정 보 보 호 학 회 지
제 33 권 제1호, 2023. 2

NIST PQC Round 4 코드 기반 암호에 대한 부채널 분석 기법 동향 분석

이 정 환*, 김 규 상**, 김 희 석*

요 약

NIST는 2022년 양자내성암호 표준화 진행 대상 알고리즘으로 KEM 1종(CRYSTALS-Kyber), 전자서명 3종(CRYSTALS-Dilithium, FALCON, SPHINCS+)을 발표하였고, 추가로 KEM 4종(Classic McEliece, HQC, BIKE, SIKE)에 대한 Round 4 진행을 공표하였다. Round 3와 마찬가지로 Round 4에서도 부채널 분석 및 오류 주입에 대한 안전성은 알고리즘 선정에 있어 중요 평가 사항 중 하나이다. 따라서 해당 암호 알고리즘에 대한 새로운 부채널 분석 기술에 대한 연구가 활발히 진행되고 있다. 본 논문은 Round 4의 암호 알고리즘 중 코드 기반 알고리즘 3종(Classic McEliece, HQC, BIKE)에 대한 부채널 분석 방법론의 동향을 파악하고 향후 연구 방향을 제시한다.

Ⅰ. 서 론

1994년 Peter Shor에 의해 소개된 Shor 알고리즘은 양자 컴퓨터에서 인수분해 문제와 이산대수 문제를 다항시간 내에 해결할 수 있다. 따라서 양자 컴퓨터의 발전은 인수분해 문제 또는 이산대수 문제를 기반으로 하는 상용 공개키 암호 시스템에 큰 위협이 되고 있다. 미국 국립표준기술연구소 (NIST)는 2016년 PQCrpyto 컨퍼런스에서 양자 컴퓨터에 안전한 양자내성암호 표준화 사업을 발표하였다. 현재 표준화 진행 대상 알고리즘으로 KEM 1종, 전자서명 3종이 선정되었으며 KEM 4종이 후보 알고리즘으로 선정되어 Round 4를 진행 중이다. 또한, 추가적으로 전자서명에 대한 새로운 알고리즘 제안을 접수 중이다.

부채널 분석은 1996년 P. Kocher에 의해 처음 소개되었다. 수학적 안전성이 증명된 암호가 디바이스 위에서 동작할 때 시간, 소리, 전력, 전자파 등의 부가적인 정보가 누출되고 이것을 이용하여 암호의 비밀값을 복구하는 분석 기법을 부채널 분석이라 한다. 부채널 분석은 공격자 가정 및 분석 방법론에 따라 시간분석, 단순전력분석, 상관전력분석, 프로파일링 공격 등이 존재한다.

이러한 부채널 분석을 대응하기 위한 부채널 분석 대응기술에는 대표적으로 마스킹 기법과 하이딩 기법이 있다. 마스킹 기법은 암호가 동작하기 전 난수를 생성하고 이를 이용하여 암호의 중간값을 무작위하게 보이도록 만드는 대응 기법이다. 마스킹 기법을 사용하면 공격자가 부채널 정보를 통해 중간값을 추론하기 어려워진다. 하이딩 기법은 암호 동작 시 Jitter를 생성하거나 연산 순서를 무작위하게 바꾸어 신호대잡음비(Signal-to-Noise)를 감소시키는 기법이다. 하이딩 기법을 사용하면 부채널 신호와 암호 중간값 간의 관계성이 감소하기 때문에 공격자가 부채널 신호로부터 유의미한 정보를 얻기 어려워진다.

PQC Round 4에 제안된 KEM 4종 중 3종이 코드 기반 암호 알고리즘이며 Round 3과 마찬가지로 Round 4에서도 부채널 분석 및 오류 주입에 대한 안전성은 알고리즘 선정에 있어 중요 평가 사항 중 하나이다. 이에 따라 최근 코드 기반 암호 알고리즘에 대한 부채널 분석 기법이 활발하게 제시되고 있다. 본 논문에서는 NIST PQC Round 4 코드 기반 암호 KEM(Classic McEliece, HQC, BIKE)에 대한 부채널 분석 기법을 조사하여 동향을 파악하고 향후 연구 방향을 제시한다.

본 연구는 과학기술정보통신부 및 정보통신기획평가원의 대학ICT연구센터육성지원사업의 연구결과로 수행되었음(IITP-2023-RS-2022-00164800).

* 고려대학교 인공지능사이버보안학과 (학부생, hwani0814@korea.ac.kr, 부교수, 80khs@korea.ac.kr)
** 고려대학교 정보보호대학원 정보보호학과 (대학원생, ks9509@korea.ac.kr)

본 논문의 구성은 다음과 같다. 2장에서 코드 기반 암호와 부채널 분석 기술 배경지식을 소개한다. 3장에서 Classic McEliece, HQC, BIKE 각각에 대한 최신 부채널 분석 기법을 소개하고 4장에서 향후 연구 방향을 제시하며 결론을 맺는다.

Ⅱ. 배경지식

2.1. 코드 기반 암호

2.1.1. Classsic McEliece

Classic McEliece는 Syndrome Decoding Problem (SDP)을 기반 문제로 하는 공개키 암호화 알고리즘이다. 1978년 이진 Goppa 코드를 사용한 초기 McEliece가 제안되었으며, 1986년 그 dual 버전인 Niederreiter 프레임워크가 소개되었다. 현재 Classic McEliece의 경우 이진 Goppa 코드를 기반으로 한 Niederreiter 프레임워크를 사용하고 있으며 IND-CCA2 안전성를 만족하기 위해 기존 IND-CPA 안전성을 가지는 PKE를 KEM으로 변환하였다[14].

