http://www.calsec.or.kr/jsebs
ISSN: 2288-3908

The Journal of Society for e-Business Studies
Vol.20, No.1, Feb. 2015, pp.89-97
http://dx.doi.org/10.7838/jsebs.2015.20.1.089

BOZ-PAD 방법을 사용하는 블록암호 기반 CBC|CBC 이중 모드에 대한 패딩 오라클 공격

Padding Oracle Attack on Block Cipher with CBC|CBC-Double Mode of Operation using the BOZ-PAD

황성진(Seongjin Hwang)*, 이창훈(Changhoon Lee)**

초 록

최근 개인정보 관련 사고들이 빈번하게 발생함에 따라 전자거래나 응용 환경의 개인 정보 및 민감한 정보들의 안전성에 대한 관심이 높아졌다. 인터넷 환경에서 데이터나 정보를 안전하게 보호하기 위해서 안전한 암호 알고리즘을 사용한다. 하지만 암호 적용방식이 올바르지 않으면 악의를 가진 공격으로부터 안전하지 않을 수 있다는 것이 연구 결과와 방법들로 소개되고 있다. 본 논문에서는 다양한 공격 방법들 중 CBC|CBC 모드에서 BOZ-PAD 방법을 사용하는 환경에 대해 패딩 오라클 공격을 적용한 결과를 소개한다.

ABSTRACT

In the various application environments on the internet, we use verified cipher algorithm to protect personal information of electronic commerce or application environments. Even so, if an application method isn't proper, the information you want to keep can be intercepted. This thesis studied about result of Padding Oracle Attack, an application environment which apply CBC|CBC operational mode based on block cipher and BOZ padding method.

키워드 : 패딩 오라클 공격, 정보보호, 블록암호
Padding Oracle Attack, Information Security, Block Cipher

이 연구는 서울과학기술대학교 교내 학술연구비 지원으로 수행되었습니다.

* First Author, Seoul National University of Science and Technology(hgoon6754@gmail.com)
** Corresponding Author, Seoul National University of Science and Technology(chlee@seoultech.ac.kr)

Received: 2014-12-08, Review completed: 2015-01-07, Accepted: 2015-01-23

1. 서 론

전자상거래 상에서 빈번하게 유출되고 있는 개인정보 및 민감한 정보를 보호하기 위해, 안전성이 검증된 암호기술을 권고하고 정보보호 점검 기준을 마련하여 시행[3, 6, 9]하는 등 기술적, 정책적으로 보안을 위해 노력하고 있다. 그렇지만 실제적으로 암호기술이 다양한 전자상거래 환경에서 잘못 적용되고 있는 있고, 이런 경우에는 민감한 정보들이 악의를 가진 공격에 노출될 수 있는 취약점을 가지게 된다.

특히, 2012년에는 인터넷 상 보안통신에 많이 사용되는 보안 토큰과 보안 프로토콜 Openssl의 취약점을 이용하는 패딩 오라클 공격이 제시되었고, 13분 안에 민감한 정보인 13분 안에 찾을 수 있음을 보였다.

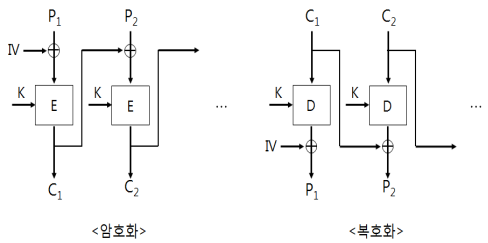
패딩 오라클 공격은 일반적으로 다음과 같은 방법으로 적용된다. 운영모드를 이용하여 메시지를 암호화할 경우, 입력 크기에 맞추기 위해서 메시지 마지막에 적당한 값을 덧붙여 입력크기를 일정하게 유지하게 된다. 만약 공격자가 메시지의 패딩이 옳은지 아닌지의 여부를 판단하는 오라클 정보를 알 수 있다면, 이 오라클 정보를 이용하여 암호문에 대응하는 평문을 알아낼 수 있다. 오라클은 공격자가 질문한 암호문을 복호화 하였을 때 평문의 패딩이 옳은지 아닌지를 판단하여 VALID 혹은 INVALID 값을 알려주는데, 공격자는 암호문을 중간에서 가로챌 수 있고, 이 과정에서 획득된 정보들을 이용하여 암호문에 대응하는 평문을 찾을 수 있다[8]. 공격자들로부터 메시지를 보호하기 위해 기존에 개발된 암호알고리즘을 보다 더 강하고 효율적인 새로운 암호 시스템과 운영 모드(modes of operation) 개발이 필요하게 되었고 그 방법으로 기존 제안되어 있는 알고리즘을 변형시키지 않고 안전성을 증가시키는 방법으로 이중 모드가 제안되었다.

