情報保護學會論文誌
第16卷 第1號, 2006. 2

다중 운영 모드에 대한 패딩 오라클 공격

이태건,[1†] 김종성,[1] 이창훈,[1] 성재철,[2‡] 이상진[1]
[1]고려대학교 정보보호기술연구센터, [2]서울시립대학교 수학과

Padding Oracle Attacks on Multiple Modes of Operation

Taekeon Lee,[1†] Jongsung Kim,[1] Changhoon Lee,[1] Jaechul Sung,[2‡] Sangjin Lee[1]
[1]Center for Information Security Technologies(CIST), Korea University
[2]Department of Mathematics, University of Seoul

요 약

일반적으로 블록암호를 이용한 암호화는 블록암호의 입력에 맞추기 위해 메시지의 마지막에 적당한 값을 패딩한다. 만약 공격자에게 평문의 패딩이 옳은지 아닌지의 여부를 판단하는 오라클이 있다면 임의의 암호문에 대응하는 평문을 찾는 공격을 패딩 오라클 공격이라 한다. 본 논문에서는 다양한 패딩 방법을 사용하는 블록암호의 이중 모드와 삼중 모드에 대해 이 공격을 수행한다. 이러한 안전성 분석 결과, 공격자가 패딩 오라클을 사용할 수 있다면 36개의 이중 모드들 중 12개, 216개의 삼중 모드들 중 22개가 패딩 오라클 공격에 취약함을 알 수 있었다. 이는 이중 모드와 삼중 모드들은 단일 모드만큼의 안전성 밖에 제시하지 못한다는 사실을 나타낸다.

ABSTRACT

This attack requires an oracle which on receipt of a ciphertext, decrypts it and replies to the sender whether the padding is VALID or INVALID. In this paper we extend these attacks to other kinds of modes of operation for block ciphers. Specifically, we apply the padding oracle attacks to multiple modes of operation with various padding schemes. As a results of this paper, 12 out of total 36 double modes and 22 out of total 216 triple modes are vulnerable to the padding oracle attacks. It means that the 12 double modes and the 22 triple modes exposed to these types of attacks do not offer the better security than single modes.

Keywords : *Block Ciphers, Modes of Operations, Padding Oracle Attacks*

Ⅰ. 서 론

2002년 Vaudenay[10]는 다양한 패딩 방법을 사용하는 CBC 모드에 대한 패딩 오라클 공격을 소개하였다. 일반적으로 블록 암호를 이용하여 메시지를 암호화할 경우, 블록암호의 입력 크기에 맞추기 위해 메시지의 마지막에 적당한 값을 패딩한다. 만약 공격자에게 평문의 패딩이 옳은지 아닌지의 여부를 판단하는 오라클을 이용하여 임의의 암호문에 대응하는 평문을 찾는 공격을 패딩 오라클 공격이라 한다. 오라클은 공격자가 질문한 암호문을 복호화하여 얻은 평문의 패딩이 옳은지 아닌지를 판단하여 *VALID* 또는 *INVALID* 값을 공격자에게 대답한다. 그리고 공격자는 다중 운영 모드로 암호화된 암호문(초기치 *IV*포함)을 중간에서 획득할 수 있고,

접수일 : 2005년 10월 14일 ; 채택일 : 2006년 1월 3일
† 주저자 : imml97@cist.korea.ac.kr
‡ 교신저자 : jcsung@uos.ac.kr

위에서 언급한 오라클에게 질의하여 응답을 얻을 수 있다고 가정한다. 이 공격자의 목적은 오라클에 대한 여러 번의 질의, 응답을 이용하여 중간에 가로챈 임의의 암호문에 대응하는 평문을 찾는 것이다.

패딩 오라클을 이용하여 운영 모드를 공격하는 방법은 2002년 Vaudenay[10]에 의해 처음 소개되었고 이 공격은 Black, Urtubia[4]에 의해 다양한 패딩 방법을 사용하는 운영모드들에 대해서 일반화되었다. 그 후에 Paterson, Yau[9]와 Klima, Rosa[8]등에 의해 패딩 오라클 공격에 대한 많은 연구가 이루어졌다.