표 1는 Classic McEliece의 파라미터를 나타낸 것이다. 파라미터 m은 이진 Goppa 코드의 유한체 크기를 결정한다. 예를 들어, m이 12인 이진 Goppa 코드는 $\mathbb{F}_{2^{12}}$ 위에서 계산된다. 파라미터 t는 오류 정정이 가능한 비트의 개수를 나타낸 것이며 메시지의 해밍무게 조건과 같다. 파라미터 n은 코드의 길이를 나타낸 것이다.

〔표 1〕 Classic McEliece 파라미터

version.	m	t	n	$k=n-mt$	*level*
348864	12	64	3488	2720	1
460896	13	96	4608	3360	3
6688128	13	128	6688	5024	5
6960119	13	119	6960	5413	5
8192128	13	128	8192	6528	5

2.1.2. HQC

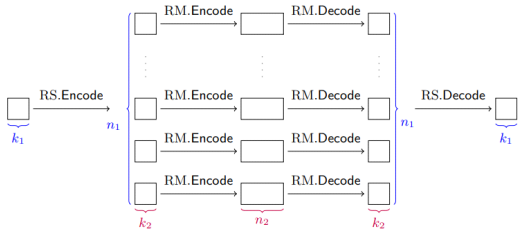
HQC는 quasi-cyclic 코드의 SDP를 기반 문제로 하는 공개키 암호화 알고리즘이다. NIST PQC Round 1에서 제안된 초기 버전의 HQC의 경우 BCH 코드와 repetition 코드를 연접하여 사용하였다. 그러나 키 사이즈를 줄이기 위해 현재 HQC는 Reed-Solomon 코드와 Reed-Muller 코드를 연접하여 사용한다. HQC 또한 IND-CCA2 안전성을 만족하기 위해 IND-CPA 안전성을 가지는 PKE를 KEM으로 변환하였다[15].

표 2는 HQC의 파라미터를 나타낸 것이다. Reed-Muller 코드는 $[n_2,8,n_2/2]$로 정의되며 그림 1와 같이 중첩된 Reed-Muller 코드가 내부에서 인코딩/디코딩을 진행한다. 축소된 Reed-Solomon 코드는 $[n_1,k,n_1-k+1]$로 정의되며 외부에서 인코딩/디코딩을 진행한다. Reed-Muller와 Reed-Solomon에 의해 연접된 코드 C의 길이는 n_1n_2이다. 그러나 연산 과정에서 대수적 공격에 대한 안전성 문제 때문에 n_1n_2보다 큰 첫 번째 소수 n에 대해 $\mathbb{F}_2[X]/(X^n-1)$ 다항식 환을 사용한다.

〔표 2〕 HQC 파라미터

Instance	$RS-S$			$DuplicatedRM$		
	n_1	k	d_{RS}	*Mult*	n_2	d_{RS}
hqc-128	46	16	31	3	384	192
hqc-192	56	24	33	5	640	320
hqc-256	90	32	49	5	640	320

Instance	n_1n_2	n	w	$w_r=w_e$
hqc-128	17,664	17,669	66	75
hqc-192	35,840	35,851	100	114
hqc-256	57,600	57,637	131	149



〔그림 1〕 HQC에서 사용되는 RMRS 인코딩/디코딩 과정

2.1.3. BIKE

BIKE는 quasi-cyclic 코드의 SDP를 기반 문제로 하는 공개키 암호화 알고리즘이다. 큰 구조적으로 볼 때, QC-MDPC 코드를 기반으로 한 Niederreiter 프레

〔표 3〕 BIKE 파라미터

Security	r	w	t	DFR
1	12,323	142	134	2^{-128}
3	24,659	206	199	2^{-192}
5	40,973	274	264	2^{-256}

임워크를 사용하고 있다[16].

표 3은 BIKE의 파라미터를 나타낸 것이다. 인덱스가 2인 QC-MDPC 코드는 두 개의 다항식에 의해 결정되는데 이때 각 다항식은 표 3의 파라미터 r에 따라 $\mathbb{F}_2[X]/(X^r-1)$인 다항식환으로 그리고 $w/2$의 해밍 무게를 가지도록 선택되었다. 파라미터 t는 BIKE에서 사용하는 오류 벡터의 해밍무게를 나타낸다. Decryption Failure Rate(DFR)은 주어진 파라미터 (r,w,t)에 대해 오류 정정에 실패할 확률을 표시한다. BIKE는 IND-CCA 안전성을 만족하기 위해 IND-CPA 안전성을 가지는 PKE를 KEM으로 변환하였다. 그러나 BIKE 디코딩 과정에서 사용하는 Black-Gray-Flip 디코더에 대한 DFR의 상한이 증명되지 않았으므로 해당 디코더에 대한 완전성을 가정해야 IND-CCA를 만족한다.

2.2. 부채널 분석 기법

암호의 이론적 안전성이 증명되었다 하더라도 디바이스 위에서 동작할 때 시간, 전력, 전자파 등의 부가 정보가 누출된다. 이러한 부가 정보들을 부채널 정보라 한다. 부채널 정보는 암호 동작과 긴밀한 연관을 가지고 있는데 이를 분석하여 비밀 정보를 복구하는 분석 기법을 부채널 분석이라 한다.

부채널 분석은 부채널 정보의 종류와 비밀 정보를 얻는 방식에 따라 다양한 분석 기법이 존재하는데 대표적으로 시간분석, 전력분석, 프로파일링 공격 등이 있다. 시간 분석은 암호의 동작 시간이 암호의 비밀 정보에 의존하는 것을 이용하여 비밀정보를 추출하는 분석 기법이다. 전력분석은 해밍무게 또는 해밍거리 모델을 가정하여 암호 동작 시 전력의 세기가 중간값의 해밍무게(또는 해밍거리)에 의존하는 것을 이용한다. 전력분석 방법 중 하나인 단순전력분석은 단일 또는 적은 전력파형을 이용한 부채널 분석 기술이다. 암호 알고리즘에 대한 파형의 특징을 파악하여 비밀 정보를 복구한다. 또 다른 전력분석 방법인 상관전력분석은 다수의 전력파형을 이용한 부채널 분석 기술이다. 암호의 비밀 정보를 추측하여 중간값을 연산하고 파형과 중간값 사이의 상관분석을 통해 비밀 정보를 복구한다.

