Journal of The Korea Institute of Information Security & Cryptology ISSN 1598-3986(Print)
VOL.31, NO.6, Dec. 2021 ISSN 2288-2715(Online)
https://doi.org/10.13089/JKIISC.2021.31.6.1149

Grover 알고리즘 공격 비용 추정을 통한 DES에 대한 양자 암호 분석*

장 경 배,[1†] 김 현 지,[1] 송 경 주,[1] 심 민 주,[1] 엄 시 우,[1] 서 화 정[2‡]
[1,2]한성대학교 (대학원생, 교수)

Quantum Cryptanalysis for DES Through Attack Cost Estimation of Grover's Algorithm*

Kyung-bae Jang,[1†] Hyun-Ji Kim,[1] Gyeong-Ju Song,[1] Min-Ju Sim,[1]
Eum-Si Woo,[1] Hwa-Jeong Seo[2‡]
[1,2]Hansung University (Graduate student, Professor)

요 약

Brute force 공격을 가속화 시키는 Grover 알고리즘은 대칭키 암호의 키 복구에 적용 가능하며, NIST에서는 대칭키 암호에 대한 Grover 공격 비용을 양자 후 보안 강도 추정에 활용하고 있다. 본 논문에서는 DES를 양자 회로로 최적화 구현하여 Grover 알고리즘 공격 비용을 추정한다. NIST에서는 128, 192, 256-bit 키를 사용하는 대칭키 암호에 대해 AES의 공격 비용을 기준으로 양자 후 보안 강도를 추정하고 있다. DES에 대해 추정한 공격 비용은 DES가 양자 컴퓨터의 공격으로부터 어느 정도의 내성을 가지고 있는지 분석해볼 수 있다. 현재 64-bit 키를 사용하는 대칭키 암호들에 대한 양자 후 보안 지표가 아직 존재하지 않기 때문에 본 논문에서 추정한 64-bit 키를 사용하는 DES에 대한 Grover 공격 비용이 기준으로 활용될 수 있다. 제안하는 DES의 양자 회로 구현 적합성 및 공격 비용 분석에는 양자 프로그래밍 툴인 ProjectQ가 사용되었다.

ABSTRACT

The Grover algorithm, which accelerates the brute force attack, is applicable to key recovery of symmetric key cryptography, and NIST uses the Grover attack cost for symmetric key cryptography to estimate the post-quantum security strength. In this paper, we estimate the attack cost of Grover's algorithm by implementing DES as a quantum circuit. NIST estimates the post-quantum security strength based on the attack cost of AES for symmetric key cryptography using 128, 192, and 256-bit keys. The estimated attack cost for DES can be analyzed to see how resistant DES is to attacks from quantum computers. Currently, since there is no post-quantum security index for symmetric key ciphers using 64-bit keys, the Grover attack cost for DES using 64-bit keys estimated in this paper can be used as a standard. ProjectQ, a quantum programming tool, was used to analyze the suitability and attack cost of the quantum circuit implementation of the proposed DES.

Keywords: Grover algorithm, Symmetric key cryptography, DES, Quantum circuit

Received(10. 29. 2021), Modified(12. 06. 2021), Accepted(12. 06. 2021)

* 이 성과는 부분적으로 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (〈Q|Crypton〉, No.2019-0-00033, 미래컴퓨팅 환경에 대비한 계산 복잡도 기반 암호 안전성 검증 기술개발, 50%) 그리고 이 성과는 부분적으로 2021년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임 (No. NRF-2020R1F1A1048478, 50%).