Biham[2]은 여러 단일 모드 (즉, ECB, CBC, OFB, CFB, CBC^{-1}, CFB^{-1})들을 연결한 다중 모드들을 소개하였고, Wagner[11]는 이 모드들을 선택 초기치 및 선택 평문/선택 암호문 공격 환경 하에서 분석하였다. 그러나 이 분석 방법은 비현실적이고 비실용적이다. 그래서 홍득조[5]등은 가정을 완화시킨 기지 초기치 선택 평문/선택 암호문 환경 하에서 모든 삼중 모드들을 분석하였다.

본 논문에서는 패딩 오라클을 이용하여 다양한 패딩 방법을 사용하는 모든 이중 모드와 삼중 모드들을 공격한다. 안전성 분석 결과로 $36(=6^2)$개의 이중 모드들 중 12개, $216(=6^3)$개의 삼중 모드들 중 22개가 패딩 오라클 공격에 취약함을 알 수 있었다.

본 논문의 구성은 다음과 같다. 2장에서는 본 논문에서 사용되는 표기법, 모드에 사용되는 패딩 방법과 다중 모드에 대해 설명한다. 그리고 3장에서는 이중 모드와 삼중 모드에 대한 패딩 오라클 공격을 소개하고 마지막으로 4장에서는 본 논문의 결과를 요약한다.

II. 표기와 다중 운영 모드

본 절에서는 이 논문에 사용되는 표기법들과 운영 모드에 사용되는 패딩 방법 그리고 다중 운영 모드에 대해 소개한다.

2.1 기호 및 표기

다음은 본 논문에서 사용되는 기호 및 표기이다.

- C : 암호문 데이터
- IV : 초기값
- C_i : i 번째 암호문 블록
- C_i^j : i 번째 암호문 블록의 j번째 바이트
- n : 평문 한 블록의 바이트 수
- P : 패딩된 평문 데이터
- M : 평문 데이터
- L_M : 평문 데이터 M의 비트 길이
- e_j : j번째는 1이고 나머지는 0인 $8n$ 비트 이진수
- $X \| Y$: X와 Y의 연결 연산
- $X \oplus Y$: X와 Y의 XOR 연산
- O : 복호화된 평문이 옳은 패딩인지 아닌지를 판별할 수 있는 오라클
- *VALID* 또는 *INVALID* : 공격자의 질의에 대한 오라클의 대답으로서 임의의 암호문에 대한 복호문, 즉 평문이 가능한 값인지, 불가능한 값인지를 나타냄

2.2 운영 모드에 사용되는 패딩 방법

다음의 패딩 방법은 운영 모드에 사용되는 여러 가지 패딩 방법들이다. 본 논문에서는 이중 모드와 삼중 모드에서 이 패딩 방법들을 사용했을 때의 안전성을 살펴본다. 평문 데이터의 크기가 $(8n \times q + m)$ 비트 $(0 \le m \le 8n,\ 0 \le q)$라 하자. 운영 모드에 사용되는 패딩 방법은 다음과 같다.

가) CBC-PAD

만약 m이 0이 아닐 경우, 그리고 패딩할 바이트가 1 바이트면 01x를, 2 바이트면 0202x를 패딩한다. 이런 식으로 패딩해야 할 바이트 수만큼 16진수를 패딩한다. 만약 m이 0이면, 한 블록을 $nnn\cdots n_x$으로 패딩한다. 예를 들어 n=16일 경우 한 블록을 $1010\cdots10_x$으로 패딩한다.

나) ESP-PAD

만약 m이 0이 아닐 경우, 그리고 패딩할 바이트가 1 바이트면 01_x을, 2 바이트면 0102_x를 패딩한다. 이런 식으로 패딩해야 할 바이트 수만큼 16진수를 패딩한다. 만약 m이 0이면, 한 블록을 $01020 3\cdots n_x$으로 패딩한다.

