通信情報保護學會誌
第3卷 第4號, 1993. 12

메세지 체인 방식에 의한 인증 알고리즘에 관한 연구†

A Study on the Authentication Algorithm Based on the Message Chainning

안효범* · 박창섭**

요 약

본 연구에서는 기존의 메세지 인증 방식인 MAC(Message Aunthentication Code)와 MDC(Manipulation Detection Code)에 대한 제 3자의 적극적인 공격(active attack)하에서의 구조적인 취약점을 분석하고, 이를 보완하는 새로운 인증 방식을 제안하고 검증하였다. 새로운 메세지 인증 방식은 기존의 방식과는 다르게 제 3자의 공격을 받은 메세지 블럭을 바로 검출할 수 있다는 장점을 가지고 있기 때문에 통신 시스템 상에서 불필요한 메세지의 재전송을 줄일 수 있다는 측면에서 효율적인 인증 방식이다.

1. 서 론

통신채널 상에서 요구되어지는 데이타 보안(data security) 상의 요건들이 다양해 짐에 따라, 인증 시스템의 구현은 점점 더 복잡해지고, 또한 많은 인증방식들이 제안되고 있다.[1,3,4] 통신 시스템(communication system) 상에서 송수신자간에 메세지(message)의 교환이 일어날 때, 메세지의 내용과 송신자 측에 대한 검증(verification)은 새로운 메세지를 받아 볼 때 마다 항상 허용되야 한다. 이러한 보안 서비스(security service)를 메세지 인증(message authentication)이라 부른다. 메세지 인증 서비스는 다음과 같은 것을 검증해야 한다.[1)]

(1) 정당한(authorized) 메세지 송신자에 대한 확인
(2) 메세지 변조 여부
(3) 동일한 메세지의 수신 여부
(4) 메세지 블럭(block)의 전송 순서 확인

메세지 인증에는 3가지의 주된 방법이 있다. 인증 함수(authentication function)를 이용한 메세지 인증 방식, 대칭적 암호화 방법을 기반으로 한 메세지 인증 방식, 그리고 공개키 암호화 방법을 기반으로 한 인증 방식이다.[1,2)] 이 논문에서는 인증 함수를 이용한 메세지 인증 방식을 다루기로 한다.

송신자가 수신자에게 메세지 블럭을 전송할 때

* 단국대학교 전산통계학과
** 단국대학교 전자계산학과
† 이 논문은 1992년도 교육부 지원 학술진흥재단의 자유공모과제 학술연구조성비에 의해서 연구되었음.

이제까지 발표된 인증 방식은 수신자가 메세지 블럭과 맨 마지막 블럭의 인증자(authenticator)를 받아야만, 메세지에 대한 인증을 수신자가 수행할 수 있다.[1,2,4] 이 인증 방식은 요즘과 같이 많은 정보가 이동하는 통신채널상에서 공격을 받은 이후의 필요없는 메세지 블럭을 전송해야 하는 작업을 요구할 뿐만 아니라 시스템에 과중한 재전송작업을 부담케 한다. 메세지 블럭에 대한 제 3자의 적극적인 공격을 방지하지는 못하지만, 메세지 블럭에 대한 공격을 메세지 수신자가 수신된 각각의 메세지 블럭에 대해 인증을 바로 수행할 때 한개의 메세지 블럭내에서의 어느 정도의 정보율의 하락은 예측 되지만, 전반적인 시스템의 효율성은 증대된다. 이 논문에서는 메세지 인증 방식 중 대표적인 MAC(Message Authentication Codes)와 MDC(Manipulation Detection Codes)의 운영 방식과 이 인증 방식이 가지고 있는 메세지 인증에 대한 문제점을 밝히고 그것을 보완하는 새로운 인증 방식을 제안하고 검증을 하였다.

2. Message Authentication Codes에 대한 공격

MAC(Message Aunthentication Codes)는 DES를 사용한 CBC(Cipher Block Chaining) 운영 모드를 통해 형성된다.[3] CBC는 DES를 사용한 대표적인 운영 모드로, 암호화할 때 바로 앞 블럭의 암호문 블럭을 암호화할 평문 블럭과 XOR(eXclusive-OR)하는 방법을 취한다. 먼저 CBC의 암호화와 복호화 방법을 식으로 표현하면 (2.1)과 같다.

$$\text{암호화} : Y_i = E_k\,(X_i \oplus Y_{i-1}) \qquad (2.1)$$
$$\text{복호화} : X_i = Y_{i-1} \oplus D_k\,(Y_i),\ 1 \le i \le n,\ Y_o = IV$$

여기서 X_i와 Y_i는 각각 평문과 암호문 블럭이고, k는 비밀 키(secret key)이고 IV는 초기값(initial value)이다. E_k와 D_k는 각각 암호화와 복호화 알고리즘이다.

위와 같이 CBC운용모드로 생성된 암호문 블럭의 맨 마지막 블럭 Y_n을 MAC으로 취한다. 인증자 MAC는 메세지 블럭의 마지막 블럭으로 첨가되어 메세지와 같이 전송된다. 수신자는 송신자가 보낸 메세지를 받은 후 인증자 MAC를 분리한 후 송신자와 같은 방식으로 메세지 블럭에 대한 MAC를 생성한다. 이때 생성된 MAC와 송신자가 보낸 MAC의 비교를 통해 인증 기능을 수행하게 된다.

MAC는 CBC방식을 사용한다는 측면에서 제 3자가 인증자를 계산할 수 없다는 장점을 가지고 있지만 암호화 모듈(module)을 사용하여야 하므로 구현이 복잡하다. 또한 비밀성을 요구하는 메세지를 전송할 때 MAC를 만드는 비밀키와 메세지를 암호화 할 때 사용하는 비밀키를 가지고 있어야 되므로 키의 관리가 복잡하다.[6]

이번 장에서는 메세지의 비밀성이 요구되는 MAC와 메세지 비밀성이 보장되지 않는 MAC로 나누어 제 3자의 적극적인 공격하에서의 구조적인 취약성을 검토해 본다.