* 본 논문은 2021년도 한국정보보호학회 충청지부 학술대회에 발표한 우수논문을 개선 및 확장한 것임

† 주저자, starj1023@gmail.com

‡ 교신저자, hwajeong84@gmail.com(Corresponding author)

I. 서 론

양자 알고리즘을 활용하는 양자 컴퓨터는 특정 문제를 해결하는 데 있어 기존 컴퓨터보다 월등한 계산 능력을 보여준다. 대표적으로, Grover 알고리즘은 정렬되지 않은 N개의 데이터 셋에서 특정 데이터를 $O(\sqrt{N})$번 만에 높은 확률로 찾는 양자 알고리즘이다[1]. Grover 알고리즘은 대칭키 암호의 키를 복구하는 전수조사에 사용될 수 있어 대칭키 암호의 보안 레벨을 기존 n-bit에서 $n/2$-bit로 줄일 수 있다. 양자 컴퓨터에서 동작하기 때문에 우선 대상 암호의 암호화가 양자 회로로 구현되어야 한다. 이에, 많이 활용되고 있는 대칭키 암호 AES 양자 회로 구현을 시작으로[2], 다양한 대칭키 암호들을 양자 회로로 최적화 구현하는 연구들이 진행되고 있다[3-7]. 중첩 성질의 큐비트를 사용하는 양자 컴퓨터 상에서 암호화를 수행하면 모든 키 값에 해당하는 암호문들을 한 번에 생성할 수 있다. 하지만 생성된 암호문들은 확률로서 존재하기 때문에 Grover 알고리즘을 사용하여 올바른 암호문인지 비교한 뒤, 해당 암호문을 생성하는 키 값의 관측 확률을 증가시키는 과정이 필요하다.

본 논문에서는 대칭키 암호 DES의 암호화를 효율적으로 구현한다. NIST는 대칭키 암호의 양자 후 보안 강도 추정에 있어 Grover 알고리즘의 공격 비용(양자 게이트 × 회로 depth)을 기준으로 활용하고 있다[8]. 제안하는 DES 양자 회로를 기반으로 Grover 알고리즘 공격 비용을 추정하고 DES의 양자 후 보안 강도를 평가한다. 양자 회로 구현 및 분석에는 양자 프로그래밍 툴인 ProjectQ가 사용되었으며 구현 코드는 GitHub[9]에 공개되어 있다.

II. 관련 연구

2.1 DES

DES는 64-bit 키와 64-bit 평문을 사용하는 대칭키 블록암호 알고리즘이다. 64-bit 키 중 8-bit는 parity bit로 사용된다. Fig.1.과 같은 Feistel 구조를 가지고 있으며 Permutation 연산과 Substitution 연산들이 사용되며 F 함수 내부 연산은 Fig.2.와 같다.

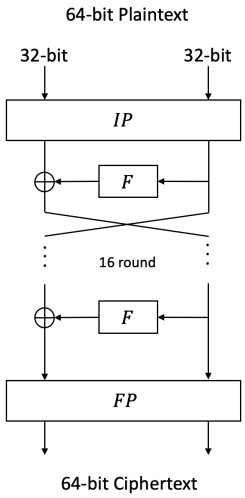


Fig. 1. DES structure

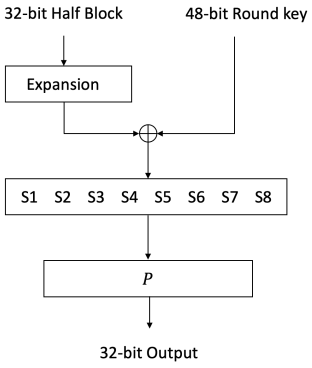


Fig. 2. F function

2.2 양자 컴퓨팅

양자 컴퓨터는 모든 연산들에 대해 가역적인 특징을 가지고 있다. 즉 출력 값들만을 사용하여 다시 초기 상태로 복구할 수 있어야 한다. Fig.3.은 고전 컴퓨터들의 대표적인 논리 게이트들을 대체할 수 있는 가역적 특징의 양자 게이트들이다. 첫 번째 행은 NOT 연산과 XOR 연산을 대체하는 X 게이트와 CNOT 게이트이며, 두 번째 행은 AND 연산과 OR 연산을 대체하는 Toffoli 게이트와 양자 OR 게이트이다. Fig.3.의 양자 게이트들을 활용하여 다양한 암호 알고리즘에 필요한 연산들을 구현할 수 있다.