다) XY-PAD

이 패딩 방법은 두개의 서로 다른 바이트 상수 값 X, Y를 사용한다. 평문 데이터 M 뒤에 X를 1 바이트 패딩하고 나머지 바이트에는 필요한 만큼 Y값을 패딩한다. 만약 m이 0일 경우 한 블록을 X Y…Y로 패딩한다.

라) ISO(9797-1)-PAD 3

만약 마지막 블록에 t(>0)비트가 필요하면 필요한 만큼 t비트 스트링 0^t을 패딩한다. 그리고 첫 블록에는 평문 데이터 M의 길이 L_M를 이진수로 표현하여 함께 패딩한다. 즉, 패딩을 포함은 평문 데이터는 $P=(L_M)_2 \| M \| 0^t$이 된다. 만약 m이 0일 경우, 평문 데이터는 $P=(L_M)_2 \| M$이 된다.

마) ISO(10118-1)-PAD 3

마지막 블록 끝 64 비트에 평문 데이터 M의 길이 $(L_M)_2$를 패딩하고 메시지와 데이터 길이 사이에 필요한 만큼 $10\cdots0_2$을 패딩한다. 즉, 패딩을 포함한 평문 데이터는 $P=M \| 10\cdots 0_2 \| (L_M)_2$이 된다.

2.3 다중 모드

여러 개의 암호 상자들 사이의 중간 값에 내부 피드백을 연결시켜 만든 운영 모드를 다중 운영 모드라 한다. 즉, 여러 개의 단일 모드들 (ECB, CBC, OFB, CFB, CBC^{-1}, CFB^{-1})을 연결하여 만들어지며, 이전 블록의 피드백이 다음 블록의 내부에 작용하도록 만든다. 그림 1과 그림 2는 각각 이중 모드와 삼중 모드의 한 예이다.

Ⅲ. 이중 모드와 삼중 모드에 대한 패딩 오라클 공격

본 절에서는 36개 이중 모드와 216개 삼중 모드 모두에 대해 패딩 오라클 공격을 수행한다. 그 결과로서, 이 공격에 취약한 12개의 이중 모드들과 22개의 삼중 모드들을 제시한다.

3.1 패딩 오라클 공격의 개념

Vaudenay[10]는 다양한 패딩 방법을 사용하는 CBC 모드에 대한 패딩 오라클 공격을 소개하였다. 이 공격에서 사용하는 오라클 O는 공격자가 질문한 암호문을 복호화 하여 얻은 평문의 패딩이 옳은지 아닌지를 판단하여 *VALID* 또는 *INVALID* 값을 대답해준다. 공격자는 운영 모드로 암호화된 암호문을 중간에서 가로챌 수 있고 위에서 언급한 오라클에게 질의에 대한 응답을 얻을 수 있다고 가정한다. 이 공격자의 목적은 오라클에 대한 여러 번의 질의, 응답을 이용하여 중간에 가로챈 임의의 암호문에 대한 평문을 찾는 것이다.

3.2 이중 모드에 대한 패딩 오라클 공격

본 소절에서는 패딩 오라클 공격 모델에서 제시된 패딩 오라클을 사용하여 주어진 암호문에 대응하는 평문을 찾는다. 여기서는 편의상 다양한 패딩 방법을 사용하는 이중 모드 CBC|CBC 운영 모드에 대한 패딩 오라클 공격만을 제시한다. 공격은 두 단계로 나뉘는데 첫 번째 단계에서는 오라클을 이용하여 패딩 길이를 구하고, 두 번째 단계에서는 암호문에 대응하는 평문을 구하게 된다.

우선 공격자에게 패딩 오라클 O와 암호문 $C=(IV_1, IV_2, C_1, C_2, \cdots, C_q)$가 주어졌다고 가정한다.

가) CBC-PAD

· *패딩 길이 구하기*

공격자는 평문 메시지 P의 패딩 길이를 구하기 위해 다음과 같이 이진 검색(binary search)을 시행한다.