2.1. 메세지의 비밀성이 보장되지 않는 MAC에 대한 공격

인증만을 요구하는 메세지는 평문에 MAC을 첨가하여 전송한다. 이 경우는 메세지의 비밀성이 보장되지 않을 때의 메세지 전송을 예를 들어 검토하기로 한다.

2-1-1. 임의의 메세지를 만들어 보내는 공격[6]

제 3자가 메세지에 대한 공격을 하기 위해서는 첫째 송신자와 수신자가 인증에 사용한 IV=0이고, 둘째 제 3자는 송신자와 수신자 간의 메세지 전송에 사용한 한쌍의 암호문과 그것에 해당하는 평문을 알고 있고, 세째 인증에 사용한 비밀키를 계속해서 사용한다고 가정한다.

제 3자가 알고 있는 암호문을 Y라 하고, 그것에 해당하는 평문을 X라고 하자(여기서 암호문과 평문의 관계는 $E_k(X)=Y$와 $D_k(Y)=X$이다). 이때 (2.2)와 같은 임의의 메세지를 제 3자가 수신자에게 보낼 때, 수신자의 인증 검사를 통과할 수 있다. (2.2)에서의 MAC는 Y이다.

$$X,\ X\oplus Y,\ X\oplus Y,\ \ldots,\ X\oplus Y,\ Y \qquad (2.2)$$

(2.2)와 같은 구조의 임의의 메세지를 수신자가 받은 후 MAC를 검사하기 위해 X, $X\oplus Y$, $X\oplus Y$, ..., $X\oplus Y$을 (2.3)에서와 같이 CBC 방식으로 암호화하여 수신측에서의 MAC를 얻을 수 있고, 이 재계산된 MAC와 송신자가 보낸 메세지의 마지막 블럭과 비교하게 된다. 이때 수신자가 계산한 MAC와 제 3자가 보낸 임의의 메세지의 마지막 블럭이 일치함을 볼 수 있다.

$$E_k(X\oplus IV)=Y, \quad E_k(X\oplus Y\oplus Y)=Y, \quad (2.3)$$
$$E_k(X\oplus Y\oplus Y)=Y, \ldots, E_k(X\oplus Y\oplus Y)=Y$$

그러므로, 제 3자의 임의의 메세지를 만들어 보내는 공격은 수신자의 MAC검사를 통과하게 된다.

2-1-2. 임의의 메세지를 첨가하는 공격

이 공격 방법은 임의의 메세지를 만들어 보내는 방법의 변형으로 초기값이 0의 값을 갖지 않더라도 성공할 수 있다. 이 공격의 조건은 첫째 제 3자가 송수신자간에 교환된 한쌍의 $E_k(X^*)=Y^*$와 $D_k(Y^*)=X^*$의 관계를 가진 암호문 Y^*와 X^*를 알고 있고, 둘째 송신자와 수신자가 계속하여 MAC 생성에 사용하는 비밀키와 초기값을 사용한다고 가정한다.

$X_1, X_2, \ldots, X_{i-1}, MAC$이 송신자가 보내는 메세지라 할 때, 제3자는 X^*, Y^*를 이용해 (2.4)와 같이 메세지를 변형시켜 보낸다.

$$X_1, X_2, \ldots, X_{i-1}, MAC\oplus X^*, X^*\oplus Y^*, \ldots, X^*\oplus Y^*, Y^* \quad (2.4)$$

(2.4)에서 MAC는 Y^*이다.

수신자는 이 메세지에서 MAC으로 사용된 Y^*를 뺀 나머지 메세지 블럭을 CBC방식으로 메세지를 (2.5)같이 암호화 한다. 그때 맨 마지막에 만들어진 암호문과 MAC을 비교하게 된다.

$$\begin{aligned} &Y_1 = E_k(X\oplus IV),\\ &Y_2 = E_k(X_2\oplus Y_1),\\ &\cdots\cdots,\\ &Y_{i-1} = E_k(X_{i-1}\oplus Y_{i-2}),\\ &Y_i = E_k(MAC\oplus X^*\oplus Y_{i-1}), \end{aligned} \quad (2.5)$$

$$\begin{aligned} &Y_{i+1} = E_k(X^*\oplus Y^*\oplus Y^*),\\ &\cdots\cdots,\\ &Y_{n-1} = E_k(X^*\oplus Y^*\oplus Y^*) \end{aligned}$$

(2.5)에서 보면 변조되지 않은 메세지의 MAC는 Y_{i-1}이므로 i번째의 블럭은 Y_{i-1}와 MAC이 상쇄됨으로써 Y^*가 된다. 이렇게 계산하면 변조된 메세지는 Y^*가 암호문으로 발생한다. 그리고 마지막 블럭 $Y_{n-1}=E_k(X^*\oplus Y^*\oplus Y^*)$을 계산하면 Y^*임으로, 수신자가 계산한 MAC과 제 3자에 의해서 변조된 메세지의 MAC은 동일하게 된다. 그러므로 임의의 메세지를 첨가하는 공격은 수신자의 인증검사를 통과하게 된다.

2.2. 메세지의 비밀성이 보장되는 메세지에 대한 공격

메세지의 비밀성을 요구하는 메세지는 송신자가 보내려는 평문이 $X=X_1, X_2, \ldots, X_{n-1}$일때, 송신자는 MAC을 생성하기 위해 메세지를 CBC방식으로 암호화하여 맨 마지막 블럭을 MAC으로 평문과 연결하여 하나의 메세지를 만들고 MAC을 첨가한 메세지에 대한 비밀을 보장하기 위해 CBC 방식으로 그림 1과 같이 암호화 한다. 이때 메세지에 대한 제 3자의 공격은 CBC 운영모드의 self-synchronizing 성질로 인하여 가능하다.[3)]

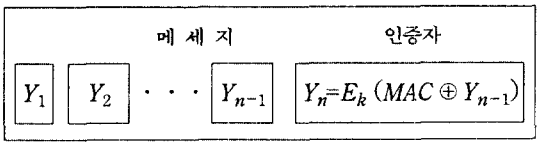


그림 1. 메세지의 비밀성이 보장되는 MAC의 메세지 구성

2-2-1. 인증과 암호화에 하나의 키를 사용할 경우의 공격.