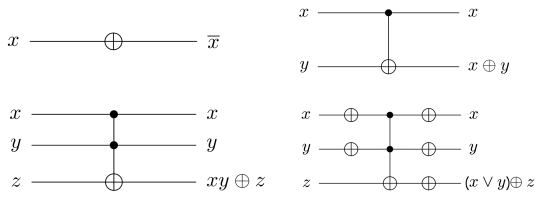


Fig. 3. Quantum gates

2.3 Grover 알고리즘을 활용한 키 복구

대칭키 암호에 대한 brute force 공격은 특정 평문, 암호문 쌍에 대한 비밀 키를 찾는 것이다. n-bit 키를 사용하는 대칭키 암호에 대한 brute force 공격의 경우 $O(2^n)$의 무작위 대입이 필요하다. Grover 알고리즘은 대칭키 암호에 대한 brute force 공격 시 $\sqrt{2^n}$번 만에 높은 확률로 키를 복구하는 양자 알고리즘이다. Grover 알고리즘을 활용한 키 복구 절차는 다음과 같다.

첫 번째, 다음 수식과 같이 Hadmard 게이트를 통과한 중첩 상태($|\psi\rangle$)의 n-qubit 키를 준비한다. 그 결과, n-qubit에는 2^n개의 모든 키 값들이 동일한 amplitude로서 존재하게 된다.

$$|\psi\rangle = H^{\otimes n}\ |0\rangle^{\otimes n} = \left(\frac{|0> + |1>}{\sqrt{2}}\right)^{\otimes n} = \frac{1}{2^{n/2}}\sum_{x=0}^{2^n-1} |x\rangle \quad (1)$$

두 번째, 양자 게이트들로 구현 된 공격 대상 알고리즘의 암호화 양자 회로가 Oracle에 자리한다. Oracle $f(x)$에서는 중첩 상태의 n-qubit 키로 입력 평문을 암호화하여 모든 키 값에 대한 암호문을 생성한다. 이후 알고 있는 암호문과 비교하여 일치하는 경우의 키 값의 부호를 음수로 변경시킨다. $f(x)=1$일 때 부호가 음수로 변경되며 모든 상태에 적용한다.

$$f(x) = \begin{cases} 1 \text{ if } Enc(k) = C \\ 0 \text{ if } Enc(k) \neq C \end{cases} \quad (2)$$

$$U_f(\ |\psi\rangle\ |-\rangle\ ) = \frac{1}{2^{n/2}}\sum_{x=0}^{2^n-1}(-1)^{f(x)}\ |x\rangle\ |-\rangle \quad (3)$$

마지막으로, Diffusion operator가 동작하며 Oracle에서 반환한 키의 amplitude를 증폭시킨다.

Grover 알고리즘은 Oracle과 Diffusion operator를 반복하여 올바른 키의 amplitude를 충분히 높인 뒤 관측한다. [10]에서는 Grover 알고리즘의 상세한 분석을 통해 2^n의 탐색 공간에 최적의 반복 횟수는 $\frac{\pi}{4}\sqrt{2^n}$임을 제시하였다. Grover 알고리즘을 활용한 키 복구에서 가장 많은 비용을 차지하는 부분은 Oracle이다. Oracle에 자리하는 암호화 회로를 얼마나 효율적으로 구현하는지에 따라 최종 비용이 결정된다. Diffusion operator는 정해진 구조를 가지고 있어 특별한 구현 기법이 요구되지 않는다. Grover 알고리즘 양자 회로는 Oracle이 대부분을 차지하기 때문에 비용 추정 시 Diffusion operator는 대부분 무시된다.

