KIPS Tr. Comp. and Comm. Sys.
Vol.4, No.2 pp.73~80 pISSN: 2287-5891

http://dx.doi.org/10.3745/KTCCS.2015.4.2.73

Analysis of Padding Oracle Attack Possibility about Application Environment; SRTP, MIKEY, CMS, IPSec, TLS, IPTV

Seongjin Hwang† · Myungseo Park†† · Dukjae Moon††† · HyungChul Kang†††† · Jongsung Kim††††† · Changhoon Lee†††††††

ABSTRACT

In the various application environments on the internet, we use verified cipher algorithm to protect personal information. Even so, if an application method isn't proper, the information you want to keep can be intercepted. One of the representative examples of it is a PADDING ORACLE ATTACK. This thesis studied about STRP, MIKEY, CMS, IPSec, TLS, IPTV, an application environment which apply CBC operational mode based on block cipher and CBC padding method, and about whether we can attack against the Padding Oracle Attack as well as the vulnerable points.

Keywords : Block Cipher, Padding, Padding Oracle Attack

SRTP, MIKEY, CMS, IPSec, TLS, IPTV에 대한 패딩 오라클 공격 가능성 분석

황 성 진† · 박 명 서†† · 문 덕 재††† · 강 형 철†††† · 김 종 성††††† · 이 창 훈†††††††

요 약

인터넷상의 다양한 응용환경에서는 개인정보 및 민감한 정보를 보호하기 위해서 안전성이 검증된 암호 알고리즘을 사용한다. 하지만 안전성이 검증된 암호 알고리즘을 사용하여도 주어진 암호 알고리즘을 적용하는 방식이 올바르지 못하면 보호하고자 하는 정보들이 누출될 수 있다는 연구 결과와 공격기법들이 소개되고 있다. 이 공격방법들 중 대표적인 사례가 패딩 오라클 공격이다. 본 논문에서는 블록 암호 기반 CBC 운영모드와 CBC 패딩 방법을 적용하는 응용환경인 STRP, MIKEY, CMS, IPSec, TLS, IPTV에 대해 패딩 오라클 공격을 적용하고, 패딩 오라클 공격에 대한 공격 가능 여부 및 취약점에 관하여 연구하였다.

키워드 : 블록 암호, 패딩, 패딩 오라클 공격

1. 서 론

최근 개인정보 관련 사고들이 빈번히 발생하면서 개인정보 및 민감한 정보들의 안전성에 대한 관심이 높아졌다. 기본적으로 정보들을 보호하기 위해서는 안전성이 검증된 암호 알고리즘을 사용한다. 하지만 주어진 암호 알고리즘이 응용환경의 적용방법에 따라서 잘못 적용될 경우에는 보호하고자 하는 정보들이 노출될 수 있는 위험성을 가지고 있다.

실제 이러한 위험성이 알려지게 된 연구는 2002년 Vaudenay에 의해 처음 소개[1]되었는데, CBC 운영모드 및 CBC-PAD를 사용하는 응용환경에 패딩 오라클 공격방법을 적용한 연구 결과를 발표하였다. 복호화된 평문의 패딩이 옳은지 아닌지를 판단하여 응답하는 패딩 오라클을 이용하여 메시지의 길이 정보나 패딩 패턴 등을 알아내어 평문 메시지를 알아내는 공격방법으로, 이를 패딩 오라클 공격이라 한다.

이후 2010년에는 Duong과 Rizzo가 Black Hat Europe에서 CAPTCHA(Completely Automated Public Turing test Computers and Humans Apart) 시스템과 JSF(JavaServer Faces) 등의 취약점을 소개[2]하였고 2012년에는 미국 EMC 산하 RSA사가 제공하는 인증제품인 SecurID 800이 생성하는 암호키를 패딩 오라클 공격으로 13분 만에 찾아낼 수 있다는 연구 결과가 보도되었으며, 실제 공격 결과가 저명 암

† 준 회 원 : 서울과학기술대학교 컴퓨터공학과 석사과정
†† 준 회 원 : 국민대학교 수학과 석사과정
††† 정 회 원 : 고려대학교 정보보호학과 박사
†††† 준 회 원 : 고려대학교 정보보호대학원 석 · 박사통합과정
††††† 정 회 원 : 국민대학교 금융정보보안학과 조교수
††††††† 종신회원 : 서울과학기술대학교 컴퓨터공학과 조교수
Manuscript Received : November 19, 2014
Accepted : December 11, 2014
* Corresponding Author : Changhoon Lee(chlee@seoultech.ac.kr)

호 학술대회인 CRTPTO 2012에서 발표되었다. 이 후 2013년 5월 IEEE S&P에서 영국의 Royal Hollway 대학 정보보호 연구실은 블록 암호 기반 CBC 운영모드에 대한 패딩 오라클 공격을 이용하여 TLS(Transport Layer Security) 프로토콜을 통해 암호화 통신되는 데이터에서 평문정보를 획득할 수 있다는 결과를 소개하였다. 실제 다양한 응용환경에서 TLS 프로토콜을 적용하고 있는 만큼 이 공격에 대한 파급효과가 클 것으로 예상된다.