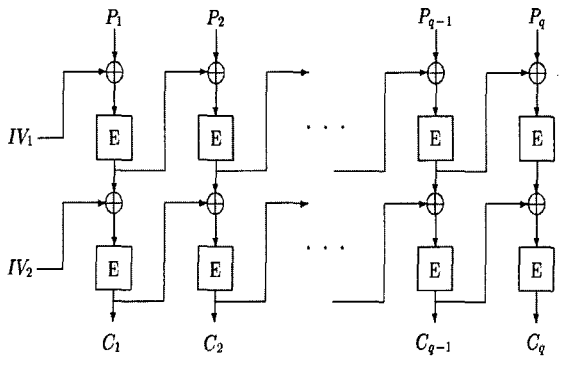

그림 1. 이중 모드의 예 : CBC|CBC

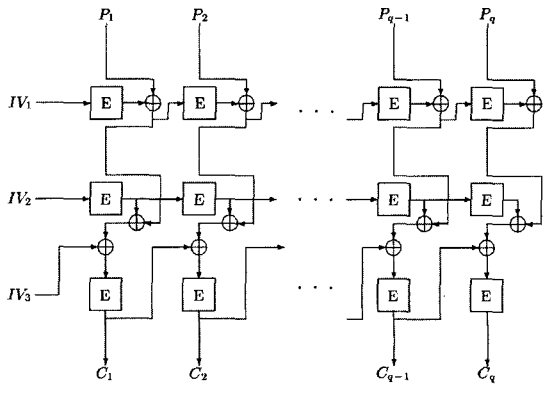

그림 2. 삼중 모드의 예 : CFB|OFB|CBC

평문의 마지막 블록(메시지 부분과 패딩 부분이 함께 있는 블록)에서, 공격자가 만약 패딩 바이트 부분의 임의의 1 바이트에 해당하는 암호문 블록의 바이트를 변화시켜 오라클 O에게 질문을 던졌다면, 오라클 O는 항상 *INVALID* 값을 대답할 것이다. 반대로, 메시지 바이트 부분의 임의의 1 바이트에 해당하는 암호문 블록의 바이트를 변화시켜 오라클 O에게 질문을 던졌다면, 오라클 O는 항상 *VALID* 값을 내놓게 될 것이다. 따라서 한 번에 1 바이트를 변형하여, 반복적으로 이진 검색을 수행하면 평문 메시지의 패딩 길이를 구할 수 있다.

먼저 암호문 블록 C_{q-2}의 중간 바이트를 변형하여 오라클에게 질문하면 오라클은 *VALID* 또는 *INVALID* 값을 대답할 것이다. 만약 오라클이 응답한 값이 *VALID*이면 공격자는 오른쪽 $n/4$번째 바이트를 변형하여 오라클에게 질문하고, 그렇지 않으면, 왼쪽 $n/4$번째 바이트를 변형하여 질문한다.

이런 식으로 위 과정을 반복하게 되면 $\log_2(n)$번의 오라클 질문으로 패딩의 길이를 알 수 있다. 여기서 n은 한 블록의 바이트 수이다.

다음의 표 1은 알고리즘은 평문 메시지 P의 패딩 길이를 구하는 이진 검색 알고리즘이다. 이 알고리즘에서 C^m_{q-2}은 암호문의 $q-2$번째 블록의 m번째 바이트가 변형되었음을 의미한다.