III. 제안 기법

제안하는 DES 양자 회로는 최초 구현 결과이며, 본 장에서는 해당 구현 기법에 대해 상세히 설명한다. Permutation 연산인 Initial Permutation과 Final Permutation 구현에는 실제 양자 Swap 게이트를 사용하는 것이 아닌 logical swap을 활용함으로써 양자 자원을 전혀 사용하지 않는다. F 함수를 구현하기 위한 양자 자원만이 사용 되는데, F 함수에서 사용하는 라운드 키 생성을 위한 키 스케줄 또한 logical swap을 활용하여 양자 자원이 소비되지 않는다.

3.1 Logical swap을 활용한 *IP*, *FP* 구현

양자 컴퓨터에서는 두 큐비트간의 상태를 서로 변경하기위해 Swap 게이트가 사용된다. 비트 순서를 변경하는 *IP*, *FP*는 Swap 게이트들로 구현할 수 있지만 본 구현에서는 logical swap을 사용하여 Swap 게이트를 사용하지 않고 공격 비용을 최소화하였다. Logical swap은 큐비트간의 인덱스들을 변경하여 Swap gate를 대체하는 것을 의미한다. 이와 같이 논리적인 연산을 통해 물리적인 연산을 대체할 수 있다. ProjectQ는 파이썬을 기반으로 하는 양자 프로그래밍 툴이며 이를 활용한 *IP* logical swap구현은 Fig.4.에 나타나 있다. Index의 요소들은 *IP* 테이블과 같다. 동일한 방식으로 *FP*의 테

Algorithm : *IP* logical swap

Input: Block $B([0], [1],\cdots, [63])$
Output: Block B_{New}
1: Index = 〔57, 49, 41, 33, 25,...,14, 6〕
2: B_{New} = 〔〕
3: for $i=0$ to 63
4: B_{New}.append(B〔Index〔i〕〕)
5: **return** B_{New}

Fig. 4. *IP* logical swap

이블을 구성하여 Fig.4.와 같이 구현할 수 있기 때문에 *FP* logical swap에 대한 자세한 설명은 생략하도록 한다.

3.2 *F* 함수 양자 회로 구현

F 함수에의 입력 단계에서는 48-bit 라운드 키를 XOR하기 위해 32-bit 중간 값을 48-bit로 확장하는 과정이 존재한다. 확장된 48-bit는 라운드 키와 XOR되고 6-bit씩 8개의 Sbox($S1$, $S2$,...,$S8$)에 입력되어 Sbox 출력 4-bit가 8개인 최종 32-bit를 출력한다. 마지막으로 출력한 32-bit에 Permutation 연산을 수행하고 *F* 함수가 종료된다. *F* 함수의 양자 회로 구현은 Fig.5.와 같다.

DES는 48-bit로 확장하기위해 32-bit 중 16-bit를 temp 값으로 복사하여 사용한다. 제안하는 양자 회로 구현 기법은 큐비트 사용 최적화를 위해 32-qubit을 48-qubit으로 확장하지 않고 on-the fly 방식을 사용한다. 즉 32-qubt을 확장하지 않고, CNOT 게이트를 사용하여 $S1$을 위한 라운드 키만을 XOR한 뒤, $S1$을 수행한다. $S1$에 입력되는 6-qubit에만 라운드 키를 XOR하기 때문에, 6개의 CNOT가 사용된다. $S1$이 끝났다면 수행했던 라운드 키의 XOR 연산을 다시 수행한다(reverse). 그 결과, reverse 연산으로 인해 32-qubit은 다시 초기 상태로 돌아오게 되고 $S2$를 위한 라운드 키만을 XOR한 뒤, $S2$를 수행한다. 이러한 방식으로 라운드 키 XOR + S + reverse 연산을 $S8$까지 반복한다. *F* 함수의 Sbox 연산 후 *IP*와 유사한 Permutation 연산이 수행되는데, *IP*와 동일한 방식으로 Fig.6.과 같이 구현한다.