이 공격은 첫째 인증과 암호화에 사용하는 초기값 IV이 동일하고, 둘째 인증과 암호화에 사용되어지는 비밀 키 k가 같을 경우를 가정한다. 송신자가 평문 $X=X_1, X_2, \ldots, X_n$을 IV와 k를 사용하여 CBC 모드로

암호화하여 암호문 $Y = Y_1, Y_2, ..., Y_n$을 얻어내어 암호문의 마지막 블럭을 MAC으로 한다. 그리고, MAC가 첨부된 메세지 $X' = X_1, X_2, ..., X_n, MAC$는 MAC을 생성할 때 사용한 IV와 k를 사용하여 암호화하여 암호문 $Y' = Y_1, Y_2, ..., Y_n, Y_{n+1}$을 수신자에게 보내게 된다. 이때, 제3자는 $Y_1, Y_2, ..., Y_{n-1}$을 $Y_1^*, Y_2^*, ..., Y_{n-1}^*$로 변조하여 수신자에게 보내는 공격을 한다. 이 변조된 메세지 $Y_1^*, Y_2^*, ..., Y_{n-1}^*, Y_n$를 받은 수신자는 MAC검사를 하기 위해 메세지를 복호화하면 (2.6)과 같은 변조된 암호문의 평문을 얻을 수 있다.

$$X_1^*, X_2^*, ..., X_{n-1}^*, X_n^* \qquad (2.6)$$

수신자가 MAC을 재계산하기 위해 (2.6)을 다시 암호화하면, CBC방식의 self-synchonizing 성질로 인하여 변조된 메세지와 같은 암호문 $Y_1^*, Y_2^*, ..., Y_{n-1}^*, Y_n$이 되어 수신자가 재계산한 MAC은 변조된 메세지의 MAC과 동일하게 된다. 그러므로 이 공격은 성공하게 된다.

2-2-2. 메세지 블럭의 재배열(reordering) 공격.

재배열 공격방법은 첫째 제3자가 CBC방법으로 암호화된 암호문을 알고 있고, 둘째 메세지의 인증과 암호화에 사용된 비밀키 k가 동일하다고 가정 했을 경우에 재배열 공격의 예를 든다. 보내지는 메세지는 평문 $X_1, X_2, ..., X_{n-1}$을 CBC방식으로 암호화하여 MAC을 생성한 후, 이 MAC을 포함한 평문 $X_1, X_2, ..., X_{n-1}, Y_{n-1}$을 암호화하여 암호문 $Y_1, Y_2, Y_3, ..., Y_{n-1}, Y_n$을 보내게 된다. 이때 제3자가 암호문 $Y_1, Y_2, Y_3, ..., Y_{n-1}, Y_n$의 두 블럭의 위치를 바꾼다 할지라도 수신자는 MAC을 통해 블럭들의 재배열을 검출해 낼 수 없다.

MAC을 제외한 전송되는 메세지는 $Y=Y_1, Y_2, Y_3, ..., Y_{n-1}$일때, 제3자가 Y를 (2.7)과 같이 i번째 블럭과 j번째 블럭의 위치를 재배열하여 수신자에게 보내게 된다.

$$Y^* = Y_1, Y_2, ..., Y_j, ..., Y_i, ..., Y_{n-1} \qquad (2.7)$$

단 $i < j$ 이다.

재배열된 (2.7)을 MAC의 검사를 하기 위해 복호화하면 (2.8)이 된다.

$$X^* = X_1^*, X_2^*, ..., X_{n-1}^* \qquad (2.8)$$

MAC의 검사를 하기위해 (2.8)을 CBC방식으로 암호화하게 되는데 CBC방식의 self-synchronizing성질로 인하여 (2.7)의 암호문이 생성된다. 그러므로 재배열된 메세지의 MAC도 Y_{n-1}이 되어 재배열되지 않은 블럭의 MAC과 동일하게 된다. 그러므로 제3자의 재배열된 메세지는 수신자의 MAC 검사를 통과하게 된다.

3. Manipulation Detection Codes에 대한 공격

MDC의 생성방식은 (3.1)과 같다.[4,7] 메세지가 $X_1, X_2, ..., X_{n-1}$일 때,

$$MDC = X_1 \oplus X_2 \oplus ... \oplus X_{n-1} \qquad (3.1)$$

여기서 ⊕는 XOR 연산자이다.

(3.1)에서 생성된 MDC를 메세지에 첨가한후 암호화하여 보낸다. 이 논문에서는 DES와 CBS 운영모드를 사용하여 메세지를 암호화하여 전송할 경우를 검토하는데 전송되어지는 암호문 블럭은 그림 2와 같다.

CBC 운영 모드를 사용하여 암호화한 메세지 블럭의 MDC에 대한 공격은 암호화 된 메세지에 임의의 블럭을 삽입(insertion)하는 공격과 암호문 블럭들을 재배열(Reordering)하는 공격으로 나누어 검토한다.

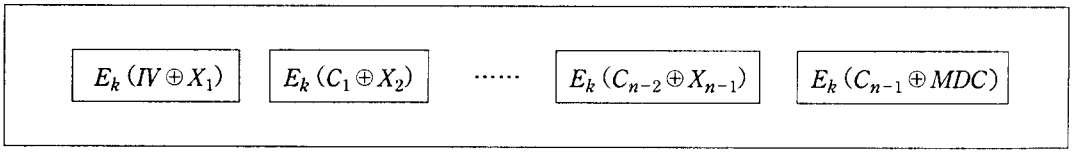


그림 2. MDC를 인증자로 사용한 메세지의 구조 (IV는 초기값이고, C_i는 암호문을 나타냄.)

3.1. 암호문 블럭에 임의의 메세지 블럭의 삽입.

CBC 모드를 사용하였을 때, 제 3자는 ciphertext-only attack[3] 방법을 사용하여 암호화 된 메세지 블럭에 가짜의 블럭을 쉽게 삽입할 수 있다.