본 논문에서는 SEED 암호 운영모드, STRP, MIKEY, CMS, IPSec, TLS, IPTV의 명세된 문서들에 대해서 패딩 오라클 공격을 적용하고 분석하였다. 그 결과로 SEED 운영모드, TLS v1.0은 취약한 것으로 나타났고 그 외 CMS나 IPSec가 조건적으로 패딩 오라클 공격이 적용 가능하다는 것을 알 수 있었다(Table 1 참고). Table 1에서 조건적 공격 가능의 경우란, 암호화 인증 기술을 적용하는 경우 공격 적용이 어렵지만 이 기술을 사용하지 않을 경우 공격 적용이 가능한 경우를 말한다. 본 논문은 안전한 암호 알고리즘을 사용하더라도 적용방법이 올바르지 않다면 패딩 오라클 공격의 위험성으로부터 자유롭지 못함에 대해서 제시하고자 한다.

본 논문의 구성은 다음과 같다. 2절에서는 기본적으로 운영모드와 패딩 방법에 대해서 설명하고, 3절에서는 관련 연구로 2002년 제안된 Vaudeney에 의해 소개된 패딩 오라클 공격방법과 적용 서비스 대상들에 대해서 설명하고, 4절에서는 각 웹 응용환경에 대해 패딩 오라클 공격을 적용한 결과를 분석한다. 그리고 5절에서 본 논문의 결과를 요약한다.

Table 1. Padding Oracle Attack possibility for a variety of application environments

Application	Availability of Padding Oracle Attack
SRTP	Disable
MIKEY	Disable
CMS	Conditionally available
IPSec v1	Conditionally available
IPSec v2	Conditionally available
IPSec v3	Conditionally available
TLS v1.0	Conditionally available
TLS v1.1	Disable
TLS v1.2	Disable
IPTV	Disable

2. CBC 운영모드 및 CBC-PAD 방법

2.1 CBC 운영모드

운영모드는 평문의 길이가 블록 암호의 블록 크기보다 클 경우 블록 암호를 적용시키기 위한 방법이다. 다양한 응용환경에 적절한 암호화 도구로 사용할 수 있는 효율적인 운영 방식으로 CBC 운영모드를 제시하고 있다.

CBC(Cipher Block Chanining) 운영모드는 ECB 운영모드의 평문 블록의 패턴이 암호문 블록에도 그대로 나타나는 단점을 보완한 방식으로, 각 평문 데이터 블록이 이전 단계의 암호문 블록과 XOR된 후에 암호 알고리즘을 통해 암호화가 진행된다(Equation (1)). 이전 암호 블록이 없는 첫 번째 블록인 경우에는 IV(Initial Vector)라는 임의의 데이터 혹은 정해진 데이터값과 XOR한 후에 암호화된다.

$$\begin{aligned} &Encryption: C_i = E_K(P_i \oplus C_{i-1})\ (C_0 = IV) \\ &Decryption: P_i = D_K(C_i) \oplus C_{i-1}\ (C_0 = IV) \end{aligned} \quad (1)$$

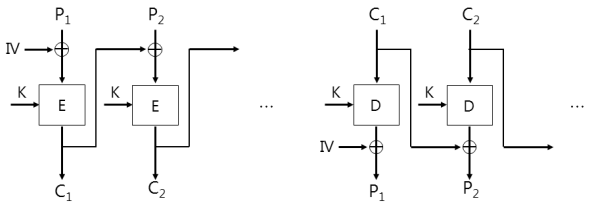

Fig. 1. CBC-mode

CBC모드는 다양한 응용 프로토콜에서 가장 널리 사용되는 방법으로 특성은 다음과 같다.

동일한 평문을 동일한 키와 동일한 IV를 사용하여 암호화하면 같은 암호문이 생성된다. 그러나 IV 또는 평문의 첫 블록을 바꾸게 되면 암호문은 이전 블록의 영향을 받게 되므로 암호문이 모두 바뀐 값으로 생성된다.

암호문 블록 C_i는 모든 평문의 영향을 받아 생성된다. 또한 평문을 제대로 복호화하기 위해서는 암호문의 순서가 올바르게 배열되어있어야 한다.

복호화 시, 암호문 블록 C_i에서의 한 비트 에러는 P_i와 P_{i+1}에 영향을 준다. 이때, P_{i+1}은 C_i의 에러 발생 위치와 같은 위치에서 에러가 발생하게 된다.

암호화 시, 암호문 블록 C_i에 변형이 일어나면 이후의 모든 암호문이 변하게 된다.

CBC 운영모드는 평문의 특정 블록이 변하면 그다음에 출력되는 블록들이 모두 영향을 받기 때문에 암호학적인 특성이 우수하다고 할 수 있다. 그리고 IV는 송신자가 선택하여 수신자에게 보내게 되는데 이 값은 무결성이 보호되어야 하지만 비밀값으로 유지되지 않아도 된다. 왜냐하면 공격자가 IV를 조작하여 복구될 암호문의 첫 번째 블록의 특정 비트를 알아낼 수도 있기 때문이다.