Algorithm : Quantum circuit design for *F*

Input: 32-qubit Block $B([0], [1],\cdots, [31])$
Output: Block B_{New}
1: CNOT6(RK, ($B[31], B[0], B[1], B[2], B[3], B[4]$))
2: B_{New}〔0:4〕 = $S1(B[31], B[0], B[1], B[2], B[3], B[4])$
// reverse
3: CNOT6(RK, ($B[31], B[0], B[1], B[2], B[3], B[4]$))

4: CNOT6(RK, ($B[3], B[4], B[5], B[6], B[7], B[8]$))
5: B_{New}〔4:8〕 = $S2(B[3], B[4], B[5], B[6], B[7], B[8])$
// reverse
6: CNOT6(RK, ($B[3], B[4], B[5], B[6], B[7], B[8]$))
.
.
.
7: CNOT6(RK, ($B[27],B[28],B[29],B[30],B[31],B[0]$))
8: B_{New}〔28:32〕 = $S8(B[27],B[28],B[29],B[30],B[31],B[0])$
// reverse
9: CNOT6(RK, ($B[27],B[28],B[29],B[30],B[31],B[0]$))
10: **return** B_{New}

Fig. 5. Quantum circuit design for *F*

F 함수가 끝났다면 마지막으로 우측 32-bit를 좌측 32-bit에 XOR하는 연산을 수행하며 이는 32개의 CNOT 게이트를 사용하여 간단하게 구현 가능하다.

Algorithm : Permutation in *F*

Input: Block $B([0], [1],\cdots, [31])$
Output: Block B_{New}
1: Index = 〔16, 7, 20, 21, 29,...,4, 25〕
2: B_{New} = 〔〕
3: for $i=0$ to 31
4: B_{New}.append(B〔Index〔i〕〕)
5: **return** B_{New}

Fig. 6. Permutation in *F*

3.3 Sbox 양자 회로 구현

SPN(Substitution Permutation Network) 구조 블록 암호의 양자 회로 구현 시, 대부분의 경우 Substitution 연산에 가장 많은 자원이 소모된다. 따라서 Sbox 부분이 양자 회로 구현에서 최적화 될 수 있는 구조를 가지고 있다면 전체적으로 낮은 자원이 요구된다.

사전 테이블을 활용한 Sbox 구현은 고전 컴퓨터에서 일반적으로 사용되는 기법이다, 하지만 출력 값만으로 입력 값이 복구되어야 하는 가역적인 양자 컴퓨터에서는 사전 테이블 방식의 Sbox 구현은 매우 비효율적이다. 따라서 입력 값끼리의 연산들을 통해 결과 값을 출력하는 ANF(Algebraic Normal Form)의 Sbox가 구현되어야 한다. 이를 위해 본 논문에서는 M. Kwan이 제시한 bit-sliced DES[11]의 Sbox를 활용한다. 해당 bit-sliced Sbox 구현을 살펴보면, 다수의 XOR, AND, OR 연산들로 구성되어 많은 양자 게이트가 사용되어야 한다. 또한 temp 값을 위한 추가 큐비트들이 할당되어야 한다. 제안하는 DES의 양자 회로의 대부분의 큐비트, 양자 게이트는 Sbox 구현에 사용된다. $S1, S2, \ldots, S8$의 세부적인 양자 게이트 구성은 본 논문에서 생략하도록 하며 [9]에서 확인할 수 있다.

3.4 키 스케줄

DES는 64-bit 키를 사용하지만 8-bit는 parity bit로 사용되고 암호화에서는 56-bit만이 라운드 키 생성에 사용된다. 라운드 키 생성에는 28-bit로 나뉘어져 28-bit 단위의 좌측 j-bit Rotation ($j=1,2$)이 수행된다. Rotation 연산 또한 logical swap을 사용하여 구현하며 Fig.7.과 같다. 마지막으로 Rotation이 수행 된 56-bit에서 8의 배수 bit가 버려지고 48-bit 라운드 키가 생성된다.