3-1-1. Pairwise block 삽입공격

만약 제 3자가 임의의 암호문 Y'를 알고 있을 때, 송신되는 암호문 $Y=Y_1, Y_2, \ldots, Y_{n-1}, Y_n$에 대하여 임의의 메세지를 쌍(pair)으로 삽입하는 공격을 할 경우 MDC에 변화를 주지 않는다.

예를 들면, 정당한 송신자로 부터 송신된 변조되지 않은 암호문이 $Y=Y_1, Y_2, \ldots, Y_{n-1}, Y_n$이고, 이에 해당하는 평문이 $X=X_1, X_2, \ldots, X_{n-1}, X_n$라 하고, MDC는 $X_n = X_1 \oplus X_2 \oplus X_3 \oplus \ldots \oplus X_{n-2} \oplus X_{n-1}$ 이다. 이때 제 3자가 알고 있는 임의의 암호문 $Y' = E_k(X_i' \oplus Y_{i-1}')$를 마지막 블럭을 제외한 임의의 위치에 쌍(pair)으로 삽입하면 (3.2)가 된다.

$$Y=Y_1, Y_2, Y', Y_3, \ldots, Y', Y_{n-2}, Y_{n-1}, Y_n \quad (3.2)$$

(3.2)를 제 3자에 의해서 전송된 변조된 메세지를 수신자가 인증 검사를 하기 위해 복호화하면 다음과 같은 평문을 얻을 수 있다.

$$X=X_1, X_2, Y_2 \oplus D_k(Y_i'), Y_i' \oplus D_k(Y_3), \ldots, Y_{n-3} \oplus D_k(Y_i'), Y_i' \oplus D_k(Y_{n-2}), X_{n-1}, X_n$$

위 평문 X의 MDC를 구하면 다음 식으로 주어진다.

$$MDC = X_1 \oplus X_2 \oplus Y_2 \oplus D_k(Y_i') \oplus Y_i' \oplus D_k(Y_3) \oplus \ldots \oplus Y_{n-3} \oplus D_k(Y_i') \oplus Y_i' \oplus D_k(Y_2) \oplus X_{n-1}$$

여기서 Y_i'와 $D_k(Y_i')$는 XOR의 성질에 의해서 상쇄되어 제 3자에 의해 변조 됐음에도 불구하고 변조되지않은 메세지의 MDC와 동일한 MDC를 수신자는 얻게 된다. 수신자가 계산한 MDC를 정리하면 다음과 같다.

$$MDC = X_1 \oplus X_2 \oplus X_3 \oplus \ldots \oplus X_{n-2} \oplus X_{n-1} = X_n$$

그러므로, 제 3자의 Pairwise block 삽입 공격은 수신자의 인증 검사를 통과하게 된다.

3-1-2. 메세지의 값이 0(Zero)인 임의의 메세지 삽입 공격.

메세지에 $Y_i^* \oplus D_k(Y_i^*) = 0$인 블럭 Y_i^*를 삽입시켜 MDC에 대한 공격을 할 수 있다. 제 3자는 $Y_i^* \oplus D_k(Y_i^*) = 0$의 성질을 가진 Y_i^*를 i번째의 블럭에 삽입시킨다. 이것을 식으로 표현하면 다음과 같다. 여기서 MDC는 Y_n이다.

$$Y=Y_1, Y_2, \ldots, Y_i^*, \ldots, Y_{n-1}, Y_n$$

제 3자에 의해서 변조된 메세지를 수신자가 인증 검사를 하기 위해, 인증자를 제외한 메세지를 복호화하여 다음과 같은 평문을 얻을 수 있다.

$$X=X_1, X_2, \ldots, Y_{i-1} \oplus D_k(Y_i^*), Y_i^* \oplus D_k(Y_i), \ldots, X_{n-1}$$

위의 변조된 메세지의 MDC는 다음과 같이 형성된다.

$$\begin{aligned} MDC &= X_1 \oplus X_2 \oplus \ldots \oplus Y_{i-1} \oplus D_k(Y_i^*) \oplus Y_i^* \oplus D_k(Y_i) \oplus \ldots \oplus X_{n-2} \oplus X_{n-1} \\ &= X_1 \oplus X_2 \oplus \ldots \oplus X_{n-2} \oplus X_{n-1} \end{aligned}$$

위의 MDC에서 보는 것과 같이 $Y_i^* \oplus D_k(Y_i^*) = 0$ 성질로 인하여 수신자가 계산한 MDC와 제 3자에 의해서 변조된 MDC와 같게 되므로, 수신자는 인증 검사로 메세지의 변조를 알아낼 수 있다.

3.2. 메세지의 재배열에 따른 공격.

재배열 공격은 전송되는 메세지 블럭을 제 3자가 임의대로 블럭의 순서를 바꾸는 공격이다.[5] 이 공격은 CBC 운영 모드와 같이 체인 방식을 이용하는 암호화 방법에는 메세지에 많은 변화를 파급시키게 된다. 여기서 제시되는 재배열공격은 ciphertext-only attack[8]의 경우 MDC에 의해 검출되지 않는다.

3-2-1. 메세지 블럭 재배열에 대한 공격사례 1.

송신자가 수신자에게 암호화된 블럭 $Y=Y_1, Y_2, \ldots, Y_{n-1}, Y_n$를 전송했다면, 그리고 제 3자가 메세지 블럭의 순서를 재배열하는 공격을 가하여 메세지

$Y'=Y_1, Y_2, \ldots, Y_{i+1}, Y_i, Y_{i-1}, Y_n$를 수신자가 받아 보았을 때를 예를 들어본다. 여기서 Y_n은 MDC로 사용된 블럭 X_n에 해당하는 암호문이다.