· *평문 구하기*

본 공격은 위에서 구한 패딩 길이를 이용한다. 공격자는 이미 패딩 길이를 알고 있으므로 패딩이 01_x, 0202_x, 030303_x, … 중에 한 가지로 패딩되었음을 알고 암호문 C_{q-2}를 패딩된 바이트보다 1만큼 큰 값으로 조작할 수 있다. 만약 마지막 블록의 2 바이트가 패딩된 경우라면, 공격자는 P_q값의 P_q^{14}, P_q^{15}값이 0202_x에서 0303_x값이 되도록 C_{q-2}^{14}, C_{q-2}^{15}값을 조작한다. 그 다음 공격자는 $q-2$번째 암호문 블록의 14번째 바이트인 C_{q-2}^{13}을 모든 t에 대해, $t \oplus C_{q-2}^{13}\ (0 \le t \le 2^8-1)$값으로 바꾸어, 바뀐 암호문을 패딩 오라클에게 질문한다. 이때, 오라클이 VALID 값을 응답했을 경우, 공격자는 그 때의 t값을 사용하여 평문 $P_q^{13}(=t \oplus 03_x)$값을 구한다.

따라서, 위와 같은 방법을 반복 사용하여 계속해서 1 바이트씩 평문을 찾아 나간다. 그럼으로써 공격자는 평문 전체를 알아 낼 수 있으며, 맨 처음의 블록 P_1값 역시 앞의 암호문 대신 초기벡터 IV값을 조작·변형함으로써 알아낼 수 있다. 공격자가 한 블록의 평문을 알아내기 위해서는 평균적으로 2^7번의 질의·응답을 n번 시행해야 하며, q개의 평문 블록을 모두 알아내기 위해서는 총 $2^7 \times n \times q + \log_2(n)$의 시간 복잡도가 필요하다.

나) ESP-PAD, XY-PAD

ESP-PAD, XY-PAD 패딩 방법의 공격방법 역시 위에서 설명한 CBC-PAD의 공격방법과 같은 방법으로 암호문 C에 대응하는 평문 P를 구할 수 있으므로, 자세한 공격 과정은 생략한다. 공격 결과는 표 3에 잘 나타나 있다.

다) ISO(9797-1)-PAD 3

· *패딩 길이 구하기*

암호문 블록이 4 이상인 경우에는 위 소절과 유사한 공격 방법으로 암호문 C에 대응하는 평문 P를 구할 수 있다. 그러나 암호문 블록이 4 보다 작

표 1. 이진 검색(binary search) 알고리즘

```
Length(x,y)
IF x = y Then Return x
m ← ⌈ (x+y)/2 ⌉
IF O(…, C^m_{q-2}, C_{q-1}, C_q) = INVALID
Then
  Length(x,m)
Else
  Length(m-1,y)
```

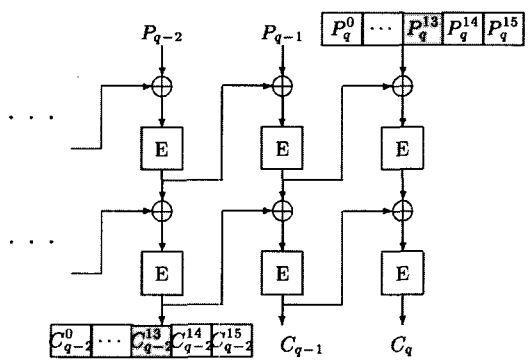

그림 3. 패딩 오라클 공격 : CBC|CBC

표 2. 평문 구하는 알고리즘

Input : $L_M, IV_1, IV_2, C_1, C_2, \cdots, C_q$
Output : $P_i^1, P_i^2, \cdots, P_i^{127}$
$-P_i$의 최하위 127비트, $1 \le i \le q$
여기서 P_i^j는 i번째 평문블록 P_i의 j번째 비트를 의미한다.

IF $3 \le i \le q$
for $j = 127$ *to* 1 *do*
$IV_1' = IV_1 \oplus L_M \oplus (256 + j)$
$C' = (IV_1', IV_2, C_1, C_{i-2}, C_{i-1}, C_i)$
$b = \Omega(C')$
$\begin{cases} \Omega(C') \doteq 0, \text{if } O(C') = VALID \\ \Omega(C') \doteq 1, \text{if } O(C') = INVALID \end{cases}$
$P_i^j = b$
IF $(b=1)$
$C_{i-2} = C_{i-2} \oplus e_j$
($i=3$인 경우, 두번째 블록에 C_1을 대신한다.)
end for

return $P_i^1, P_i^2, \cdots, P_i^{127}$

IF $i=2$,
위 알고리즘에서
$C' = (IV_1', IV_2, C_1, C_{i-2}, C_{i-1}, C_i)$대신에
$C'' = (IV_1', IV_2, C_1, IV_2, C_1, C_2)$를 사용한다.

