영수증 확보의 최대 시간 차이를 한정지었으며, 이로 인해 기존 기법에서 수신자가 수신을 무한정 늦추거나 수신자가 과도한 의무감을 가져야 하는 문제점을 해결했다. 제시된 개선 기법은 프로토콜을 최적화시켜 메시지 개수를 더 이상 늘리지 않았으며, 대부분의 기존 낙관적인 배달 증명 전자 메일 프로토콜에 적용될 수 있다.

3) 우리는 ATS 시스템의 선수들을 두 편 (G, G^c)으로 나눌 때, 가능한 모든 경우에 대해 ATL 식을 작성하여 검사했으며, 각 그룹 G의 목적 명제를 다양하게 만들어 검사했다.

참 고 문 헌

[1] R. H. Deng, L. Gong, A. A. Lazar, and W. G. Wang, Practical protocols for certified electornic mail, Journal of Network and System Manage ment, vol. 4, no. 3, pp. 279~297, 1996.

[2] A. Bahreman, Certified electornic mail, in Procee dings of Symposium on Network and Distributed Systems Security, pp. 3~19, 1994.

[3] J. Zhou and D. Gollman, Certified electronic mail, in ESORICS'96, pp. 55~61, 1996.

[4] B. Schneier and J. Riordan, A certified e-mail protocol, in ACSAC'98, pp. 232~238, 1998.

[5] M. M. Puigserver and J. L. F. Gomila, Certified electronic mail protocol resistant to a minority of malicious third parties, in Infocom'00, pp. 1401~1405, 2000.

[6] S. Micali, Simultaneous electronic transactions. technical report 566420, tech. rep., 1995.

[7] G. Ateniese, B. Medeiros, and M. T. Goodrich, TRICERT: A distributed certified e-mail sche me, in NDSS'01, pp. 47~58, 2001.

[8] D. K. Pradhan, Fault-Tolerant Computer System Design. Prentice-Hall Inc., 1996.

[9] N. Asokan, V. Schoup, and M. Waidner, Optimis tic fair exchange of digital signatures, IEEE Journal on Selected Area in Communications, vol. 18, no. 4, pp. 593~610, 2000.

[10] N. Asokan, V. Schoup, and M. Waidner, Asynch ronous protocols for optimistic fair exchange, in IEEE Symposium on Research in Security and Privacy, pp. 86~99, 1998.

[11] S. Even, O. Goldreich, and A. Lempel, A randomized protocol for signing contracts, in CRYPTO'82, pp. 205~210, 1982.

[12] M. Ben-Or, O. Goldreich, S. Micali, and R. Rivest, A fair protocol for signing contracts, IEEE Transactions of Information Theory, vol. 36, no. 1, pp. 40~46, 1990.

[13] C. Boyd and P.Kearney, Exploring fair exchange protocols using specification animation, in ISW '2000, pp. 209~223, 2000.

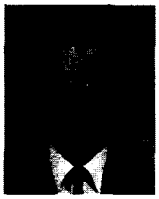
[14] S. Schneider, Formal analysis of a non-repudiation protocol, in CSFW'98, pp. 54~65, 1998.

[15] V. Shmatikov and J.Mitchell, Analysis of a fair exchange protocol, in NDSS'00, pp. 119~128, 2000.

[16] S. Kremer and J.-F. Raskin, A Game-Based Verification of Non-Repudiation and Fair Ex change Protocols, in CONCUR'01, Accepted for publication, 2001.

박 용 수

1992년 ~ 1996년 한국과학기술원 전산학과(학사). 1996년 ~ 1998년 서울대학교 컴퓨터공학과(석사). 1998년 ~ 현재 서울대학교 컴퓨터공학과 박사과정. 관심분야는 정보보안, 네트워크보안, 암호학



조 유 근

1971년 서울대학교 건축공학과 학사 1978년 미네소타대학교 컴퓨터과학 박사 1979년 ~ 현재 서울대학교 컴퓨터공학부 교수. 1984년 ~ 1985년 미네소타대학교 교환 교수. 1993년 ~ 1995년 서울대학교 중앙교육연구전산원장. 1999년 ~ 2001년 서울대학교 공과대학 부학장. 2001년 ~ 2002년 한국정보과학회 회장. 관심분야는 운영체제, 알고리즘 설계 및 분석, 암호학