2.2 CBC-PAD

패딩은 블록 암호 운영모드에서 입력된 데이터의 크기가 블록의 크기인 n보다 작을 경우에 블록 크기를 일치시키기 위해서 사용한다. 패딩 방법은 처리 단위에 따라 바이트, 비트 기반으로 나누어진다.

CBC-PAD는 바이트 기반 패딩이며, 덧붙이는 바이트 수에 따라서 패딩 패턴이 결정된다. 덧붙이는 바이트의 수가 i 바이트라 가정한다면 패딩 패턴은 $0\xi iii$가 된다. 예를 들

어, 부족한 바이트가 2 바이트일 때에는 $0x0202$를 패딩하게 된다. 또한, 패딩되지 않은 블록을 구분하기 위해서 한 블록을 추가하여 $0xnnnn...$을 패딩한다.

Block 1	Block 2	Block 3	……	Block (q-1)	Block q
					0202

Block 1	Block 2	Block 3	……	Block (q-1)	Block q
					nnn…nnn

Fig. 2. Examples of CBC-PAD

3. 관련 연구

3.1 Vaudeney가 제안한 패딩 오라클 공격방법

2002년 Vaudeney는 복호화된 평문의 패딩 방식이 옳은지 아닌지를 판단하여 정보를 제공하는 패딩 오라클을 이용하여 평문을 복구할 수 있는 공격방법을 처음 소개하였다[1]. 이 공격방법은 CBC 운영모드와 CBC-PAD를 사용하였을 때 적용 가능한 것으로 SSL, TLS v1.0 및 WTLS 등의 응용환경에 적용 가능하다.

이 공격은 공격자가 조작하여 보낸 암호문을 복호화하여 얻은 평문의 패딩값이 옳으면 VALID를, 옳지 않으면 INVALID를 리턴하는 패딩 오라클을 이용한다. 이 공격의 목표는 오라클에 대한 여러 번의 질의 및 응답을 이용하여 획득한 암호문에 대한 평문을 복구해내는 것이다.

획득한 암호문에 대한 평문을 복구하기 위한 공격 과정은 두 단계로 이루어져 있다($O(x)$는 x에 대한 패딩 오라클의 리턴값).

1단계: 마지막 암호문 블록 패딩 부분 워드 복구

o 입력: $R\|C_q$ ($R=r_1,r_2,\cdots,r_n$ - 랜덤한 n 워드 블록)
o 출력: 패딩 부분 평문 복구
o 과정:
1. 초기화: $i=0$
2. $R=r_1,r_2,\cdots,(r_n\oplus 1)$
3. 만일 $O(R\|C_q)$가 INVALID면 i를 증가시켜 2번 과정 수행
4. 만일 $O(R\|C_q)$가 VALID면 $r_n=r_n\oplus 1$
5. j를 n부터 2까지 감소시키며 다음 과정 수행
- $R=r_1,r_2,\cdots,r_{n-j+1},(r_{n-j+1}\oplus 1),r_{n-j+2},\cdots,r_n$
- 만일 $O(R\|C_q)$가 INVALID면 과정을 멈추고, $(r_{n-j+1}\oplus j),\cdots,(r_n\oplus j)$ 출력
6. 그렇지 않으면 $r_n\oplus 1$ 출력

2단계: 마지막 암호문 블록 복구

o 입력: $R\|C_q$ ($R=r_1,r_2,\cdots,r_{k-1},R_k,\cdots,R_n$)
- $R_k,\cdots,R_n$: 1 단계를 통해 고정된 값
o 출력: 마지막 암호문 블록 $k-1$ 번째 평문 워드 복구
o 과정:
1. 초기화: $i=0$
2. $R=r_1,r_2,\cdots,(r_{k-1}\oplus 1),R_k,\cdots,R_n$
3. 만일 $O(R\|C_q)$가 INVALID면 i를 증가시켜 2번 과정 수행
4. 만일 $O(R\|C_q)$가 VALID면 $r_{k-1}\oplus i\oplus(n-k+2)$ 출력

마지막 평문을 복구하면, 마지막 암호문은 제거하고 그 앞의 암호문을 마지막 암호문으로 가정한다. 위의 과정을 동일하게 적용하면 해당 평문을 복구할 수 있고, 이 과정을 반복 적용하면 전체 평문을 복구할 수 있다. 이 공격 과정은 평균적으로 2^7번의 오라클 질의가 필요하다. 따라서 한 평문 블록을 복구하기 위해서는 $2^7\times n$번의 오라클 질의가 필요하다. 또한, 2단계에서 마지막 암호문 블록 C_q 대신 다른 암호문 블록을 적용하면 전체 평문 정보를 복구할 수 있다. 이때 필요한 오라클 질의 수는 $2^7\times n\times q$이다.