본 논문에서는 BOZ-PAD 방법을 사용하는 이중 모드에 대한 패딩 오라클 공격을 적용하였다. 그 결과로 이중 모드를 사용하더라도 오라클 정보를 알 수 있는 환경에서는 단일 모드보다 더 나은 안전성을 제공하지 못한다는 사실을 제시한다.

2. CBC 모드 및 BOZ-PAD 방법

2.1 CBC 모드

CBC 모드는 메시지의 특정 블록이 변하게 되면 그 이후의 블록들이 모두 영향을 받기 때문에 암호화적인 특성이 우수하다고 할 수 있다. 그리고 IV는 송신자가 선택하여 수신자에게 보내게 되는데 이 값의 무결성은 보호되어야 한다. 무결성이 보호되지 않으면 공격자가 IV를 조작하여 복구될 암호문의 첫 번째 블록의 특정 비트를 알아낼 수도 있기 때문이다[7].



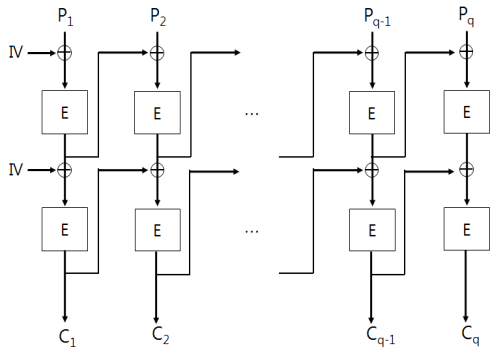
〈Figure 1〉 CBC-Mode

2.2 CBC|CBC 모드

다중 운영모드 방식(multiple modes of operation)은 기존에 개발된 암호알고리즘을 보다 더 강하고 효율적인 새로운 암호 시스템과 운영 모드 개발이 필요해 졌을 때, 기존 알고리즘을 변형시키지 않고 안전성을 증가시키는 방법 중에 하나로 다중 운영 모드 방식이 제안되었다. 다중 운영모드 방식은 여러 개의 단일 모드들로 구성된다. 즉 이전 모드의 출력이 다음 모드의 입력으로 하여 이전 블록의 출력을 다음블록의 입력이 되도록 하여 서로 피드백이 작용하도록 만드는 방식이다. 다중 모드 중 삼중 DES나 ECB 모드는 안전한 모드이지만 단일 모드에 비해 세 배의 암호화가 수행되어야 한다는 점에서 효율성이 떨어진다.

CBC|CBC 모드의 특성은 다음과 같다.

1. 동일한 메시지를 동일한 키와 동일한 IV를 사용하여 암호화 하면 같은 암호문이 생성된다. 그러나 IV 또는 메시지의 첫 블록을 바꾸게 되면 이후 암호문 블록이 이전 블록의 영향을 받아 모두 바뀐 값으로 생성된다.
2. 암호문 블록 C_q는 이전 블록의 모든 메시지의 영향을 받아 생성된다. 또한 메시지를 제대로 복호화 하기 위해서는 암호문의 순서가 올바르게 배열되어 있어야 한다.
3. 복호화 시, 암호문 블록 C_q에서의 한 비트 에러는 P_q와 그 이후 블록 모두에 영향을 준다. 이때, P_{q+2}은 C_q의 에러 발생 위치와 같은 위치에서 에러가 발생하게 된다.
4. 암호화 시, 암호문 블록 C_q에 변형이 일어나면 이후의 모든 암호문이 변하게 된다.



〈Figure 2〉 CBC|CBC-Mode

이중 모드를 포함한 다중 모드의 구성인 단일모드는 ECB 모드를 제외하고는 모두 피드백을 가지고 있는 모드이기 때문에 이를 기반으로 한 이중 모드는 내부에서 내부로 연결이 되는 여러 개의 피드백이 단일 모드들을 유기적으로 연결시켜 더욱 안전할 것으로 기대되었고 효율성 면에서도 효과적일 것으로 기대되었다. 그러나 이런 이중 모드도 여러 환경과 공격 방법에 따라 기대했던 안전성 보다 좋지 않은 결과를 가지게 된다.

패딩은 블록 암호 운영모드에서 입력된 데이터의 블록 크기를 일치시키기 위해서 사용한다. 패딩 방법은 처리 단위에 따라 바이트, 비트 기반으로 나누어진다.