수신자는 제 3자에 의해 재배열 공격을 받은 메세지 블럭 Y'에 대한 인증 검사를 하기 위해 MDC를 계산한다. 먼저 MDC를 계산하기 위해 암호문을 복호화하면 다음 식과 같다.

$$\begin{aligned} X' = {} & IV \oplus D_k(Y_1),\ Y_1 \oplus D_k(Y_2),\ \ldots, \\ & Y_{i-2} \oplus D_k(Y_{i+1}),\ Y_{i+1} \oplus D_k(Y_i), \\ & Y_i \oplus D_k(Y_{i-1}),\ Y_{i-1} \oplus D_k(Y_{i+2}), \\ & \ldots,\ Y_{n-1} \oplus D_k(Y_n) \\ = {} & X_1, X_2, \ldots, X'_{i-1}, X'_i, X'_{i+1}, \ldots, X_n \end{aligned}$$

위의 복호화된 메세지의 MDC를 구하면 다음과 같다.

$$\begin{aligned} MDC' = {} & IV \oplus D_k(Y_1) \oplus Y_1 \oplus D_k(Y_2) \oplus \ldots \oplus \\ & Y_{i-2} \oplus D_k(Y_{i+1}) \oplus Y_{i+1} \oplus D_k(Y_i) \\ & \oplus Y_i \oplus D_k(Y_{i-1}) \oplus Y_{i-1} \oplus D_k(Y_{i+2}) \\ & \oplus \ldots \oplus Y_{n-2} \oplus D_k(Y_{n-1}) \\ = {} & X_1 \oplus X_2 \oplus \ldots \oplus X_{i-1} \oplus X_i \oplus X_{i+1} \oplus \ldots \oplus X_{n-1} \end{aligned}$$

여기서, 수신자가 계산한 MDC'와 제 3자에 의해서 재배열된 메세지의 MDC인 X_n이 같게 되므로, 제 3자의 메세지에 대한 재배열 공격은 수신자의 인증 검사를 통과한다.

3-2-2. 메세지 블럭의 재배열 공격사례 2.

제 3자에 의한 재배열 공격은 여러 각도로 이루어질 수 있다. MDC블럭과 그 전의 블럭을 제외하고는 어떤 블럭에도 재배열 공격이 가해질 수 있다. 예를 들면, 송신자가 수신자에게 평문 $X=X_1, X_2, \ldots, X_n$을 암호화하여 암호문 $Y=Y_1, Y_2, \ldots, Y_{n-1}, Y_n$를 보낼 때, 공격자가 재배열 공격을 하여 $Y'=Y_{n-3}, Y_{n-4}, \ldots, Y_1, Y_{n-2}, Y_{n-1}, Y_n$로 하여 수신자에게 보냈을 경우(여기에서 Y_n는 MDC로 사용된 블럭 X_n의 암호문이다), 제 3자에 의해 재배열된 암호문 Y'는 다음과 같이 복호화 된다.

$$\begin{aligned} X' = {} & IV \oplus D_k(Y_{n-3}),\ Y_{n-3} \oplus D_k(Y_{n-4}),\ \ldots, \\ & Y_3 \oplus D_k(Y_2),\ Y_2 \oplus D_k(Y_1),\ Y_1 \oplus D_k(Y_{n-2}), \\ & Y_{n-2} \oplus D_k(Y_{n-1}),\ Y_{n-1} \oplus D_k(Y_n) \\ = {} & X_{n-3} \oplus Y_{n-4} \oplus IV,\ X_{n-4} \oplus Y_{n-3} \oplus Y_{n-5},\ \ldots, \\ & X_2 \oplus Y_3 \oplus Y_1,\ X_1 \oplus Y_2 \oplus IV,\ X_{n-2} \oplus Y_1 \oplus Y_{n-3}, \\ & X_{n-1} \oplus X_n \end{aligned}$$

X의 MDC는 $X_n=X_1 \oplus X_2 \oplus \ldots \oplus X_{n-2} \oplus X_{n-1}$임으로, 재배열된 메세지 X'로 부터 MDC를 재계산한 후 비교한다. X'의 MDC는 다음과 같이 정리할 수 있다.

$$\begin{aligned} & X_{n-3} \oplus Y_{n-4} \oplus IV \oplus X_{n-4} \oplus Y_{n-3} \oplus Y_{n-5} \oplus \ldots \oplus X_2 \oplus \\ & Y_3 \oplus Y_1 \oplus X_1 \oplus Y_2 \oplus IV \oplus X_{n-2} \oplus Y_1 \oplus Y_{n-3} \oplus X_{n-1} \oplus X_n \end{aligned}$$

위의 식을 정리하면 $X_1 \oplus X_2 \oplus \ldots \oplus X_{n-1} \oplus X_n$이 된다. 이때 $X_n=X' \oplus X_n$과 같이 등호관계가 성립하면 MDC검사는 통과한다. 여기서는 등호 관계가 성립하므로 MDC검사를 통과하게 된다. 그러므로 MDC에서 재배열 공격은 맨 마지막 블럭과 전 블럭에 대한 위치만 바꾸어 놓지 않으면 MDC검사를 통과할 수 있다.

4. 새로운 인증 방식의 제안

지금까지 살펴본 각 인증 방식은 제 3자에 대한 임의의 메세지 삽입과 재배열 공격을 검출하지 못하는 문제점을 가지고 있다. 일반적으로, 소극적인 공격(passive attack)은 발견할 수는 없지만 암호화 방법을 사용하여 쉽게 방지할 수 있는 반면에 적극적인 공격(active attack)은 쉽게 발견할 수는 있지만 방지할 수는 없다.[1,5] 이 논문에서 다루는 것은 적극적인 공격이므로 아무리 좋은 보안 서비스와 인증 서비스를 가진 통신 시스템이라도 제 3자의 공격을 방지하기는 힘들다. 그러므로 통신채널 상에서 고려되야 될 문제는 얼마나 빨리 메세지에 대한 제 3자의 공격을 검출하여 재전송을 요구하는 서비스를 제공해 주는 것이 될 것이다. 메세지의 재전송 서비스를 하기 위해서는 통신 채널상에서 발생하는 제 3자의 메세지에 대한 적극적인 공격을 빨리 대응할 수 있는 인증 시스템을 구현하는 것이 중요한 문제가 된다. 새로 제안된 알고리즘은 메세지의 마지막 블럭에 인증자를 첨가하여 보내는 다른 알고리즘과는

달리 각 메세지 블럭마다 인증자를 첨부하는 메세지를 인증하는 알고리즘이다.