프로파일링 공격은 공격자가 공격 대상과 동일한 디바이스를 가지고 있어 프로그래밍할 수 있는 환경을 가정한다. 이를 이용하여 공격자는 암호의 공격 대상 시점의 중간값(또는 그 해밍무게)을 라벨로 하는 부채널 파형을 수집할 수 있고 각 라벨별로 많은 파형을 수집하여 특징화할 수 있다. 대표적으로 템플릿 공격과 딥러닝 기반 프로파일링 공격이 존재한다. 템플릿 공격은 부채널 파형을 특징화 할 때 다변수 정규분포를 이용한다. 공격 단계 시 공격 대상에서 추출한 파형과 템플릿을 비교하여 가장 가까운 템플릿의 라벨값을 중간값으로 추정하고 이를 통해 비밀 정보를 복구한다. 딥러닝 기반 프로파일링 공격은 부채널 정보를 특징화할 때 MLP, CNN 등의 딥러닝 네트워크를 이용한다. 공격 단계 시 공격 대상에서 추출한 파형을 학습된 딥러닝 네트워크에 입력하여 확률이 가장 높은 라벨값을 중간값으로 추정하고 이를 통해 비밀 정보를 복구한다.

Ⅲ. NIST PQC Round 4 코드 기반 암호에 대한 부채널 분석

3.1. Classic McEliece에 대한 부채널 공격

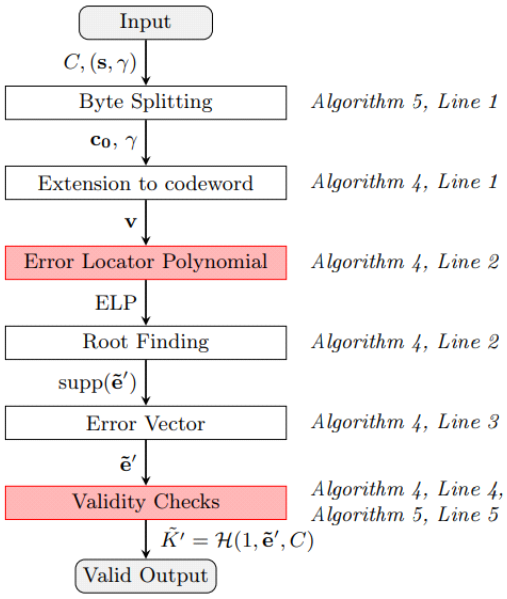
2020년에 Classic McEliece KEM 디캡슐화 과정에 대한 전자파 파형을 이용한 템플릿 공격이 연구되었다[1]. 이 공격을 통해 Classic Mceliece의 메시지를 복구할 수 있다. 공격자에게 신드롬 s가 주어졌을 때 공개키인 parity-check 행렬의 i열을 s에 더하여 i번째 비트가 반전된 메시지 e에 해당하는 신드롬 s'을 생성할 수 있다. 공격자는 s'을 입력으로 디캡슐화 오라클을 동작하고 Berlekamp-Massey(BM) 알고리즘에서 전자파 파형을 수집한다. 이때 t 검정을 통해 메시지 e의 오류가 증가한 파형과 감소한 파형이 구분 가능하다. 공격자는 이를 이용하여 메시지 e를 복구한다. [1]에서는 iterative chunking 알고리즘과 information set decoding (ISD)를 통해 메시지 e 복구를 위한 오라클 동작 횟수를 줄였다.

2021년에 Classic McEliece 암호화 과정에 대한 레이저 오류주입 메시지 복구 공격이 연구되었다[2]. ARMv7-M의 Thumb 명령어 집합에서 XOR 연산

input : Classic McEliece parameters $n, m, t \in \mathbb{N}^+$,
binary parity-check matrix $H'' = [\mathbb{I}_{mt}|K] := (h_{i,j}) \in \mathrm{GF}(2)^{mt \times n}$,
syndrome $s \in \mathrm{GF}(2)^{mt}$.
output: Error vector $e \in \mathrm{GF}(2)^n$.

$e \leftarrow (0, \ldots, 0)$;
for $i \leftarrow 0$ **to** $n-1$ **do**
  $s' \leftarrow s \oplus H''[i]$;
  if Oracle$(s') = true$ **then** $e[i] \leftarrow 1$;
end
return e;

〔그림 2〕 〔3〕의 공격 절차



〔그림 3〕 오류 주입 Classic McEliece 디캡슐화 과정

opcode와 add-and-carry 연산 opcode는 한 비트 차이이다. 공격자는 이를 이용하여 플래시 메모리 뒷면에 레이저 오류를 주입하여 신드롬 행렬 연산을 진행할 때 XOR 연산을 add-and-carry 연산으로 바꿀 수 있다. 이를 통해 신드롬 계산을 $\mathbb{F}_2$가 아닌 $\mathbb{N}$ 상에서 진행시킨다. 마지막으로 주어진 공개키 H와 $\mathbb{N}$에서 복구된 신드롬에 대해 $\mathbb{N}-SDP$ 문제를 구성하여 Integer Linear Programming (ILP) solver를 통해 메시지 e를 복구한다.