2.2 BOZ-PAD

BOZ-PAD 방법은 먼저 처음 패딩되는 바이트에 0x80, 그 이후에는 0x00로 붙여 패딩 패턴을 만들게 된다. 패딩 바이트가 n바이트라 가정한다면 패딩 패턴은 0x8000…0000가 된다. 또한 <Figure 3>과 같은 경우에는 입

력된 데이터의 크기가 블록 사이즈와 일치하는 경우이다. 이 때 BOZ-PAD는 유효 패딩 자리 수를 구분하기 위해서 한 블록을 추가하여 0x8000…0000으로 패딩 처리를 하여 구분하도록 한다.

Block 1	Block 2	Block 3	……	Block (q-1)	Block q
					8000…000000

〈Figure 3〉 BOZ-PAD

3. 관련 연구

송신자가 메시지를 전송할 때 메시지 뒤에 블록 크기보다 부족한 만큼 덧붙이고 암호화한 뒤 수신자에게 전송을 한다. 이렇게 전송된 암호화된 메시지는 수신자가 복호화 한 후, 이 때 패딩 정보가 올바른지 아닌지를 판단하여 송신자에게 VALID 혹은 INVALID 값을 전송한다. 공격자는 이 때 이 정보를 이용하여 패딩 오라클 공격을 적용할 수 있다. 여기서는 편의상 CBC 모드에 대한 패딩 오라클 공격을 제시한다.

3.1 Vaudenay가 제안한 패딩 오라클 공격 방법

운영 모드에 대한 패딩 오라클 공격은 EUROCRYPT 2002에 Vaudenay에 의해 처음으로 소개되었다[10]. 2002년 Vaudenay는 복호화 된 메시지의 패딩 방식이 옳은지 아닌지를 판단하여 정보를 제공하는 패딩 오라클을 이용하여 암호문에 대응하는 평문을 복구할 수 있는 공격 방법을 처음 소개하였다 CBC-PAD 방법은 PKCS #7에 명시되어 있는 패딩 방법이다[4]. 이 공격은 공격자가 조작하여 보낸 암호문을 복호화하여 얻어진 메시지의 패딩 값이 옳으면 VALID를, 옳지 않으면 INVALID를 리턴하는 패딩 오라클을 이용한다. 이 공격의 목표는 오라클에 대한 여러 번의 질의 및 응답을 이용하여 획득한 암호문에 대응하는 평문을 복구해내는 것이다. 획득한 암호문에 대응하는 메시지를 복구하기 위한 공격 과정은 두 단계로 이루어져 있다.

첫 번째 단계에서는 마지막 암호문 블록 패딩 부분의 워드를 복구하는 과정으로, 한 블록의 임의의 값($R = r_1, r_2, \cdots, r_n$)을 선택하여 오라클에게 $O(R||C_q)$에 대해 질의한다. 그러면 오라클로부터 VALID 혹은 INVALID 값을 받을 수 있다. 만약 오라클로 부터의 응답이 INVALID면, 처음 0으로 초기화 시킨 i를 증가시켜 다시 오라클에게 질의하게 된다. 이와 같은 과정을 VALID 응답이 올 때까지 질의하게 된다. VALID 응답이 오게 되면 r xor i값을 r_n으로 대치한다.

그리고 j를 n으로 설정한 후 2까지 감소시키며 $R = r_1, r_2, \cdots, r_{n-j+1}, r_{n-j+2}, \cdots, r_n$으로 설정한 후 오라클에게 질의를 진행한다. 이때 오라클의 응답이 INVALID이면 멈추고 (r_{n-j+1} xor j)…(r_n xor j)값을 출력하고 그렇지 않으면 (r_n xor 1)값을 출력한다.

두 번째 단계에서는 마지막 암호문 블록에 대응하는 평문 블록을 찾는다. 평문 블록을 $D(C_q)$라 하고 $D(C_q) = a_1\ a_2 \cdots a_n$이라 한다. 이미 앞에 첫 번째 과정을 통해 $a_k \cdots a_n (k \leq n)$을 얻었다. 그러면 i을 k로 설정한 후 n까지 $r_i = a_i$ xor(n-k+2)을 진행한다. 임의의 값 ($R = r_1, r_2, \cdots, r_{k-1}$)을 선택하고 i를 0으로 설정

한 후 R = r_1, r_2, …, r_{k-2} …, r_n로 설정한다. 이 때 오라클로부터 응답이 INVALID이면 i를 증가시키고 다시 오라클에게 질의한다. 그리고 VALID 응답을 받았을 때 r_{k-1} xor i xor (n-k+2)의 값을 출력하게 된다.