Algorithm : Rotation using logical swap

Input: 28-qubit $K([0], [1], \cdots, [27])$
Output: 28-qubit K_{New}
1: K_{New} = []
3: for $i=0$ to 27
4: K_{New}.append($K[(i+j)\%28]$)
5: **return** K_{New}

Fig. 7. Rotation using logical swap

IV. 성능 평가 및 양자 암호 분석

본 장에서는 제안하는 DES 암호화 양자 회로를 기반으로 Grover 공격 비용을 추정한다. 제안하는 DES 양자 회로에 필요한 양자 자원들은 Table.1.과 같다. Table.1.에는 NCT(NOT, CNOT, Toffoli) 단계의 양자 게이트 자원 분석이 나타나 있다.

Grover 공격 비용 추정에는 Clifford + T 게이트 단계의 세부적인 자원 분석이 필요하다. CNOT, X 게이트는 각각 Clifford 게이트로 대응되며 Toffoli 게이트는 Clifford 와 T 게이트의 다양한 조합으로 분해될 수 있다. 우리는 [12]의 접근 방식을 따라 1개의 Toffoli 게이트를 7개의 T 게이트와 8개의 Clifford 게이트로 분해하여 Oracle에 필요한 양자 자원들을 추정한다. Oracle에서는 DES 양자 회로가 2번 작동한다. 첫 번째는 암호화를 위한 수행이며 두 번째는 다음 Grover 반복을 위해 다시 평문으로 복구하는 reverse 연산이다. 따라서 큐비트를 제외하고 Table.1. × 2의 자원이 수행된다. 추가적으로 암호화 회로와 reverse 연산 사이에 생성된 암호문이 알려진 암호문과 일치하는지 확인하는 과정이 필요하다. 이때 n-bit 암호문의 경우 $n-$controlled X 게이트가 필요하며 ($32\times n-84$) T 게이트로 분해된다. 최종적으로 Grover Oracle의 Clifford + T 게이트 단계의 자원 분석 결과는 Table.2.와 같다.

DES는 64-bit 키 중 56-bit 키만이 암호화에 사용되지만 블록 암호에 대한 표준을 제시하기 위해 64-bit 키에 대한 Grover 공격 비용을 추정한다.

Table 1. Quantum resources for DES implementation (NCT)

	Qubits	Toffoli gates	CNOT gates	X gates	Depth
DES	6,648	3,536	8,032	7,552	3,205

Table 2. Quantum resources for Grover Oracle (Clfford + T)

	Qubits	Clifford gates	T gates	Total gates	Depth
DES	6,649	87,744	51,468	139,212	6,411

사용하는 DES는 Grover 알고리즘을 적용하면 $\left\lfloor \frac{\pi}{4}2^{32} \right\rfloor$ 번의 검색만으로 키를 복구할 수 있다. Grover 알고리즘은 Oracle과 Diffusion operator로 구성되어 동작하지만 2.3 장에서 언급하였듯이, Diffusion operator에 대한 비용은 무시하고 Oracle에 대한 비용만을 공격 비용으로 추정한다. Oracle은 $\left\lfloor \frac{\pi}{4}2^{32} \right\rfloor$ 번 반복되어 큐비트를 제외하고 Table.2. $\times \left\lfloor \frac{\pi}{4}2^{32} \right\rfloor$ 가 되어 DES에 대한 최종 Grover 공격 비용은 Table.3.과 같다.

Table 3. Quantum resources for Grover key search

	Total gates (Clifford+T)	Depth	Total gates × Depth
DES	1.668×2^{48}	1.229×2^{44}	1.025×2^{93}

Grover 공격 알고리즘이 대칭키 암호 시스템에 적용되어 보안 레벨이 절반으로 떨어지는 것은 정립된 사실이다. 따라서 2차적으로 판단해야 하는 중요 요소는 공격 요구 비용이다. 보안 레벨을 떨어뜨릴 수 있다 해도 공격에 필요한 비용이 너무 크다면 해당 대칭키 암호 시스템은 양자 컴퓨터의 공격에 내성을 가지고 있다고 판단이 가능하다. Table.3.의 비용 추정 결과를 보면, DES의 보안 강도를 기존 2^{58}에서 절반인 2^{29}로 감소시키는 양자 공격에는 2^{93}의 양자 자원이 필요하다는 것을 알 수 있다.