3.2 Postfix 일치 확인 오라클 공격

본 공격은 공격 대상 서비스에 패딩 오라클 질의의 횟수 제한이 있는 경우에 적용 가능한 방법이다[1]. 이 공격방법은 평문의 마지막 블록의 일부 정보를 예측 가능할 경우에 적용 가능한 방법으로, 예측 가능한 정보에는 평문의 패딩 정보나 서비스에서 제공하는 메시지 패턴 등이 있다. 이 공격을 위해서는 확률적으로 공격이 수행되는 폭탄 오라클을 구성하게 되는데, 이 방법은 예측한 정보가 맞을 경우 오라클 공격이 성공하지만, 정보가 틀릴 경우에는 오라클 공격이 감지되어 서비스가 차단되어 실패하게 된다. $m>n$인 경우, 암호문을 분할하여 폭탄 오라클을 적용한다.

폭탄 오라클

o 초기 예측 정보: $W_1,W_2,\cdots,W_m$ $(m\le n)$
o 출력: 예측 정보의 일치여부
o 과정:
1. $n-m$ 랜덤 워드 $R_1,R_2,\cdots,R_{n-m}$ 선택
2. $R_{n-m+k}=W_k\oplus m$ $(k=1,\cdots,m)$
3. $O(R\|C_q)$ $(R=R_1,R_2,\cdots,R_n)$
4. 만일 $m=1$이면
$R'_k=R_k$ $(k=1,\cdots,n-2,n)$, $R'_{n-1}=R_{n-1}\oplus 1$
아니면, $R'_k=R_k$ $(k=1,\cdots,n-1)$, $R'_n=W_m\oplus 1$
5. $O(R'\|C_q)$ $(R'=R'_1,R'_2,\cdots,R'_n)$
6. True 출력

3.3 블록 암호 기반 CBC 운영모드 및 CBC-PAD를 사용하는 SSL/TLS, WTLS 환경에서의 기존 분석 결과

[1]에서는 SSL/TLS v1.0 및 WTLS에 대한 패딩 오라클 공격 적용 결과에 대해서 소개하였다. 본 절에서는 이 응용환경에 대해서 소개하고 패딩 오라클 공격이 적용되는 이유를 설명한다.

1) SSL/TLS v1.0

SSL과 TSL v1.0 프로토콜들은 CBC 운영모드를 사용하고 CBC-PAD를 사용한다. 따라서 앞서 소개된 패딩 오라클 공격이 적용될 수 있다. TSL v1.0의 경우에는 메시지 인증 코드(MAC) 알고리즘을 사용할 수 있으나, MAC 알고리즘

을 패딩 전에 사용하는 MAC-Encoding-Encrypt(MEE) 방식을 적용하여 MAC값 확인 전에 패딩을 판별한다. 따라서 패딩 오라클 공격의 적용이 가능하다. SSL v3.0의 경우에는 MAC값 오류와 패딩 오류를 같은 오류로 제시하기 때문에 기본적인 패딩 오라클 공격 적용이 불가능해 보이지만, 이 두 오류들을 구별할 수 있는 방법이 제안된다면 공격이 가능함을 [1]에서 소개하였다. 특히, 실제 서비스에서는 패딩 오류 시 세션을 종료시키기 때문에 Postfix 일치 확인 오라클을 이용한 확률적 패딩 오라클 공격을 적용해야 한다.

2) WTLS

WTLS 프로토콜은 CBC 운영모드를 사용하고 CBC-PAD를 사용한다. 전송 데이터에 대한 오류가 발생하였을 때 재전송 횟수를 제한하므로 여러 번 질의하는 형태인 기본적인 패딩 오라클 공격의 적용은 불가능하다. 특히, WTLS 내 서로 다른 프로토콜 규격에 따른 데이터 캡슐화 과정으로 각 프로토콜이 제공하는 암호화 방식이 적용되는데 이 과정을 우회할 수 있는 경우 패딩 오라클 공격을 적용할 수 있는 것으로 소개되었다.

4. STRP, MIKEY, CMS, IPSec, TLS, IPTV 분석 결과

본 절에서는 패딩 오라클 공격을 STRP, MIKEY, CMS, IPsec, TLS, IPTV 등의 응용환경에 적용 가능한가에 대한 분석 결과를 소개한다.

4.1 SRTP

SRTP(Secure Real-time Transport Protocol)는 오디오, 비디오와 같은 실시간 데이터를 전송할 때 적합한 실시간 프로토콜의 기밀성, 메시지 인증, 재생 공격 방어를 제공하는 안전한 실시간 전송 프로토콜이다[3].

SRTP에서 암호 알고리즘은 AES를 기본(default)으로 사용하며, 운영모드로는 Counter 모드[4]와 OFB 모드의 변종인 f8 모드[5]를 사용한다. MAC은 HMAC-SHA1을 기본으로 사용한다. 패딩 방법(RTP padding)은 사용자가 임의로 정하여 사용한다.