이 공격 과정은 평균적으로 2^7번의 오라클 질의가 필요하다. 따라서 한 평문 블록을 복구하기 위해서는 2^7×n번의 오라클 질의가 필요하다 또한 2단계에서 마지막 암호문 블록 C_q 대신 다른 암호문 블록을 적용하면 전체 평문 정보를 복구할 수 있다. 이 때 필요한 오라클 질의 수는 2^7×n×q이다.

3.2 Black, Urtubia의 패딩 오라클 공격

Black 등은 Vaudenay의 공격 방법을 개선시키고 CBC-PAD 패딩 방법 이외에 다양한 패딩 방법을 사용 했을 경우의 CBC 운영 모드에 대한 패딩 오라클 공격에 대해 적용하였다[2].

Black 등이 개선한 공격방법은 이진 검색 알고리즘을 통해 먼저 패딩 길이를 구하는 것이다. 그리고 나서 나머지 평문을 구한다. 공격자는 평문 메시지 P의 패딩 길이를 구하기 위해서 이진 검색(binary search)을 수행한다. 메시지의 마지막 블록에서, 공격자가 만약 패딩 바이트 부분의 임의의 1바이트에 해당하는 암호문 블록의 바이트를 변화시켜 오라클 O에게 질의를 하였다면, 오라클 O는 항상 INVALID값을 대답할 것이고, 반대로 메시지 바이트 부분을 변화시켜 질의를 하였다면 오라클 O는 항상 VALID 값을 대답할 것이다. 따라서 한 번에 1바이트를 변형하여, 반복적으로 이진 검색을 수행하면 메시지의 패딩 길이를 구할 수 있다. 먼저 공격자는 CBC 모드로 암호화되어 있는 두 블록으로 구성된 암호문 C = {IV, C_1}를 획득하였다고 가정할 때, 먼저 암호문 블록 IV의 중간 바이트를 변형하여 오라클에게 질문하면 오라클은 VALID 또는 INVALID 값을 대답할 것이다. 만약 오라클이 응답한 값이 VALID이면 공격자는 오른쪽 n/4 번째 바이트를 변형하여 오라클에게 질문하고, 그렇지 않으면 왼쪽 n/4번째 바이트를 변형하여 질문한다. 이런 식의 과정을 반복하면 $\log_2(n)$번의 오라클 질문으로 패딩의 길이를 알 수 있다.

여기서 n의 값은 한 블록의 크기를 의미한다.

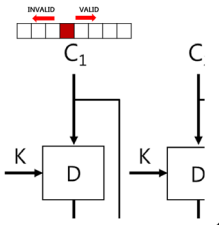
3.3 CBC-PAD를 사용하는 이중모드에 대한 패딩 오라클 공격

이중 모드에 대한 패딩 오라클 공격은 단일 모드에 대한 패딩 오라클 공격 방법에서 제시된 패딩 오라클을 사용하여 주어진 암호문에 대응하는 평문을 찾는다[1, 5]. 본 소절에서 패딩 오라클 공격 적용 모델은 CBC|CBC 이중 모드를 사용하고 CBC-PAD를 적용한다. 패딩 오라클 공격에 안전하지 못한 패딩 방법으로는 CBC-PAD, ESP-PAD, XY-PAD, ISO(9797-1)-PAD, ISO(10118-1)-PAD3가 있다.

3.3.1 패딩 길이 구하기

P_q에서의 패딩 길이를 알기 위해서는 C_{q-2}의 조작을 통해 이진 검색을 시행한다. 가운데 한 바이트를 변형하고 오라클의 응답 값이 VALID이면 공격자는 오른쪽 n/4번째 바이트를 변형하여 오라클에게 다시 질문하고 응답 값이 INVALID 이면 왼쪽으로 n/4번째 바이트로 이

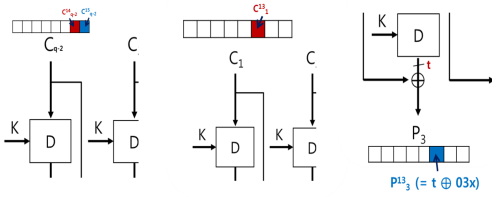
동하여 오라클에게 질문하는 방식을 반복하면 $\log_2(n)$ 만에 패딩 길이를 알 수 있다.