4.1. 메세지 구성과 인증 방식

여기서는 메세지의 비밀성이 보장되지 않은 경우의 인증과 관련하여 제 3자의 Known-plaintext attack[8]이 취해졌을 경우를 고려한다. 메세지의 구성은 그림 3과 같다.

보내지는 전체 메세지 S는 메세지 블럭 S_i으로 이루어 지고, 메세지 블럭 S_i는 각각 부블럭(sub-block) $m_{ij}(1 \le i,j \le n)$와 인증자 H_i로 이루어진다.

이 인증 방식은 메세지를 이루는 주블럭과 부블럭을 함께 인증한다.

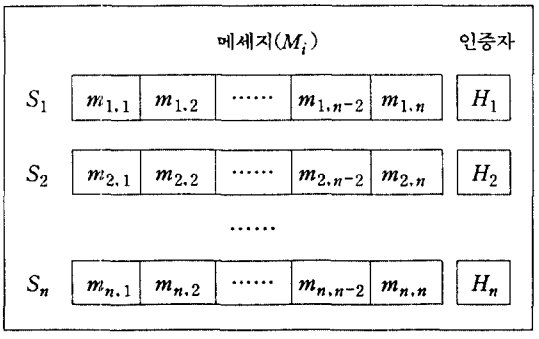


그림 3. 메세지 블럭의 구성.

메세지의 구성과 인증 방식을 수식적인 방법으로 표현을 하면 (4.1),(4.2),(4.3),(4.4)과 같다.

전체 메세지 : $S=S_1,S_2, \ldots, S_n$ (4.1)

메세지 블럭의 구조 :

$S_i=(M_i,H_i)$ (4.2)

$M_i=(m_{i,1},m_{i,2}, \ldots ,m_{i,n})$ (4.3)

인증방식 : $H_0=IV$

$H_i=E_k(H_{i-1} \oplus h(M_i))$ (4.4)

단, $1 \le i \le n$이다.

인증 방식에서 사용한 $h()$는 해쉬함수로 다음과 같은 3가지의 특성을 가진 함수를 사용한다.[4]

(1) 임의의 길이를 가진 메세지 M을 정해진 길이로 "요약(digest)"할 능력을 가져야 한다.

(2) One-way이어야 한다. 즉, 해쉬함수의 공변역에 속하는 y가 있을 때, 계산적으로 $h(M)=y$의 M을 발견하기가 불가능 해야 한다.

(3) Collision-free이어야 한다. 즉, $h(M)=h(M')$의 성질을 가진 한쌍의 메세지 M, M'을 찾는 것이 계산적으로 불가능해야 한다.

이 인증 방식에서 인증자를 암호화하는 이유는 암호화하지 않았을 경우 제 3자가 초기값 IV를 안다면 첫번째 인증자 $H_1=H_0 \oplus h(M_1)$에서 해쉬함수를 $h(M_i)=H_0 \oplus H_1$ 방식으로 유도할 수 있고, 또한 해쉬함수를 제 3자가 알고 있을 경우에는 초기값 IV를 유도해 낼 수 있다는 문제점을 해결하기 위해서이다. 그리고 이 인증자를 암호화하기 위해서는 해쉬함수의 값 h_i의 길이와 인증자 H_i는 DES로 암호화 할 수 있는 크기를 가져야 된다.

4.2. 인증 알고리즘의 운영

앞에서 제시한 인증방식을 송신측과 수신측 알고리즘으로 나누어 표현할 수 있다. 인증자 $H_i=E_k(H_{i-1} \oplus h(M_i))$는 앞 블럭의 인증자와 해쉬값을 XOR 연산을 하게 되는데, 이유는 블럭과 블럭 간의 인증자에 연관 관계를 부여함으로 앞 블럭에 제 3자의 적극적인 공격이 있을시에 블럭에서 발생한 error가 다음 블럭까지 파급되어 제 3자의 공격을 바로 검출할 수 있게 된다.

수신측 인증 알고리즘에서 인증자 H_i는 부블럭을 포함한 M_i을 해쉬함수에 의해 일정한 크기의 값을 만들어 낸 후 앞 블럭의 인증자와 XOR연산을 하여 만든다. 인증자를 만드는 송신측 알고리즘은 아래와 같다.

단계 1. $H_0=IV$
  $i=1 \ldots n$ *동안 다음 단계를 반복한다.*
  단계 2. $h_i=h(M_i)$
  단계 3. $H_i=E_k(H_{i-1} \oplus h_i)$
  단계 4. Add H_i *to* M_i
단계 5. 끝

송신자가 보낸 메세지를 받고 수신자는 인증자를 검사하기 위해 송신측 알고리즘과 같은 과정을 거치게 되고 수신측에서의 인증자를 검사하기 위한 알고리즘은 다음과 같다.

단계 1. $H_0=IV$
$i=1 \ldots n$ *동안 다음 단계를 반복한다.*
단계 2. 송신자가 보낸 i번째 블럭을 받는다.
단계 3. 메세지 블럭 뒤에 붙은 인증자 H'_i*를 분리한다.*
단계 4. 분리한 메세지 M에 대한 $h_i=h(M_i)$*를 구한다.*
단계 5. $H_i=E_k(H_{i-1}\oplus h_i)$*을 구한다.*
단계 6. 만약 $H_i=H'_i$*이면 메세지를 받는다.*
아니면 메세지 블럭에 대한 수신을 거부.
단계 7. 끝

위와 같은 알고리즘을 사용하여 수신자가 메세지를 확인하므로 수신자는 메세지에 대한 적극적인 공격이 있을 때 메세지의 수신을 거부한다. 이 인증 알고리즘의 특징은 제3자의 메세지에 대한 적극적인 공격시에 공격받은 블럭을 바로 검출할 수 있다는 장점을 가진다.

4.3. 알고리즘의 검증.