2022년에 Classic McEliece 암호화 과정에 대한 전력파형을 이용한 템플릿 공격이 연구되었다[3]. 이 공격은 Classic Mceliece의 메시지를 복구한다. 공격자는 사전에 신드롬 행렬 연산에 대한 전력파형을 수집하여 해밍무게 템플릿을 생성한다. 공격 단계시, 신드롬 행렬 연산의 중간값에 대한 전력파형을 수집하여 가장 가까운 템플릿의 해밍무게로 신드롬의 중간값에 대한 해밍무게를 복구한다. 공격자는 이 중간값의 해밍무게를 이용하여 신드롬을 $\mathbb{N}$ 상에서 복구할 수 있다. 따라서 주어진 공개키 H와 $\mathbb{N}$ 신드롬에 대해 [2]과 마찬가지로 $\mathbb{N}-SDP$ 문제를 구성하는 것이 가능하다. 다만, 신드롬 오류에 민감한 ILP solver를 대체하여 내적과 ISD를 통해 메시지 e를 복구한다.

2022년에 Classic McEliece KEM 디캡슐화 과정에 대한 전력 파형을 이용한 딥러닝 기반 프로파일링 공격이 연구되었다[4]. 이 공격을 통해 Classic McEliece KEM의 개인키를 복구할 수 있다. Classic McEliece KEM은 디캡슐화 과정 중 PKE 복호화 과정에서 Error-Locator Polynomial(ELP)가 0이 되게 하는 값의 집합 $\{\alpha_1,\ldots\alpha_n\}$를 찾기 위해 additive FFT를 계산한다. 따라서 $\{\alpha_1,\ldots\alpha_n\}$에 따라 additive FFT의 전력파형 양상이 바뀌게 된다. 특히, 메시지의 해밍무게가 1일 경우 ELP는 일차식이 되고 α_i값 한 개에 의존하여 additive FFT 전력파형이 결정된다. $\alpha_i \in \mathbb{F}_{2^m}$이므로 공격자는 사전에 $p \in \mathbb{F}_{2^m}$인 ELP $\sigma(x) = x - p$를 구성하여 2^m 종류의 additive FFT 파형을 생성한다. 그런 다음 $p \in \mathbb{F}_{2^m}$을 라벨로 이에 해당하는 파형이 p로 분류되도록 딥러닝 네트워크를 학습시킨다. 공격 단계시, 공격자는 $supp(e_i) = \{1\}$를 만족하는 e_i에 해당하는 암호문을 생성한다. 이 암호문을 입력으로 디캡슐화 과정을 동작시켜 additive FFT 전력파형을 얻는다. 공격자는 해당 파형을 딥러닝 네트워크에 입력하여 가장 확률이 높은 라벨값을 개인키 α_i라고 판단한다. 이 과정을 $i \in \{1,\ldots,n\}$에 대해 진행하여 개인키 $\{\alpha_1,\ldots\alpha_n\}$를 복구한다. 또한 이진 Goppa 코드가 $\Sigma^{n_{i=1}} \frac{c_i}{x-\alpha_i} \equiv 0 \bmod g(x)$를 만족하는 것을 이용하여 유효 암호문 생성을 통해 다항식 $c(x) = \Sigma^{n_{i=1}} \frac{c_i}{x-\alpha_i} \Pi_{j \in I(c)}(x-\alpha_j)$을 구성하고 인수분해하여 $g(x)$를 복구한다.

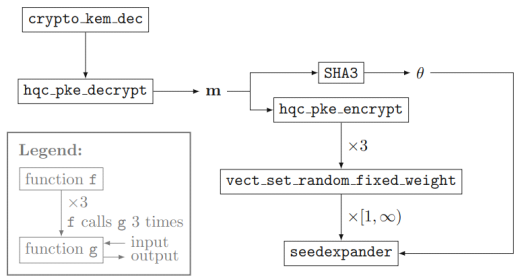
2021년에 Classic McEliece KEM의 디캡슐화 과정에 대한 오류 주입 공격이 연구되었다[5]. 이 공격은 같은 공유키를 생성할 수 있게 하는 곧, $\Gamma(g,\alpha) = \Gamma(\tilde{g},\tilde{\alpha})$인 대체 개인키를 복구한다. 먼저 공격자는 해밍무게가 2인 메시지 e에 해당하는 신드롬 s를 공개키를 통해 생성한다. 이 s를 입력으로 디캡슐화 오라클을 동작시킨다. 계속해서 Error-Locator

Polynomial (ELP)를 계산하는 과정에서 상수항 또는 일차항이 계산될 때 오류를 주입하여 계수가 하나 다른 오류 ELP를 유도한다. 또한 디캡슐화 과정 중 유효 암호문을 확인하는 비교 연산에 대해 오류를 주입하여 디캡슐화 오라클이 오류 공유키를 출력하도록 한다. 선택한 메시지의 해밍무게가 2이므로 전수 조사를 통해 오류 공유키에 해당하는 오류 메시지 복구가 가능하다. 공격자는 메시지에 해당하는 ELP와 오류 메시지에 해당하는 ELP의 관계를 이용하여 개인키 $\alpha_{i_1},\alpha_{i_2},\alpha_{i_3},\alpha_{i_4}$를 해로 하는 선형 또는 2차 다항식 $f_i \in \mathbb{F}_{2^m}[x_1,x_{2,\ldots,}x_n]$을 생성할 수 있다. 이 과정을 반복하여 개인키를 해로 하는 방정식 시스템을 만들 수 있고 이 시스템을 풀어 개인키 후보 $\widetilde{\alpha_i}$를 복구할 수 있다. 마지막으로 Goppa 다항식 g가 $g|\Sigma_{i\in supp(c)}\prod_{j\in supp(c)\setminus\{i\}}(x-\alpha_j)$를 만족함을 이용하여 $\Gamma(\widetilde{\alpha},\widetilde{g})=\Gamma(\alpha,g)$를 만족하는 기약 다항식 $\widetilde{g}$를 생성한다.