은 경우에는 블록을 추가하는 방법을 사용하여 메시지의 패딩 길이를 구할 수 있다.

공격의 편의상 한 블록을 128 비트로 가정하자. 만약 주어진 암호문이 3 블록이면, 즉 $C=(IV_1, IV_2, C_1, C_2, C_3)$이면, 첫 번째 블록의 메시지 길이를 변형하여 한 블록을 추가한다. 그러므로 오라클 O에게 질문하는 암호문은 $C' = (IV_1 \oplus e_7, IV_2, C_1, C_1, C_2, C_3)$이 된다. 이때 오라클의 대답이 *VALID*이면 메시지 길이 L_M는 256이다. 왜냐하면 메시지 길이는 $128 < L_M \le 256$이기 때문이다. 만약 오라클의 대답이 *INVALID*이면 C' 대신에 $C'' = (IV_1 \oplus e_{8,7}, IV_2, C_1, C_1, C_2, C_3)$를 질문한다. 이 경우는 4블록 이상인 경우와 같은 방법으로 두 번째 블록의 C_1을 변형함으로써 $\log_2(128)$번의 오라클 질문으로 평문 메시지의 패딩 길이를 구할 수 있다. 두 블록인 암호문에 대해서도 이와 유사한 방법으로 패딩 길이를 구할 수 있다.

· *평문 구하기*

위 과정을 통해 얻은 평문 메시지의 패딩 길이 정보와 표 2의 알고리즘을 이용하여 주어진 암호문에 해당하는 평문을 구한다.

이 패딩 방법은 평문 메시지가 128 비트의 배수이면 메시지 뒷부분에 패딩을 하지 않는다. 그러므로 위 알고리즘을 통해 평문 각 블록의 최상위 1 비트를 제외한 모든 평문을 구할 수 있다. 그리고 남아 있는 각 블록의 최상의 비트들 $q-1$개는 전수조사를 통해 모든 평문을 구할 수 있다. 이 공격에 필요한 오라클 질문의 평균수는 $128 \times (q-1) + \log_2(128)$이지만, 일반적으로는 $8n \times (q-1) + \log_2(8n)$의 오라클 질문이 요구된다.

표 3. 이중 모드에 대한 공격 복잡도

패딩방법	CBC-PAD	ESP-PAD	XY-PAD
오라클 질문 수	$128nq + \log_2(n)$	$128nq + \log_2(n)$	$128nq + \log_2(n)$
패딩 오라클 공격이 가능한 이중모드	CBC|CBC, CBC|OFB, OFB|CBC, OFB|OFB, OFB|CFB, CFB|OFB, CFB|CFB, OFB|CFB-1, CFB|CFB-1, CFB-1|OFB, CFB-1|CFB, CFB-1|CFB-1		
패딩방법	ISO(9797-1) -PAD 3	ISO(10118-1) -PAD 3	
오라클 질문 수	$8n \times (q-1) + \log_2(8n)$	$8n \times (q-1) + \log_2(8n)$	
패딩 오라클 공격이 가능한 이중모드	CBC|CBC, CBC|OFB, OFB|CBC, OFB|OFB		
※ q : 암호문/평문 블록의 수 ※ n : 한 블록의 바이트 수			

라) ISO(10118-1)-PAD 3

앞 소절의 ISO(9797-1)-PAD 3 공격 방법과 유사한 방법으로 이 패딩 방법에 대한 패딩 오라클 공격을 할 수 있다. 그러므로 자세한 공격 과정은 생략한다. 공격 결과는 표 3에 나타나 있다.