이 알고리즘은 메세지 블럭 하나 하나에 인증자를 붙이므로, 각 블럭에 대한 인증은 물론 메세지 전체 블럭에 대해서도 인증을 수행할 수 있도록 하였다. 이번 절에서는 2장, 3장에서 사용된 여러 공격방식을 적용하여 새롭게 제안된 인증 방식의 안전성을 메세지 블럭과 그 블럭을 이루는 부블럭들에 대하여 나누어 검증한다.

4-3-1. 전체 메세지 블럭에 대한 검증.

여기서는 인증만을 요구하는 메세지 전송에서의 적극적인 공격을 가정하고 인증방식을 검증하기로 한다.

(1) 블럭에 대한 재배열 공격

블럭에 대한 재배열공격은 전체의 메세지 S에 대한 공격이다. 전체 메세지 S는 몇개의 메세지 블럭 S_i로 나누어지며, 메세지 블럭 S_i은 부블럭과 인증자로 구분된다. 이때 공격자가 이 메세지 블럭에 대하여 재배열 공격을 감행할 때, 메세지에 에러 파급(error propagation)이 발생하게 되어 메세지 블럭에 대한 공격이 시도된 블럭 이후에는 원래의 인증자의 값과 다른 값이 발생하게 된다. 전체 메세지 S가 $S_1, S_2, \ldots, S_n$으로 이루어져 있을 때 공격자가 메세지 블럭 S의 $i-1$번째 블럭과 $i+1$블럭의 순서를 바꾸는 재배열 공격을 하여 메세지 블럭 $S'=S_1, S_2, \ldots, S_{i+1}, S_i, S_{i-1}, \ldots, S_n$를 수신자에게 보냈을 경우, 수신자는 수신측 인증 알고리즘에 의해서 메세지 블럭에 대한 인증 작업을 바로 수행한다. 이때 제3자의 공격에 의해서 순서가 바뀐 $i-1$번째 블럭에 대한 인증 검사를 하게 된다. 재배열 공격에 의해서 $i-1$번째 블럭으로 순서가 바뀐 S_{i+1}에서 송신자가 수신자에게 보낸 M_{i+1} 인증자는 $H_{i+1}=E_k(H_i\oplus h_i(M_{i+1}))$이나, 수신자가 제3자의 재배열공격을 받은 메세지 블럭의 M_{i+1}를 계산한 인증자는 $H_{i+1}=E_k(H_{i-2}\oplus h(M_{i+1}))$ 이 된다. 그러므로, 수신자는 $i-1$번째 메세지 블럭에서 제3자의 공격이 있었다는 것을 검출할 수 있다.

(2) 블럭에 대한 삭제 공격

블럭에 대한 삭제 공격은 MDC나 MAC에서도 검출할 수 있으나 일반적인 공격 형태임으로 다루기로 한다. 메세지의 삭제는 인증자에 변화를 줄 수 있으므로 인증자를 검사하는데 쉽게 검출할 수 있다. 이 논문에서 제시한 새로운 인증방식에서 강조하는 것과 같이 메세지의 삭제가 이루어졌을 경우 삭제된 블럭 이후의 인증자는 수신자가 재계산한 인증자의 값과 다르게 나타나게 됨으로 메세지의 수신을 즉시 거부할 수 있다.

송신자가 수신자에게 전체 메세지 $S=S_1, S_2, \ldots, S_n$를 보낼때, 통신채널상에서 제3자는 이것을 중

간에 가로채어 메세지의 i번째 블럭을 삭제한 후, 수신자에게 메세지 $S'=S_1, S_2, \ldots, S_{i-1}, S_{i+1}, \ldots, S_n$를 보냈을 경우를 예를 든다. 이때 수신자는 S'의 i번째 블럭의 S_{i+1}의 인증자가 틀리다는 것을 발견할 수 있다. 즉, S'의 i번째 블럭 S_{i+1}의 인증자는 $E_k(H_i \oplus h(M_{i+1}))$이나 수신자가 계산한 인증자는 $E_k(H_{i-1} \oplus h(M_{i+1}))$이 된다. 그러므로 삭제된 블럭의 위치를 검출하는 것과 동시에 메세지에 제3자의 공격이 있었다는 것을 알 수 있다.

(3) 블럭에 대한 메세지 삽입

메세지 블럭의 삽입공격은 보내지는 메세지에 임의의 메세지를 공격자가 삽입하거나, 새로운 메세지를 만들어 삽입하는 방법이 있다. 이 절에서는 임의의 메세지의 삽입공격을 보기로 한다. 제3자가 송신자와 수신자 간의 통신에 사용한 임의의 메세지를 알고 있을때 다음 메세지는 제3자가 메세지 S에 임의의 블럭 X를 i번째에 삽입 할 경우이다.

$$S_1=(M_1, E_k(H_0 \oplus h_1))$$
$$S_2=(M_2, E_k(H_1 \oplus h_2))$$
$$\vdots$$
$$X=(X_i, A)$$
$$\vdots$$
$$S_n=(M_n, E_k(H_{n-1} \oplus h_n))$$

여기서, A는 X의 인증자이다.

삽입공격시 제3자가 인증에 사용한 해쉬함수를 알고 있다고 가정하더라도 인증자를 암호화할 때 사용한 비밀키를 알지 못하므로 메세지 X의 인증자를 수신자의 인증 검사를 통과할 수 있도록 만들수 없다. 그러므로, 수신자의 인증 검사에서 삽입된 블럭 X가 잘못된 블럭이라는 것을 쉽게 검출할 수 있다.

4-3-2. 메세지 블럭을 이루는 부블럭에 대한 공격.

메세지 블럭 S_i는 부블럭과 인증자로 구성되어있다. 이 경우 제3자의 부블럭에 대한 공격도 고려되어야 한다. 각 블럭은 다음과 같은 형태로 부블럭과 인증자를 포함하고 있다.

$$S_i=(M_i, E_k(H_{i-1} \oplus h(M_i)))$$

$$M_i=m_{i,1}, m_{i,2}, m_{i,n}$$

여기서, 해쉬함수에 입력으로 사용되는 것은 M_i를 이루는 부블럭이다. 이 절에서는 제3자가 부블럭에 적극적인 공격을 가했을 경우를 예를 든다.