NIST는 AES의 128, 192, 256-bit 키를 기준으로 Grover 공격 비용 추정하여 양자 후 보안 강도를 제시하는 지표로 활용하고 있다[8]. 비용 추정에는2016년 Grassl et al.의 AES에 대한 Grover 공격 비용 추정 결과[2]를 사용하고 있다. 최종 비용 추정에는 Clifford 게이트와 T 게이트를 더한 총 양자게이트와 최종 양자 회로 depth를 곱하는 방식을 따른다. [2]의 AES를 양자 회로를 기반으로 최종 공격 비용은 AES-128의 경우 2^{170} (Level 1), AES-192의 경우 2^{233}(Level 3), AES-256의 경우 2^{298}(Level 5)로 추정하고 있다. 양자 후 보안 강도로 해당 지표를 활용하는데, 예를 들어 Grover 공격 비용이 2^{250}인 블록암호는 Level 3을 만족한다. Level 5를 달성하기 위해서는 공격 비용이 증가하여 2^{298} 이상을 필요로 해야 한다.

Table.3.의 DES 최종 공격 비용은 2^{93}으로 양자 후 보안 강도 Level 1을 달성하지 못한다. 이는 64-bit 키의 작은 크기를 사용하기 때문이다. NIST는 128-bit 키를 시작으로 기준을 제시하고 있기 때문에 64-bit 키를 사용하는 블록암호에 대한 기준이 모호하다. 본 논문에서는 64-bit 키를 사용하는 DES를 기준으로 Grover 공격 비용을 추정하였으며 더 작은 키를 사용하는 블록암호에 대한 평가 기준으로 활용될 수 있다.

V. 결 론

대칭키 암호에 대한 Grover 키 복구 공격은 대상 암호 양자 회로를 얼마나 효율적으로 구현하는지에 따라 최종적으로 필요한 양자 자원이 결정된다. 본 논문에서는 대칭키 암호 DES를 양자 회로를 최적화하여 구현하고 이를 기반으로 최종 Grover 공격 비용을 추정하였다. 현재 NIST에서는 AES의 128, 192, 256-bit 키를 기준으로 한 공격 비용을 추정해 양자 후 보안 강도를 제시하고 있다. 작은 크기의 64-bit 키를 사용하는 DES의 경우 NIST에서 제시하는 128-bit 키의 AES에 대한 공격 비용, Level 1의 양자 후 보안 강도를 달성하지 못했으며 어떻게 보면 키 크기에 따른 당연한 결과이다. 64-bit 키에 대한 양자 후 보안 강도 기준은 현재 없기 때문에 제안하는 64-bit 키를 사용하는 DES 공격 비용은 더 적은 키 길이를 사용하는 즉, 새로운 파라미터에 대한 양자 후 보안 강도 지표로 활용될 수 있다.

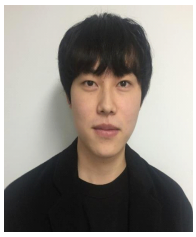
References

[1] L.K. Grover, "A fast quantum mechanical algorithm for database search," Proceedings of the twenty-eighth annual ACM symposium on Theory of computing, pp. 212 - 219, Jul. 1996.