3.2. HQC에 대한 부채널 공격

2022년에 시간분석을 이용한 HQC KEM 디캡슐화 과정에 대한 선택 암호문 공격이 연구되었다[6]. HQC KEM 디캡슐화에서 암호문이 입력되면 PKE 복호화 과정을 거쳐 복구된 메시지가 생성된다. 이 메시지를 해시함수에 입력하여 난수 시드 θ를 만들고 시드 θ를 통해 PKE 암호화에 난수를 생성한다. 다시 말해, PKE 암호화에서 사용하는 난수값은 복호화된 메시지에 의존한다. 또한 PKE 암호화 과정에서 시드 θ를 통해 r_1, r_2,e 생성을 진행할 때 r_r,r_2는 해밍무게가 HQC 파라미터 w_r이 되도록, e는 해밍무게가 파라미터 w_e가 되도록 생성한다. 이러한 조건을 만족하기 위해 HQC에는 PKE 암호화 과정 중 vect_set_random_fixed_weight 함수가 구현되어있다. 이 함수는 입력으로 해밍무게를 받고 해당하는 해밍무게를 난수를 출력한다. 특정 해밍무게의 난수를 생성하기 위해 먼저 seedexpander 호출하여 난수값을 생성하고 이 값을 이용하여 반복문을 돌 때마다 $\{0,\ldots,\mathrm{n}-1\}$ 비트 중 하나의 비트를 선택한다. 선택한 비트가 1일 경우 아무것도 하지 않고 0일 경우 1로 채운다. 이 과정을 입력값으로 받은 해밍무게만큼 채워질 때까지 반복한다. 이때, 반복문의 횟수에 따라 HQC KEM 디캡슐화 과정의 동작 시간 차이가 발생하게된다. 정리하면, HQC KEM 디캡슐화 과정 중 PKE 복호화 과정을 통해 복구된 메시지가 해시 함수를 거쳐 시드 θ를 생성하고 이 시드값을 통해 PKE 암호화 과정의 난수가 결정되는데, 이 난수를 생성하는 함수의 반복문의 횟수 또한 시드 θ값에 의존하게 되어 HQC KEM 디캡슐화 동작 시간의 차이가 발생하게 된다 [6]의 공격자는 이러한 특징을 활용하여 암호문의 해밍무게가 1 증가할 때 동작 시간 변화가 큰 암호문을 생성할 수 있고 이를 활용하여 비밀키를 복구할 수 있다.

2022년에 HQC KEM 디캡슐화 과정에 쓰이는 Reed-Solomon 디코딩 알고리즘에 대한 horizontal 공격이 연구되었다[7]. 이 공격을 통해 HQC KEM의 공유키를 알아낼 수 있다. 공격자는 그림 5와 같이 신드롬 계산 과정 중 GF_MUL에 대한 부채널 파형을 수집한다. GF-MUL은 Reed-Solomon의 입력값 $r[j]$과 파라미터 α_i^{j-1}으로 유한체 곱셈 연산을 진행하는데 같은 $r[j]$에 대하여 다른 α_i^{j-1}로 2δ번 연산을 수행함을 알 수 있다. 따라서 부채널 파형 한 개 당 2δ개의 부분 파형을 모을 수 있고 이를 활용해 CPA를 진행한 뒤 r값을 복구하여 HQC KEM의 공유키를 추출한다. 이때, Reed-Solomon 디코더의 입력값 즉, Reed-Muller 디코더의 출력값에 오류가 없음을 가정한다.



〔그림 4〕 seedexpander 호출하는 HQC KEM 디캡슐화 과정 모식도

Input: (received message $\mathbf{r}$, param $=(\alpha_i,\delta,n_1)$)
Output: syndrome **synd**

for $i=0\rightarrow 2\times\delta$ **do**
  for $j=1\rightarrow n_1$ **do**
    $\mathbf{synd}[i]=\mathbf{synd}[i]\oplus \mathrm{GF\_MUL}(\mathbf{r}[j],\alpha_i^{j-1})$
  $\mathbf{synd}[i]=\mathbf{synd}[i]\oplus\mathbf{r}[0]$
return synd

〔그림 5〕 Reed-Solomon 신드롬 계산 과정

2022년에 전력 파형을 이용한 HQC KEM 디캡슐화 과정에 대한 템플릿 공격이 연구되었다[8][9]. 이 공격을 통해 HQC KEM의 개인키를 복구할 수 있다. 디캡슐화 과정의 입력값인 암호문 $c=(u,v)$는 개인키 y와 함께 PKE 복호화 과정에서 $v-uy$로 계산되어 RMRS 디코더에 입력된다. 공격자는 $v-uy=v-y$를 만족하도록 $u=(1,0,0,..,0)\in\mathbb{F}_2^n$로 설정하고 v를 0부터 비트 수를 늘려가며 디캡슐화 과정을 계속 동작시킨다. 이때 $v-y$의 해밍무게가 RM 디코더의 오류 정정 한계를 넘으면 RS 디코더가 해당 오류를 정정하게 되고 이로인해 RS 디코더의 전력 파형 양상이 바뀌게 된다. 공격자는 사전에 RS 디코더에 대한 부채널 전력 파형 템플릿을 만들어 RS 디코더 입력에 오류가 존재하는지 파악할 수 있다. 공격자는 $v-y$가 RM 디코더의 오류 정정 한계가 되도록 v를 선택하고. 해당 v를 1비트씩 바꾸어가며 부채널 파형을 통해 RM 디코더가 오류 정정 한계를 넘겼는지 파악한다. 이를 통해 y의 각 비트값을 구하여 y를 전체복구할 수 있다.