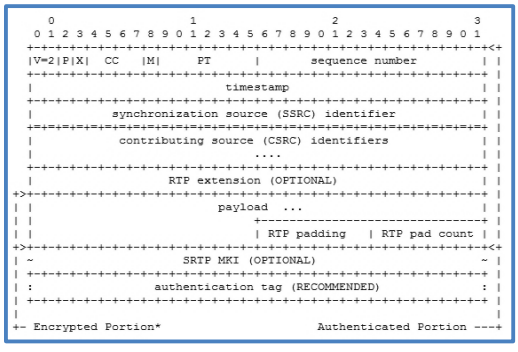

Fig. 3. Structure of the SRTP packet[6]

SRTP 패킷 중 암호화가 되는 부분은 payload+RTP padding+RTP pad count이다. Counter 모드와 f8 모드는 스트림 모드이기 때문에 별도의 패딩이 필요하지 않지만, 패킷을 32비트 단위로 맞추기 위해서 패딩을 사용한다.

SRTP 패킷 중 태그 생성을 위한 입력값이 되는 부분은 처음부터 RTP pad count 부분까지이다. 즉, 패딩까지 이용하여 태그를 생성함으로써 패딩 오라클 공격을 불가능하게 하였다. 2002년 Vaudenay가 발표한 패딩 오라클 공격을 참고문헌으로 하여 의도적으로 패딩 오라클 공격을 방어한 것이다.

SRTP를 패딩 오라클 공격에 대해 분석한 결과는 Table 2와 같다.

Table 2. Analysis result of SRTP

Operating mode	Counter, f8-mode
Padding method	Optional of user
Encrypted Authentication	HMAC-SHA1
Availability of Padding Oracle Attack	Disable

4.2 MIKEY

MIKEY(Multimedia Internet KEYing)는 SRTP에서 오디오, 비디오와 같은 멀티미디어 데이터 암호화에 사용될 키를 교환하는 키 교환 메커니즘이다.

MIKEY에서는 키를 교환하는 방법으로, 사전 공유키 기반 공유방법(Pre-shared Key), 공개키 암호 기반 공유방법(Public-key encryption), 그리고 Diffie-Hellman 기반 공유방법(Diffie-Hellman key exchange)을 명시하였다[7]. 이 중 사전 공유키 기반 공유방법에서 전송하는 패킷의 구조는 Fig. 4와 같다.

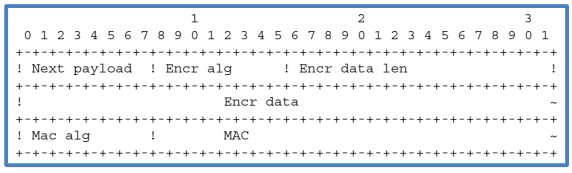

Fig. 4. Structure of the MIKEY packet[7]

MIKEY에서 암호 알고리즘은 AES를 기본(default)으로 사용하며, 운영모드로는 Counter 모드와 Key Wrap 모드[8]를 사용한다. MAC은 HMAC-SHA1을 기본으로 사용한다. MAC의 입력값에는 암호화된 Key data(Encr data)가 포함된다.

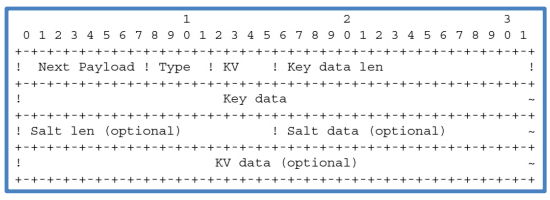

Fig. 5. Structure of Key-data[8]

Key data는 키에 대한 정보를 담고 있으며, 이름 암호화하면 Fig. 4의 Encr data가 된다. 암호화를 하는 데 Counter 모드와 Key Wrap 모드를 사용하기 때문에 패딩은 사용하지 않는다.

MIKEY는 암호화 후 태그를 생성하고, 또한 패딩을 사용하지 않기 때문에 패딩 오라클 공격이 적용되지 않는다. MIKEY를 패딩 오라클 공격에 대해 분석한 결과는 Table 3과 같다.

Table 3. Analysis result of MIKEY

Operating mode	Counter, Key Wrap-mode
Padding method	-
Encrypted Authentication	HMAC-SHA1
Availability of Padding Oracle Attack	Disable

4.3 CMS

CMS(Cryptographic Message Syntax)는 전자 서명, 압축, 인증, 암호문 등에 대한 구문, 즉 데이터 보호를 위한 요약 구문으로, 하나의 요약 구문이 다른 구문을 내포하는 복합 구문이 가능하다[10].

CMS가 처음 제안된 이후로 CBC 운영모드와 CBC-PAD는 새로운 버전으로 업데이트될 때에도 변함없이 사용되고 있다. 최근 CMS의 암호화 인증 적용에 대해 언급한 이후 암호화 인증이 도입되었으나 암호화 인증은 옵션으로 사용 혹은 비사용 되기 때문에 옵션의 적용 여부의 상황에 따라 패딩 오라클 공격이 적용 가능하다.