〈Figure 4〉 Binary Search

3.3.2 평문 구하기

공격자는 앞서 패딩 길이를 구하게 되면 패딩이 01x, 0202x,… 중에 한가지고 패딩 되었음을 안다. 따라서 암호문 C_{q-2}를 패딩 된 바이트보다 1만큼 큰 값으로 조작할 수 있다. 만약 마지막 블록의 2바이트가 패딩 된 경우라면, 공격자는 P_q값의 P^{14}_q, P^{15}_q값이 0202x에서 0303x 값이 되도록 C^{14}_{q-2}, C^{15}_{q-2} 값을 조작한다.



〈Figure 5〉 Recovery of Plaintext Corresponding to the Ciphertext

공격자는 암호문 블록의 바이트 C^{13}_q을 모든 t에 대해, t xor C^{13}_{q-2}(0< = t< = 2^8-1) 값으로 바꾸어, 바뀐 암호문을 패딩 오라클에게 질문한다. 이 때, 패딩 오라클이 VALID 값을 응답했을 경우, 공격자는 그때의 t 값을 사용하여 평문 P^{13}_q(= t xor 03x) 값을 구한다.

4. BOZ-PAD를 사용하는 CBC|CBC 모드에 대한 패딩 오라클 공격

본 장에서는 BOZ-PAD를 사용하는CBC|CBC 모드에 대한 패딩 오라클 공격을 소개한다.

4.1 패딩 길이 구하기

공격자에게 패딩 오라클 O와 정당한 암호문 C = (IV, C_1, C_2,…,C_q)를 획득하였다고 가정하고, 공격자는 평문 메시지 P의 패딩길이를 구하기 위해서 Black 등이 소개한 공격방법인 이진 검색(binary search)을 수행한다. 한 번에 한 바이트를 변화시켜 반복적으로 이진 검색을 수행하면 평문 메시지의 패딩 길이를 구할 수 있다. 먼저 암호문 블록 C_{q-2}의 중간 바이트를 변형하여 오라클에게 질의하면 오라클은 VALID 또는 INVALID 값을 응답한다. 만약 오라클이 응답한 값이 VALID이면 공격자는 오른쪽 n/4번째 바이트를 변형하여 오라클에게 다시 질의를 하고, 그렇지 않으면 왼쪽 n/4번째 바이트를 변형하여 질의하게 된다. 이런 방법으로 이 과정을 반복하게 되면 패딩 길이를 $\log_2(n)$번의 오라클 질문으로 패딩 길이를 알 수 있다.

1단계 : 이진 검색 알고리즘

Length(x, y)
만약 x의 값이 y값과 같으면 return x
m = [(x+y)/2]
만약 O(IV^m, C_1)가 INVALID이면 Length (x, y) 출력하고
그렇지 않으면 Length(m-1, y)를 출력한다.

4.2 암호문에 대응하는 평문 찾기

위 패딩 길이를 구하는 과정을 통해 얻은 평문 메시지의 패딩 길이 정보와 암호문에 대응하는 평문을 찾는 알고리즘을 이용하여 주어진 암호문에 해당하는 평문을 구한다. P_q블럭의 메시지 길이 정보를 변경하기 위해서는 패딩 정보를 통해 메시지 마지막 바이트 위치를 알기 때문에 C_{q-2} 번째의 블록의 대응되는 바이트를 변형을 시켜 메시지의 길이 정보를 변형시킨다. 그리고 오라클에게 조작한 암호문 C′값을 질의하여 오라클의 대답을 통해 평문 값을 한 바이트씩 찾아 나간다. 이러한 과정을 반복하면 평문 한 블록을 알아낼 수 있고, 계속해서 한 블록씩 찾아간다면 평문 모든 값을 알 수 있다.

2단계 : 마지막 암호문 블록 복구

○ 입력 : $R||C_q(R = r_1, r_2, \cdots, r_{k-1}, r_k, \cdots, r_n)$
- $R = r_k, \cdots, r_n$: 1단계 과정을 통해 고정된 값

○ 출력 : 마지막 암호문 블록 k번째 평문 워드 복구

○ 과정 :

1. 초기화 : $i = 0$
2. $R = r_1, r_2, \cdots, r_{k-1}, r_k, \cdots, r_n$
3. 만일 $O(R||C_{q-2})$가 INVALID이면 i를 증가시켜 2번 과정 수행
4. 만일 $O(R||C_{q-2})$가 VALID이면 r_k xor i xor (n-k+2) 출력

CBC|CBC 모드는 앞선 공격 과정을 수행하면 모든 평문을 찾을 수 있다. 이 공격 과정에 필요한 오라클 질의 수는 $2^7 \times n \times q + \log_2(n)$가 요구된다.