2022년에 전력 파형을 이용한 HQC KEM 디캡슐화 과정에 대한 템플릿 공격이 연구되었다[10]. 이 공격을 통해 HQC KEM의 개인키를 복구할 수 있다. HQC는 연접코드를 사용하므로 RM 디코더는 $v-y\in\mathbb{F}_2^{n_1n_2}$를 $(v-y)_i\in\mathbb{F}_2^{n_2}$, $i\in\{0,...,\mathrm{n}_1-1\}$ 블록 단위로 디코딩한다. [10]은 해당 블록 단위 크기에 해당하는 개인키 블록 y_i의 해밍무게가 k일 확률을 계산하였고 대부분의 경우 해밍무게가 5 이하임을 확인하였다. 결과는 그림 6과 같다. 이를 이용하여 공격자는 사전에 RM 디코딩 과정 중 Hadamard Transform 연산(FHT)에 대해 $(v-y)_i$의 해밍무게 $k\in\{0,1,...,5\}$ 템플릿을 생성한다. 그리고 난 뒤, 암호문 $c=(u,v)$을 $u=(1,0,0,...,0)\in\mathbb{F}_2^n$, $v=(0,...,0)\in\mathbb{F}_2^n$으로 설정하여 HQC KEM 디캡슐화에 입력하여 FHT의 전력파형을 생성해 사전에 생성한 템플릿에 매칭시킨다. 이를 통해 y_i의 해밍무게를 알아낼 수 있다. 계속해서 공격자는 암호문 u는 그대로, v은 해밍무게가 1이 되도록 1 비트만을 수정해주고 입력하여 위와 같은 방식으로 $(v-y)_i$의 해밍무게를 구한다. 이때 $(v-y)_i$의 해밍무게에서 y_i의 해밍무게를 뺀 값이 양수면 y의 해당 비트가 0, 음수면 1임을 의미한다. 이를 통해 개인키 y를 복구할 수 있다.

λ	mul.	n_1	n_2	ω	P_0	P_1	P_2	P_3	P_4	$P_{\geq 5}$
128	3	46	384	66	23, 44%	34, 38%	24, 83%	11, 77%	4, 12%	1, 45%
192	5	56	640	100	16, 50%	30, 00%	27, 00%	16, 04%	7, 07%	3.40%
256	5	90	640	131	23, 14%	34, 06%	24, 87%	12, 02%	4, 32%	1, 59%

〔그림 6〕 HQC의 파라미터에 따른 개인키 y의 한 블록이 해밍무게 k를 가질 확률 P_k

3.3. BIKE에 대한 부채널 공격

2019년에 QC-MDPC 디코딩 과정의 신드롬 계산에 대한 전력분석이 연구되었다[11]. 이 공격을 통해 parity-check 행렬을 복구할 수 있다. 그런데 BIKE KEM은 $h_0, h_1\in\mathbb{F}_2[X]/(X^r-1)$을 개인키로 생성하므로 KEM이 한번 동작할 때마다 parity-check 행렬이 바뀐다. 따라서 [11]의 논문 중 단일 파형 분석만 BIKE KEM 개인키 복구 공격에 적용 가능하다.

그림 7의 디캡슐화 과정 중 디코더에 입력하는 c_0h_0 곱셈에 대한 단일 파형 분석을 통해 h_0를 복구한다. 먼저 h_0 다항식을 표현한 순환 행렬을 H_0하고 c_0을 표현한 벡터를 $c\in\mathbb{F}_2^r$이라 하자. 그러면 다항식 곱셈 c_0h_0을 $H_0c^T=\Sigma_{i\in I_0}R_i(c)^T$로 표현 가능하다. 이때 $R_i(c)$는 벡터 c를 왼쪽으로 i비트만큼 회전한 벡

Decaps : $(h_0,h_1,\sigma),c\mapsto K$
Input: $((h_0,h_1),\sigma)\in\mathcal{H}_w\times\mathcal{M}$, $c=(c_0,c_1)\in\mathcal{R}\times\mathcal{M}$
Output: $K\in\mathcal{K}$
1: $e'\leftarrow\mathtt{decoder}(c_0h_0,h_0,h_1)$ ▷ $e'\in\mathcal{R}^2\cup\{\perp\}$
2: $m'\leftarrow c_1\oplus\mathbf{L}(e')$ ▷ with the convention $\perp=(0,0)$
3: **if** $e'=\mathbf{H}(m')$ **then** $K\leftarrow\mathbf{K}(m',c)$ **else** $K\leftarrow\mathbf{K}(\sigma,c)$