Table 4. Analysis result of CMS

Operating mode	CBC-mode
Padding method	CBC-PAD
Encrypted Authentication	CCM/GCM (option)
Availability of Padding Oracle Attack	Conditionally available

4.4 IPsec

IPSec은 네트워크상의 두 단말 사이에 안전한 통신을 제공하는 프로토콜의 모음으로, 암호화와 데이터 서명을 이용해 네트워크 트래픽을 보호하여 IP 패킷 보호와 네트워크 공격으로부터의 방어를 위해 사용된다[11].

IPSec v1은 데이터를 보호하기 위해서 DES와 CBC 운영모드를 사용하며, 암호화 인증을 위해 keyed MD5를 사용한다. 그러나 패딩 방법에 대한 언급은 없는 것으로 보아 사용자가 직접 결정하는 것으로 보인다.

IPSec v2는 암호 알고리즘으로 v1과 같이 DES와 CBC 운영모드를 사용하며, 암호화 인증을 위해 HMAC-MD5, HMAC-SHA1을 사용한다. IPSec v3는 암호 알고리즘으로 TripleDES-CBC, AES-CBC를 사용하며, 암호화 인증을 위해서 HMAC-SHA1을 사용한다. 그리고 IPSec v2와 v3의 패딩 방법은 ESP-PAD이다.

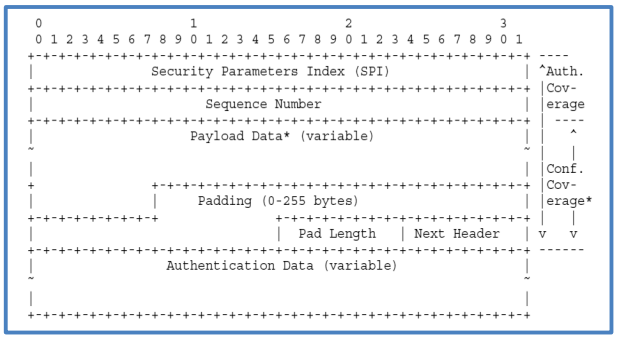

Fig. 6. Structure of ESP packet[11]

Fig. 6에서 볼 수 있듯이, 암호 알고리즘은 Payload Data+Padding+Pad Length+Next Header에 적용된다. 그리고 태그 생성을 위해서 처음부터 Next Header까지 적용한다.

암호화 인증이 패딩 부분까지 적용되지만, 옵션으로 사용 혹은 비사용 되기 때문에 상황에 따라 패딩 오라클 공격이 적용 가능하다.

Table 5. Analysis result of IPSec

Operating mode	CBC-mode
Padding method	ESP-PAD
Encrypted Authentication	HMAC-SHA1(option)
Availability of Padding Oracle Attack	Conditionally available

4.5 TLS

TLS(Transport Layer Security)는 TCP/IP 네트워크를 사용하는 통신에서 전송계층 종단 간 보안과 데이터 무결성을 확보해준다. 또한 이 규약은 웹 브라우징, 전자 메일, instant messaging, voice-over-IP 같은 응용 부분에 적용되고 있다.

1) TLS v1.0

TLS v1.0에서는 데이터를 보호하기 위해서 다양한 블록 암호 알고리즘과 스트림 암호 알고리즘을 지원하고 있다. 블록 암호 알고리즘 사용 시, 긴 데이터를 암호화하기 위해 CBC 운영모드와 CBC-PAD를 사용한다. 또한, 암호화 인증을 위해서 HMAC-MD5, HMAC-SHA1을 지원한다. 본 문서에서는 TLS v1.0 구현을 위한 의무적인(mandatory) 사항으로 TripleDES와 CBC 운영모드, 그리고 HMAC-SHA1을 사용하라고 하고 있다[12].

그러나 TLS v1.0의 암호화와 암호화 인증 과정은 일반적인 경우와 반대로 되어있다.

$$Encrypt(plaintext, padding, MAC(plaintext))$$

암호화한 후 인증 방식(Encrypt-then-Mac)을 적용하는 것이 아니라, 먼저 평문을 이용하여 태그를 생성한 후 암호화(Mac-then-Encrypt)를 적용한다. 그러므로 복호화를 할

때, 먼저 패딩을 체크하고 태그를 인증한다. 즉, 패딩을 체크할 때 오라클이 보내는 VALID/INVALID값을 이용하여 패딩 오라클 공격을 수행할 수 있다.

Table 6. Analysis result of TLS v1.0

Operating mode	CBC-mode
Padding method	CBC-PAD
Encrypted Authentication	HMAC-MD5, HMAC-SHA1
Availability of Padding Oracle Attack	Conditionally available

2) TLS v1.1

TLS v1.1은 패딩 오라클 공격이 소개된 후 업데이트된 버전이다[15]. 그렇기 때문에 패딩 오라클 공격을 방어하기 위한 조치를 취했다.

먼저, IV값을 임의로 변경할 수 없게 변경하였다. 패딩 오라클 공격은 특정 평문 블록을 복구하기 위해서 이전 블록을 임의로 변경하여야 한다. 그러나 IV값을 임의로 변경할 수 없게 되면 첫 번째 평문 블록은 복구할 수 없게 된다.