(1) 재배열공격에 대한 검증.

재배열공격은 메세지 블럭 M_i에 대하여 각 부블럭의 위치를 변경하는 공격이다. 즉 메세지 블럭 S_i가 전송될 때, M_i에 대한 재배열 공격을 다음과 같이 가할 수 있다.

공격을 받지 않은 메세지 블럭 :

$$M_i=m_{i,1}, m_{i,2}, \ldots, m_{i,j}, \ldots, m_{i,k}, \ldots, m_{i,n}$$

재배열 공격을 받은 블럭 :

$$M_i'=m_{i,1}, m_{i,2}, \ldots, m_{i,k}, \ldots, m_{i,j}, \ldots, m_{i,n}$$

이때, M_i과 M_i'의 인증자가 같다면 이 재배열 공격은 성공하게 된다. 그러나 M_i의 인증자 $E_k(H_{i-1} \oplus h(M_i))$에서 해쉬함수의 값은 $h(m_{i,1}, m_{i,2}, \ldots, m_{i,j}, \ldots, m_{i,k}, \ldots, m_{i,n})$이고, 재배열된 M_i'의 해쉬함수 값은 $h(m_{n,1}, m_{n,2}, \ldots, m_{n,k}, \ldots, m_{n,j}, \ldots, m_n)$이므로 M_i과 M_i'의 해쉬값이 다르게 된다. 그러므로 부블럭의 재배열 공격은 해쉬함수 값을 다르게 하므로 인증자에 영향을 주어 인증 검사를 통과하지 못하고 수신자에게 검출된다. 그리고, M_i과 M_i'의 해쉬값이 같다면 4.1절에서 해쉬함수의 3번째 특성을 볼 때 "$h(M)=h(M')$의 성질을 가진 두개의 메세지 M_i, M_i'을 찾는 것은 계산적으로 불가능해야 한다."라는 항목에 어긋나는 성질을 가진 해쉬함수가 된다. 그러므로 이 성질을 만족하는 해쉬함수라면 부블럭에 대한 재배열 공격은 검출되게 된다.

(2) 부블럭에 메세지 삽입과 삭제에 대한 검증.

부블럭의 삽입공격은 제3자가 새로운 부블럭을 만들어 메세지 블럭 M_i에 삽입시키는 공격을 말한다. 이 공격은 해쉬함수의 값에 변화를 주기 때문에 수신자의 인증 검사시 검출된다. 또한 삭제 공격을 받은 부블럭은 받지않은 메세지 블럭 M_i의 해쉬함수 값 $h(m_{i,1}, m_{i,2}, \ldots, m_{i,i}, \ldots, m_{i,j}, \ldots, m_{i,n})$과 다른 해쉬함수 값을 갖는다. 물론 삽입 공격에서도 공격을

받은 후의 해쉬함수 값과 받지않은 해쉬함수 값은 다르다. 그러므로 이 해쉬함수의 값의 변화는 인증자 전체의 값에 변화를 주어 원래의 인증자와 다른 인증자 값을 가지게 된다. 그러므로 부블럭에 대한 삽입, 삭제 공격은 바로 검출할 수 있다.

5. 결 론

이 논문에서는 제3자의 적극적인 공격하에서의 MAC와 MDC에 대한 구조적 취약점을 분석하였다. 그리고 이를 보완하는 새로운 인증 방식을 제안하고 검증하였다. 새로 제안된 인증 방식은 메세지를 분할하여 인증자를 붙여준다. 이것은 MAC와 MDC에서와 같이 인증자를 맨 마지막 블럭에 붙일 경우 메세지 전체를 수신 받아야만 인증검사를 할 수 있는 단점을 해결하였다. 또한, 메세지에 대한 공격이 있었던 블럭을 바로 검출할 수 있다는 장점을 가지고 있기 때문에 요즘과 같이 통신채널을 통하여 많은 정보가 이동하는 통신 시스템에 교통량을 줄일 수 있다는 점에서 새로운 인증 방식의 응용은 통신 시스템 상에서 이용될 수 있는 이점을 가진다.

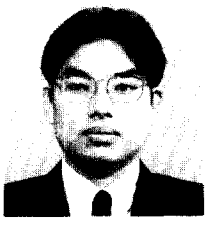
참 고 문 헌

1. Sead Muftic, *Security Mechanisms for Computer Networks*, pp 122-131, 1989 Ellis Horwood Limited.
2. Jennifer Seberry and Josef Pieprzyk, *Cryptography : An Introduction to Computer Security*, pp 131-180, Prentice Hall 1989
3. D.W. Davies and W.L. Price, *Security for Computer Networks*, pp 85-87, pp 161-131, JOHN WILEY & SONS.
4. Gustavus J. Simmons, *Contemporary Cryptology : The Science of Information Integrity*, pp 345, pp 552-554, IEEE Press 1991.
5. Victor L. Vdydock and Stephen T. Kent, "Security Mechanism in High-level Network Protocols," *Computing Surveys*, Vol. 15, No. 2, Jung 1983.
6. R.R. Jueneman, S.M. Matyas, and C.H. Meyer, "Message Authentication with Manipulation Detection Codes," Proc. of the 1983 Symposium on Security and Privacy, pp. 33-54.
7. Robert R. Jueneman, "Electronic Document Authentication," *IEEE Network*, Vol. 1, No. 2, Magazine. April 1987.
8. Dorothy Elizabeth Robling Denning, *Cryptology and Data Security*, pp 2-3, Addison Wesley.

□ 著者紹介

박 창 섭

1958년생
연세대학교 경제학과 졸업
미국 Lehigh 전자계산학과 석사
미국 Lehigh 전자계산학과 박사
현재 단국대학교 전자계산학과 조교수
관심분야 : 암호이론 및 부호이론



안 효 범

1968년생
단국대학교 전자계산학과 졸업
단국대학교 전산통계학과 대학원
관심분야 : 암호이론 및 부호이론, Operation system Security