지금까지, 가능한 모든 이중 모드에 대해 패딩 오라클 공격을 수행하였다. 그 결과 모든 36개의 이중 모드들 중에 12개의 모드가 패딩 오라클 공격에 취약함을 알 수 있었다.

3.3 삼중 모드에 대한 패딩 오라클 공격

다양한 패딩 방법을 함께 사용하는 삼중 모드들에 대한 패딩 오라클 공격은 앞 절에서 소개한 이중 모드에 대한 패딩 오라클 공격과 유사한 방법으로 공격을 수행할 수 있다.

표 4. 삼중 모드에 대한 공격 복잡도

패딩방법	CBC-PAD	ESP-PAD	XY-PAD
오라클 질문 수	$128nq$ $+\log_2(n)$	$128nq$ $+\log_2(n)$	$128nq$ $+\log_2(n)$
패딩 오라클 공격이 가능한 삼중모드	CBC|CBC|CBC, CBC|CBC|OFB, CBC|OFB|CBC, CBC|OFB|OFB, OFB|CBC|CBC, OFB|CBC|OFB, OFB|OFB|CBC, OFB|OFB|OFB, OFB|OFB|CFB, OFB|OFB|CFB-1, OFB|CFB|CBC, OFB|CFB|OFB, OFB|CFB|CFB, OFB|CFB|CFB-1, CFB|OFB|CBC, CFB|OFB|OFB, CFB|OFB|CFB, CFB|OFB|OFB-1, CFB|CFB|CBC, CFB|CFB|OFB, CFB|CFB|CFB, CFB|CFB|CFB-1		
패딩방법	ISO(9797-1) -PAD 3	ISO(10118-1) -PAD 3	
오라클 질문 수	$8n\times(q-1)$ $+\log_2(8n)$	$8n\times(q-1)$ $+\log_2(8n)$	
패딩 오라클 공격이 가능한 삼중모드	CBC|CBC|CBC, CBC|CBC|OFB, CBC|OFB|CBC, CBC|OFB|OFB, OFB|CBC|CBC, OFB|CBC|OFB, OFB|OFB|CBC, OFB|OFB|OFB		

삼중 모드들 전부에 대해 패딩 오라클 공격을 수행했을 때 216개의 삼중 모드들 중에 22개의 모드가 이 공격에 취약함을 알 수 있었다. 다음 표 4는 삼중 모드에 대한 공격결과이다. 게다가 이러한 공격은 다른 다중 모드들 (사중모드, …)도 적용 가능하다.

Ⅳ. 결 론

본 논문에서는 모든 36개 이중 모드와 216개 삼중 모드에 대한 패딩 오라클 공격을 보였다. 그 결과, 12개의 이중 모드와 22개의 삼중 모드가 이 공격에 취약함을 알 수 있었다. 이 결과는 이러한 이중 모드와 삼중 모드들을 사용하더라도 패딩 오라클을 사용하는 환경에서는 단일 모드의 안전성 보다 더 나은 안전성을 주지 못한다는 사실을 나타낸다. 표 3과 4는 각각의 공격 결과를 요약해 놓은 것이다. 그러나 이 공격에 대한 대응책으로 해쉬나 MAC값을 이용하여 메시지 인증 부분을 추가한다면 이러한 형태의 패딩 오라클 공격을 막을 수 있을 것이다.

참 고 문 헌

[1] R. Baldwin and R. Rivest, *The RC5, RC5-CBC, RC5-CBC-Pad, and RC5-CTS algorithms*, RFC 2040, 1996.

[2] E. Biham, *Cryptanalysis of multiple modes of operation*, Journal of Cryptology, Vol. 11, No. 1, pp. 45-58, 1998.

[3] E. Biham, *Cryptanalysis of triple modes of operation*, Journal of Cryptology, Vol. 12, No. 3, pp. 161-184, 1999.

[4] J. Black and H. Urtubia, *Side-Channel Attacks on Symmetric Encryption Schemes: The Case for Authenticated Encryption*, InProc. of 11th USENIX Security Symposium, San Francisco 2002, pp.327-338, 2002.