두 번째로, 패딩을 체크하여 보내는 응답을 태그를 인증할 때 보내는 응답과 동일하게 변경하였다. 패딩 오라클 공격은 오라클이 패딩을 체크할 때 보내는 응답이 있어야 적용 가능하다. 이를 막기 위해서 패딩이 틀리더라도 태그가 틀렸을 때 보내는 응답과 동일하게 하여 공격자로 하여금 알 수 없게 하였다.

세 번째로, 패딩을 체크하여 틀렸어도 프로세스를 멈추지 않고 태그를 인증하게 변경하였다. 패딩 오라클 공격은 오라클이 패딩을 체크할 때와 패딩을 체크하고서 태그를 인증하는 시간의 차이를 이용해서도 적용할 수 있다. 이를 막기 위해 패딩이 틀리나 태그가 틀리나 시간의 차이가 없게 변경하였다.

이와 같은 세 가지 이유로 인해, TLS v1.1은 패딩 오라클 공격이 적용되지 않는다.

Table 7. Analysis result of TLS v1.1

Operating mode	CBC-mode
Padding method	CBC-PAD
Encrypted Authentication	HMAC-MD5, HMAC-SHA1
Availability of Padding Oracle Attack	Disable

3) TLS v1.2

TLS v1.1에서 패딩 오라클 공격을 대응했기 때문에 v1.2에서도 역시 이 공격이 적용되지 않는다. TLS v1.2에서는 안전성을 높이기 위해서 TripleDES 대신 AES 사용을 의무사항으로 지정하였다[13].

TLS v1.2에 대한 분석 결과는 v1.1의 결과와 동일하다.

Table 8. Analysis result of TLS v1.2

Operating mode	CBC-mode
Padding method	CBC-PAD
Encrypted Authentication	HMAC-MD5, HMAC-SHA1
Availability of Padding Oracle Attack	Disable

4.6 IPTV

IPTV(Internet Protocol TeleVision)는 인터넷 망을 통한 양방향 텔레비전 서비스로 여기서 제공되는 콘텐츠를 디지털 콘텐츠라고 한다. 디지털 콘텐츠는 복제가 용이하며 복제품의 품질도 우수하기 때문에 많은 방법으로 복제되고 있다. 또한, 디지털 콘텐츠는 인터넷을 이용하여 쉽고 빠르게 확산되기 때문에 보호 기술이 필요하다. 이를 위해 우리나라에서 사용하도록 권고되어있는 알고리즘이 SEED와 ARIA이다.

IPTV 서비스 보호를 위해 스크램블링 알고리즘을 사용하는데, 스크램블링이란, 유/무선 유료 방송 전송 시 자격 있는 수신자만 정상 화면을 수신할 수 있도록 방송 신호 송출 시 해당 채널에 대해 암호화하는 방식이다. 암호화를 위해 본 표준에서는 암호 알고리즘으로 SEED와 ARIA를 사용하고 있으며, 운영모드로 변형된 CBC 운영모드를 사용한다[14].

변형된 CBC 운영모드에서는 마지막 블록만 기존의 CBC 운영모드와 다르게 스트림 모드로 사용한다. 그렇기 때문에 패딩이 필요 없다. 즉, 패딩 오라클 공격을 적용할 수 없다. IPTV를 패딩 오라클 공격에 대해 분석한 결과는 Table 9와 같다.

Table 9. Analysis result of IPTV

Operating mode	Modified CBC-mode
Padding method	-
Encrypted Authentication	-
Availability of Padding Oracle Attack	Disable

5. 결 론

본 논문에서는 STRP, MIKEY, CMS, IPSec, TLS, IPTV의 명세된 문서들에 대해서 패딩 오라클 공격을 적용하고 분석하였다. 그 결과로 TLS v1.0이 취약한 것으로 나타났고 그 외 CMS나 IPSec에 조건적으로 패딩 오라클 공격이 적용 가능하다는 것을 알 수 있었다. 조건적으로 패딩 오라클 공격이 가능한 경우에는 암호화 인증 기술을 적용하는 경우 공격 적용이 어렵지만, 인증 기술을 사용하지 않는 경우에는 공격 적용이 가능하게 된다. 최근 논문에서 제시한 결과에 따르면 부채널 정보를 이용한 Timing attack 기술을 접

목하면 패딩 오라클 공격 적용이 가능하게 된다.

패딩 오라클 공격은 기본적으로 블록 기반 운용방식에 데이터 블록 크기를 맞추기 위해 적용되는 다양한 패딩 방법의 취약성을 이용한 공격방법이지만 반드시 블록 기반 운영모드에서만 적용되는 것은 아니다. 스트림 모드를 사용하는 서비스에서도 공격자가 패딩 길이나 메시지 길이 정보를 획득할 수 있는 경우 패딩 오라클 공격을 적용할 수 있게 된다. 따라서 서비스 암호화 방식 설계 시에 패딩 길이나 패딩 패턴 등을 노출시키지 않도록 설계하는 것이 매우 중요하다.