〔그림 7〕 BIKE KEM 디캡슐화 과정

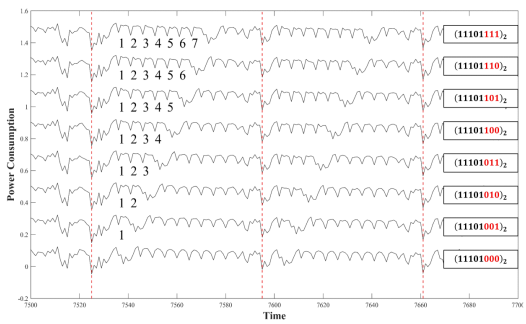
Input : $d=(d_{l-1},\cdots,d_0)_2$, $0\leq d\leq r-1$, $c_{(k)}=(c_{L-1},\cdots,c_0)_{2^w}$, $L=\lceil r/W\rceil$
Output : $x^dc_{(k)}$
1: $v\leftarrow 0$, $w\leftarrow c_{(k)}$
2: **for** $i=l-1$ down to $\log_2W$ **do** ▶ **word unit rotation** is from 2 to 13
3: $d_i\leftarrow(d\gg(l-1-i))\ \&\ 1$
4: $mask\leftarrow 0-d_i$
5: $us\leftarrow 1\ll(i-\log_2W)$
6: $ptr\leftarrow v$, $v\leftarrow w$, $w\leftarrow ptr$
7: **for** $j=0$ up to $L-1-us$ **do**
8: $w[j]\leftarrow(v[j+us]\ \&\ mask)\oplus(v[j]\ \&\ \neg mask)$
9: **end for**
10: **for** $j=1$ up to us **do**
11: $w[j+L-1-us]\leftarrow(v[j-1]\ \&\ mask)\oplus(v[j+L-1-us]\ \&\ \neg mask)$
12: **end for**
13: **end for**
14: $low\leftarrow d\ \&\ ((1\ll\log_2W)-1)$ ▶ **bit rotation** is from 14 to 22
15: $high\leftarrow W-low$
16: $tmp\leftarrow w[0]$
17: **for** $j=0$ up to $L-2$ **do**
18: $w[j]\leftarrow w[j]\gg low$
19: $w[j]\leftarrow w[j]\mid(w[j+1]\ll high)$
20: **end for**
21: $w[L-1]\leftarrow w[L-1]\gg low$
22: $w[L-1]\leftarrow w[L-1]\mid(tmp\ll high)$
23: **Return** w

〔그림 8〕 $\mathbb{F}_2[X]/(X^r-1)$ 다항식 간 곱셈의 상수 시간 알고리즘

터이다. $d=r-i$에 대해 다항식 $x^d c_0$로 다시 표현해 줄 수 있다. 그림 8은 이러한 d값과 다항식 c_0를 입력으로 $x^d c_0$을 출력하는 상수 시간 알고리즘이다. 이 알고리즘은 Word-Unit Rotation과 Bit Rotation으로 반복문이 나뉜다. Word-Unit Rotation의 경우 d_i값에 따라 $mask$ 값이 결정되며 이는 파형의 양상을 바꾼다. 알고리즘에 대한 파형이 주어졌을 때, 공격자는 k-means를 통해 모든 d_i 비트에 대한 파형의 Point of Interest(PoI)를 0과 1 두 군집으로 분류할 수 있다. 이를 통해 Word-Unit Rotation에 사용되는 d_i 비트를 복구할 수 있다. Bit rotation의 경우 $(d_{\log_2 W} \cdots d_0)$의 값에 따라 파형 모양이 그림 9과 같이 달라진다. 이를 통해 $(d_{\log_2 W} \cdots d_0)$ 복구가 가능하다. 공격자는 d값을 모두 복구하였으므로 이를 통해 개인키 h_0를 추출할 수 있다.

2022년에 동작 시간 분석을 이용한 BIKE KEM 디캡슐화 과정에 대한 선택 암호문 공격이 연구되었다 [12]. 이 공격은 [6]의 공격 논리를 BIKE KEM에 그대로 확장한 방법론이다. 따라서 [6]과 마찬가지로 이 공격을 통해 개인키 h_0를 복구할 수 있다.

BIKE KEM 디캡슐화 과정에는 암호문 유효성을 검증하기 위해 복호화된 메시지 m'을 해시 함수 H에 입력하여 디코딩된 e'과 비교하는 과정이 존재한다. 이때 사용되는 해시 함수 H는 해밍무게가 t인 값을 출력하도록 설계되어있는데 이를 구현하기 위해 Rejection Sampling이 사용된다. 따라서 복호화된 메시지에 따라 H의 동작 시간이 달라지고 이는 디캡슐화 과정 전체 동작 시간의 차이를 만든다. 한편, [13]의 공격에서는 IND-CPA인 BIKE 복호화 과정에 대해 메시지 해시값의 일부인 e_0의 support 집합 원소 간의 거리와 개인키 h_0의 support 집합 원소 간의 거리가 일치할수록 디코딩 실패 확률이 낮아진다는 통계적인 관계성을 이용하여 공격자가 e_0의 support 집합 원소 간의 거리를 조절해가며 디코딩 과정에 오류가 발생하는지 확인하고 이를 이용해 h_0의 support 집합 원소 간 거리 스펙트럼을 복구해 h_0값을 추출하였다. 이를 GJS 공격이라 부른다.



〔그림 9〕 Bit Rotation에 대한 단순파형분석

[12]의 공격자는 BIKE KEM의 동작 시간 특성을 이용하여 메시지 1비트가 바뀌면 동작 시간이 크게 차이나는 메시지 m을 선택한다. 또한 e_1를 영벡터로 e_0를 오류 정정 한계에 가깝도록 구성하여 GJS 공격을 진행한다. [13]에서는 디코딩 오류를 직접적으로 확인할 수 있었던 반면, [12]에서는 암호문 유효성 검사 때문에 디코딩 오류의 직접적인 확인이 불가능하다. 따라서 디캡슐화 과정에 대한 동작 시간 차이를 통해 디코딩 오류를 확인한다.

Ⅳ. 결 론

본 논문은 NIST PQC Round 4 코드 기반 암호 KEM 3종에 대한 부채널 분석 기법 동향을 조사하였다. 코드 기반 암호에 대한 다수의 부채널 공격은 디코더의 오류 정정 한계를 부채널 정보로 파악하여 이용하는 오라클 기반 공격이다. 공격자는 디코더 오류 정정 한계에 가까운 암호문을 선택한 뒤 해당 암호문을 한 비트씩 조작하여 디코딩 실패를 유도하고 이를 부채널 파형으로 확인하여 비밀키를 복구할 수 있다. 이를 대비하기 위해서는 마스킹 기법 등과 같이 암호 연산 전체에 대한 대응기술 적용이 이루어져야 한다.