[2] M. Grassl, B. Langenberg, M. Roetteler and R. Steinwandt, "Applying Grover's algorithm to AES: quantum resource estimates," Post-Quantum Cryptography, PQCrypto'16, LNCS, 9606, pp. 29 - 43,

Sep. 2016.
〔3〕 R. Anand, A. Maitra, and S. Mukhopadhyay, "Grover on SIMON," arXiv:2004.10686, Sep. 2020.
〔4〕 K.B. Jang, G.J. Song, H.J. Kim, H.D. Kwon, H.J. Kim, and H.J. Seo, "Efficient Implementation of PRESENT and GIFT on Quantum Computers," Applied Sciences, vol.11, no.11, pp. 4776, May. 2021.
〔5〕 K.B. Jang, G.J. Song, H.D. Kwon, S.W. Uhm, H.J. Kim, W.K. Lee, and H.J. Seo, "Grover on PIPO," Electronics, vol.10, no.10, pp. 1194, May. 2021.
〔6〕 K.B. Jang, S.J. Choi, H.D. Kwon, H.J. Kim, J.H. Park, and H.J. Seo, "Grover on Korean Block Ciphers," Applied Sciences, vol. 10, no. 18, pp. 6407, Sep. 2020.
〔7〕 Kyung-bae Jang, Hyung-jun Kim, Jae-hoon Park, Gyeung-ju Song and Hwa-jeong Seo, "Optimization of LEA Quantum Circuits to Apply Grover's Algorithm," *KIPS Transactions on Computer and Communication Systems*, 10(4), pp. 101-106, Apr. 2021.
〔8〕 NIST, "Submission requirements and evaluation criteria for the post-quantum cryptography standardization process," 〔internet〕, https://csrc.nist.gov/CSRC/media/Projects/Post-Quantum-Cryptography/documents/call-for-proposals-final-dec-2016.pdf.
〔9〕 Github: source code 〔internet〕, https://github.com/starj1023/DES_QC
〔10〕 M. Boyer, G. Brassard, P. Hoeyer, and A. Tapp, "Tight bounds on quantum searching," arXiv:quant-ph/9605034, May. 1996.
〔11〕 M. Kwan, "Reducing the Gate Count of Bitslice DES," Cryptology ePrint Archive, Report 2000/051, Oct. 2000.
〔12〕 M. Amy, D. Maslov, M. Mosca, and M. Roetteler, "A meet-in-the-middle algorithm for fast synthesis of depth-optimal quantum circuits," IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, vol. 32, no 6, pp. 818-830, Jun. 2013.

〈저자소개〉



장 경 배 (Kyung-bae Jang) 학생회원
2019년 3월: 한성대학교 IT응용시스템공학부 졸업
2021년 3월: 한성대학교 IT융합공학부 석사
2021년 3월~현재: 한성대학교 정보컴퓨터공학과 박사과정
〈관심분야〉 정보보호, 암호, 양자컴퓨터

김 현 지 (Hyun-Ji Kim) 학생회원
2020년 3월: 한성대학교 IT응용시스템공학부 졸업
2020년 3월~현재: 한성대학교 IT융합공학부 석사과정
〈관심분야〉 인공지능 보안

송 경 주 (Gyeong-Ju Song) 학생회원
2021년 3월~현재: 한성대학교 IT융합공학부 졸업
2021년 3월~현재: 한성대학교 IT융합공학부 석사과정
〈관심분야〉 정보보호, 암호, 양자컴퓨터

심 민 주 (Min-Ju Sim) 학생회원
2021년 3월~현재: 한성대학교 IT융합공학부 졸업
2021년 3월~현재: 한성대학교 IT융합공학부 석사과정
〈관심분야〉 정보보호, 부채널분석

엄 시 우 (Eum-Si Woo) 학생회원
2021년 3월~현재: 한성대학교 IT융합공학부 졸업
2021년 3월~현재: 한성대학교 IT융합공학부 석사과정
〈관심분야〉 인공지능 보안

서 화 정 (Hwa-Jeong Seo) 종신회원
2010년 3월: 부산대학교 컴퓨터공학과 졸업
2012년 3월: 부산대학교 컴퓨터공학과 석사
2016년 3월: 부산대학교 컴퓨터공학과 박사
2016년~2017년: 싱가포르 과학기술청 연구원
2019년~현재: 한성대학교 IT융합공학부 조교수
〈관심분야〉 정보보호, 암호화 구현, IoT