References

[1] S. Vaudenay, “Security Flaws Induced by CBC Padding - Applications to SSL, IPSEC, WTLS...,” Eurocrypt 2002, LNCS, Vol.2332, pp.534-545, Springer-Verlag, 2002.

[2] Juliano Rizzo, Thai Duong(2010). “Practical Padding Oracle Attacks,” *USENIX WOOT*, 2010.

[3] M. Baugher, D. McGrew, M. Naslund, E. Carrara, and K. Norrman, “The Secure Real-time Transport Protocol (SRTP),” RFC 3711, Mar., 2004.

[4] M. Dworkin, “Recommendation for Block Cipher Modes of Operation,” NIST 800-38A, 2001.

[5] J-S. Kang, S-U. Shin, D. Hong, and O. Yi, “Provable Security of KASUMI and 3GPP Encryption Mode f8,” ASIACRYPT 2001, LNCS 2248, pp.255-271, Springer-Verlag, 2001.

[6] S. Yoon, J. Kim, H. Park, H. Jeong, and Y. Won, “The SEED Cipher Algorithm and Its Use with the Secure Real-Time Transport Protocol (SRTP),” RFC 5669, Aug., 2010.

[7] J. Arkko, E. Carrar, F. Lindholm, M. Naslund, and K. Norrman, “MIKEY: Multimedia Internet KEYing,” RFC 3830, Aug., 2004.

[8] J. Schaad, R. Housley, “Advanced Encryption Standard(AES) Key Wrap Algorithm,” RFC 3394, Sep., 2002.

[9] J. Jeong, H. Kim, H. Jeong, and Y. Won, “IANA Registry Update for Support of the SEED Cipher Algorithm in Multimedia Internet KEYing (MIKEY),” RFC 5748, Aug., 2010.

[10] J. Park, S. Lee, J. Kim, and J. Lee, “Use of the SEED Encryption Algorithm in Cryptographic Message Syntax (CMS),” RFC 4010. Feb., 2005.

[11] S. Kent, K. Seo, “Security Architecture for the Internet Protocol,” RFC 4301, Dec., 2005.

[12] T. Dierks, C. Allen, “The TLS Protocol Version 1.0,” RFC 2246, Jan., 1999.

[13] T. Dierks, E. Rescorla, “The Transpord Layer Security (TLS) Protocol Version 1.2,” RFC 5246, Aug., 2008.

[14] Korea Association of Information and Communication Technology, “SEED / ARIA scrambling algorithm for IPTV-service Security,” TTAK.KO-12.0123, Dec., 2009.

[15] T. Dierks, E. Rescorla, “The Transport Layer Security (TLS) Protocol Version 1.1,” RFC 4346, Apr., 2006.

황 성 진

e-mail : hgoon6754@gmail.com

2013년 한신대학교 컴퓨터공학과(학사)

2013년~현 재 서울과학기술대학교 컴퓨터공학과 석사과정

관심분야 : 정보보호, 암호알고리즘, 네트워크 보안

박 명 서

e-mail : pms91@kookmin.ac.kr

2013년 국민대학교 수학과(학사)

2013년~현 재 국민대학교 수학과 석사과정

관심분야 : 정보보호, 암호알고리즘, 네트워크 보안

문 덕 재

e-mail : ansejrwo@korea.ac.kr

2000년 서울시립대학교 수학과(학사)

2003년 고려대학교 정보보호학과(석사)

2014년 고려대학교 정보보호학과 박사

2003년~2006년 ETRI 부설 국가보안기술연구소 연구원

관심분야 : 정보보호, 암호알고리즘, 네트워크 보안

강 형 철

e-mail : kanghc@korea.ac.kr

2010년 고려대학교 산업시스템정보공학과(학사)

2010년~현 재 고려대학교 정보보호대학원 석·박사통합과정

관심분야 : 블록 암호와 해시 함수 설계 및 분석, 인증 암호화 설계

김 종 성

e-mail : jskim@kookmin.ac.kr

2000년/2002년 고려대학교 수학(학사/이학석사)

2006년 K.U.Leuven, ESAT/SCD-COSIC 정보보호(공학박사)

2007년 고려대학교 정보보호대학원(공학박사)

2007년~2009년 고려대학교 정보보호기술연구센터 연구교수

2009년~2013년 경남대학교 e-비즈니스학과 조교수

2013년~현 재 국민대학교 수학과 조교수

2014년~현 재 국민대학교 금융정보보안학과 조교수

관심분야 : 정보보호, 암호알고리즘, 디지털 포렌식

이 창 훈

e-mail : chlee@seoultech.ac.kr

2001년 한양대학교 수학과(학사)

2003년 고려대학교 정보보호학과(석사)

2008년 고려대학교 정보보호학과(박사)

2008년 고려대학교 정보보호연구원(연구교수)

2009년~2012년 한신대학교 컴퓨터공학부 조교수

2013년~현 재 서울과학기술대학교 컴퓨터공학과 조교수

관심분야 : 정보보호, 암호학, 디지털 포렌식, 융합보안