현재까지는 코드 기반 암호에 대한 부채널 분석 대응 기법 연구가 미비한 상황이다. 그러나 코드 기반 암호가 표준 암호로 선정되기 위해서는 부채널 분석에 대한 안전성은 필수적이므로 앞으로 이에 대한 부채널 대응 기술 연구가 활발히 진행될 것으로 예상한다.

참 고 문 헌

[1] N. Lahr, R. Niederhagen, R. Petri, S. Samardjiska, "Side Channel Information Set Decoding Using Iterative Chunking", Advances in Cryptology – ASIACRYPT 2020, 2020, Volume 12491

[2] P.-L. Cayrel, B. Colombier, V.-F. Dragoi, A.

Menu, and L. Bossuet, "Message-recovery Laser Fault Injection Attack on the Classic McEliece Cryptosystem", Advances in Cryptology – EUROCRYPT 2021: 40th Annual International Conference on the Theory and Applications of Cryptographic Techniques, Zagreb, Croatia, October 17 – 21, 2021, Proceedings, Part II, Pages 438 – 467

[3] B. Colombier, V. -F. Drăgoi, P. -L. Cayrel and V. Grosso, "Profiled Side-Channel Attack on Cryptosystems Based on the Binary Syndrome Decoding Problem," in IEEE Transactions on Information Forensics and Security, vol. 17, pp. 3407-3420, 2022, doi: 10.1109/TIFS.2022.3198277.

[4] Q. Guo, A. Johansson, and T. Johansson, “A Key-Recovery Side-Channel Attack on Classic McEliece Implementations”, TCHES, vol. 2022, no. 4, pp. 800 – 827, Aug. 2022.

[5] P.-L. Cayrel, B. Colombier, V. Dragoi, A. Menu, L. Bossuet. “Message-Recovery Laser Fault Injection Attack on the Classic McEliece Cryptosystem.” International Conference on the Theory and Application of Cryptographic Techniques (2021).

[6] Schröder, R. L. , "A novel timing Side-Channel assisted key-recovery attack against HQC", Diploma Thesis, Technische Universität Wien; Technische Universität Darmstadt, reposiTUm

[7] G. Goy, A. Loiseau, P. Gaborit, "Estimating the Strength of Horizontal Correlation Attacks in the Hamming Weight Leakage Model: A Side-Channel Analysis on HQC KEM", wcc2022.uni-rostock.de

[8] T. Schamberger, J. Renner, G. Sigl, A. Wachter-Zeh, "A Power Side-Channel Attack on the CCA2-Secure HQC KEM", Smart Card Research and Advanced Applications, 2021, Volume 12609

[9] T. Schamberger, L. Holzbaur, J. Renner, A. Wachter-Zeh, Georg Sigl, "A Power Side-Channel Attack on the Reed-Muller Reed-Solomon Version of the HQC Cryptosystem", PQCrypto 2022: Post-Quantum Cryptography pp 327 – 352

[10] G. Goy, A. Loiseau, P. Gaborit, "A New Key Recovery Side-Channel Attack on HQC with Chosen Ciphertext", PQCrypto 2022: Post-Quantum Cryptography pp 353 – 371

[11] B.-Y. Sim, J. Kwon, K. Y. Choi, J. Cho, A. Park, and D.-G. Han, “Novel Side-Channel Attacks on Quasi-Cyclic Code-Based Cryptography”, TCHES, vol. 2019, no. 4, pp. 180 – 212, Aug. 2019.

[12] Q. Guo, C. Hlauschek, T. Johansson, N. Lahr, A. Nilsson, and R. L. Schröder, “Don’t Reject This: Key-Recovery Timing Attacks Due to Rejection-Sampling in HQC and BIKE”, TCHES, vol. 2022, no. 3, pp. 223 – 263, Jun. 2022.

[13] Q. Guo, T. Johansson,, P. Stankovski, “A key recovery attack on MDPC with CCA security using decoding errors”, In Jung Hee Cheon and Tsuyoshi Takagi, editors, ASIACRYPT 2016, Part I, volume 10031 of LNCS, pages 789 – 815. Springer, Heidelberg, December 2016.

[14] Bernstein, D.J, et al, Nist post-quantum cryptography standardization round 4 submission: Classic McEliece

[15] Melchor, C.A., et al, Nist post-quantum cryptography standardization round 4 submission: Hamming Quasi-Cyclic (HQC)

[16] N. Aragon, P. Barreto et al., Nist post-quantum cryptography standardization round 4 submission: Bit Flipping Key Encapsulation (BIKE)

〈저 자 소 개〉

이 정 환 (JeongHwan Lee)

2017년~현재 : 고려대학교 과학기술대학 인공지능사이버보안학과 학사과정

<관심분야> 부채널 공격, 부채널 대응, 공개키 암호

김 규 상 (GyuSang Kim)

2020년 2월 : 연세대학교 수학과 학사

2020년 9월~현재 : 고려대학교 정보보호학과 석박사 통합 과정

<관심분야> 공개키 암호 부채널 공격

김 희 석 (HeeSeok Kim)

종신회원

2006년 : 연세대학교 수학과 학사

2008년 : 고려대학교 정보보호대학원 석사

2011년 : 고려대학교 정보보호대학원 박사

2011년 9월~2012년 12월 : Bristol University 박사후 연구원

2013년~2016년 8월 : 한국과학기술정보연구원(KISTI) 선임연구원

2015년~2016년 8월 : 과학기술연합대학원대학교(UST) 조교수

2016년 9월~현재 : 고려대학교 과학기술대학 인공지능사이버보안학과 부교수

<관심분야> 부채널 공격, 암호 시스템 안전성 분석 및 고속구현, 암호 칩 설계 기술, 보안관제, 네트워크 보안