앞서 제시한 결과를 분석해 보면 CBC|CBC 모드에 BOZ-PAD 방법을 사용하는 환경이 패딩 오라클 공격에 취약할 수 있다는 것을 알 수 있다.

5. 결 론

본 논문에서는 CBC|CBC 모드에서 BOZ-PAD 방법을 사용하는 환경에 대해 패딩 오라클 공격 가능성에 대해서 분석하였고 그 결과로 CBC|CBC 모드에서 BOZ-PAD 방법이 패딩 오라클 공격에 취약할 수 있다는 것에 알 수 있었다. 패딩 오라클 공격은 기본적으로 블록기반 운용방식에 데이터 블록크기를 맞추기 위해 적용되는 다양한 패딩 방법의 취약성을 이용한 공격방법이지만 반드시 블록기반 운영 모드에서만 적용되는 것은 아니다. 스트림 모드를 사용하는 서비스에서도 공격자가 패딩 길이나 메시지 길이 정보를 획득할 수 있는 경우 패딩 오라클 공격을 적용할 수 있게 된다. 전자상거래나 응용 환경에서의 개인정보 및 민감 정보를 안전하게 보호하기 위해서 앞으로 서비스 암호화 방식 설계 시에 패딩 길이나 패딩 패턴 등을 노출시키지 않도록 설계하는 방법에 대한 연구가 필요하다.

References

[1] Biham, E., “Cryptanalysis of multiple modes of operation,” Lecture Notes in Computer Science, Vol. 917, pp. 278-292, 1995.

[2] Black, J. H. and Urtubia, “Side-Channel Attacks on Symmetric Encryption Schemes : The Case for Authenticated Encryption,” USENIX, 2002.

[3] Jin, C. Y., Kim, A. C., and Lim, J. I., "Correlation Analysis in Information Security Checklist Based on Knowledge Network," The Journal of Society for e-Business Studies, Vol. 19, No. 2, pp. 109-124, 2014.

[4] Klíma, V. and Rosa, T., "Side Channel Attacks on CBC Encrypted Messages in the PKCS#7 Format," eprint, 2003.

[5] Lee, T. K., Kim, J. S., Lee, C. H., Sung, J. C., Lee, S. J., and Hong, D. W., "Padding Oracle Attacks on Multiple Modes of Operation," Lecture Notes in Computer Science, Vol. 3506, pp. 343-351, ICISC, 2004.

[6] Oh, N. S., Han, Y. S., Eom, C. W., Oh, K. S., Lee, B. G., "Developing the Assessment Method for Information Security Levels," The Journal of Society for e-Business Studies, Vol. 16, No. 2, pp. 159-169, 2011.

[7] Paterson, K. G. and Yau, A., "Padding Oracle Attacks on the ISO CBC Mode Encryption Standard", CT-RSA 2004, LNCS, Vol. 2964, pp. 305-323, Springer-Verlag, 2004.

[8] Rizzo, J. and Duong, T., "Practical Padding Oracle Attacks," USENIX WOOT 2010.

[9] Seo, Y. J. and Han, S. Y., "An Information Flow Security Based on Protected Area in eCommerce," The Journal of Society for e-Business Studies, Vol. 15, No. 1, pp. 1-16, 2010.

[10] Vaudenay, S., "Security Flaws Induced by CBC Padding, Applications to SSL, IPSEC, TLS…", Eurocrypt 2002, LNCS, Vol. 2332, pp. 534-545, Springer-Verlag, 2002.

저 자 소 개

황성진 (E-mail : hgoon6754@gmail.com)
2013년 한신대학교 컴퓨터공학과 (학사)
2013년～현재 서울과학기술대학교 컴퓨터공학과 (석사과정)
관심분야 정보보호, 암호알고리즘, 네트워크 보안

이창훈 (E-mail : chlee@seoultech.ac.kr)
2001년 한양대학교 수학과 (학사)
2003년 고려대학교 정보보호대학원 정보보호학과 (석사)
2001년 고려대학교 정보경영공학대학원 정보보호학과 (박사)
2001년～2012년 한신대학교 컴퓨터공학부 조교수
2001년～현재 서울과학기술대학교 컴퓨터공학과 조교수
관심분야 정보보호, 암호학, 디지털포렌식, 융합보안, 컴퓨터이론