[5] D. Hong, J. Sung, S. Hong, W. Lee, S. Lee, J. Lim, and O. Yi, *Known-IV Attacks on Triple Modes of Operation of Block Ciphers*, Advances in Cryptology - ASIACRYPT 2001, LNCS 2248, pp. 208-221, Springer-Verlag, 2001.

[6] ISO/IEC 9797-1: Information technology, *Security techniques - Message Authentication Codes (MACs). Part 1: Mechanisms using a block cipher*, 1999.

[7] ISO/IEC FDIS 10118-1: Information technology, Security techniques. Hash functions. Part 1: General (Final Draft), 2000.

[8] V. Klima and T. Rosa, *Side Channel Attacks on CBC Encrypted Messages in the PKCS#7 Format*, Available at IACR Cryptology ePrint Archive, Report 2003/098, 2003.

[9] G. Paterson and Arnold Yau, *Padding Oracle Attacks on the ISO CBC Mode Encryption Standard*, CT-RSA 2004, LNCS 2964, pp. 305-323, Springer-Verlag, 2004.

[10] S. Vaudenay, *Security Flaws Induced by CBC Padding - Applications to SSL, IPSEC, WTLS . . .*, Advances

in Cryptology - EUROCRYPT 2002, LNCS 2332, pp. 534-545, Springer-Verlag, 2002.

[11] D. Wagner, *Cryptanalysis of some recently-proposed multiple modes of operation*, Advances in Cryptology - FSE 1998, LNCS 1372, pp. 254-269, Springer-Verlag, 1998.

〈著者紹介〉

이 태 건 (Taekeon Lee) 학생회원
2002년 8월 : 한신대학교 수학과 학사
2003년 3월~현재 : 고려대학교 정보보호대학원 석사 과정
〈관심분야〉 블록 암호 및 운영모드 분석 및 설계

김 종 성 (Jongsung Kim) 학생회원
2000년 8월 : 고려대학교 수학과 학사
2002년 8월 : 고려대학교 정보보호대학원 석사
2004년 8월 : 고려대학교 정보보호대학원 박사 수료
2002년 9월~현재 : 고려대학교 정보보호대학원 박사 과정
2005년 3월~현재 : 벨기에 루벤대학 COSIC 연구소 박사 과정
〈관심분야〉 대칭키 암호알고리즘 설계 및 분석

이 창 훈 (Changhoon Lee) 학생회원
2001년 2월 : 한양대학교 수학과 학사
2003년 2월 : 고려대학교 정보보호대학원 석사
2005년 2월 : 고려대학교 정보보호대학원 박사수료
2003년 3월~현재 : 고려대학교 정보보호대학원 박사과정
〈관심분야〉 블록 암호, 스트림 암호, 운영모드, 해쉬함수, MAC 알고리즘 설계 및 분석

성 재 철 (Jaechul Sung) 종신회원
1997년 8월 : 고려대학교 수학과 학사
1999년 8월 : 고려대학교 수학과 석사
2002년 8월 : 고려대학교 수학과 박사
2002년 7월~2004년 1월 : 한국정보보호진흥원 선임연구원
2004년 2월~현재 : 서울시립대학교 수학과 전임강사
〈관심분야〉 암호 알고리즘 설계 및 분석

이 상 진 (Sangjin Lee) 정회원
1987년 2월 : 고려대학교 수학과 학사
1989년 2월 : 고려대학교 수학과 석사
1994년 2월 : 고려대학교 수학과 박사
1989년 2월~1999년 2월 : 한국전자통신연구원 선임 연구원,
1999년 2월~현재 : 고려대학교 자연과학대학 부교수, 고려대학교 정보보호대학원 겸임교수, 고려대학교 정보보호기술연구센터 연구실장
〈관심분야〉 블록 암호 및 스트림 암호의 분석 및 설계, 암호 프로토콜, 공개키 암호 알고리즘의 분석, 포렌식, 심층암호.