Journal of Digital Contents Society Vol. 14 No. 1 Mar. 2013(pp. 65-72)
http://dx.doi.org/10.9728/dcs.2013.14.1.65

병렬 컴퓨팅을 이용한 DES 키 탐색 안정성 분석

윤준원*, 최장원**, 박찬열***, 공기식****

요 약

기상, 바이오, 천문학, 암호학 등 다양한 분야의 대규모 작업을 처리하기 위하여 다수의 계산 자원을 동시에 사용하기 위한 병렬 컴퓨팅 기법들이 제안되어져 왔다. 병렬 컴퓨팅은 여러 프로세서에게 작업을 분담시켜 동시에 계산을 수행하게 함으로써 프로그램의 실행시간을 단축시킬 수 있을 뿐만 아니라 해결할 수 있는 문제의 규모를 확장 시킬 수 있다.

본 논문에서는 실제 암호 알고리즘 분석하기 위하여 병렬 처리 방식을 적용하여 그 효율성을 분석하였다. 암호 알고리즘의 실질적인 안전성 요소인 키의 길이는 전수조사 계산량에 의존한다. 이에 병렬 처리 환경에서 DES 키 탐색 암호 알고리즘의 키 전수조사 작업을 수행하기 위한 세부적인 절차에 대해서 논하였고, 클러스터링 장비에 적용하여 시뮬레이션 수행하였다. 그 결과 컴퓨터의 양에 따라서 계산량의 추이를 실증적으로 예측함으로써 암호 알고리즘의 안전성 강도를 측정할 수 있다.

키워드 : 병렬컴퓨팅, 암호키, DES 키 탐색, 전수조사

Evaluation of DES key search stability using Parallel Computing

JunWeon Yoon*, JangWon Choi**, ChanYeol Park***, Ki-Sik Kong****

Abstract

Current and future parallel computing model has been suggested for running and solving large-scale application problems such as climate, bio, cryptology, and astronomy, etc. Parallel computing is a form of computation in which many calculations are carried out simultaneously. And we are able to shorten the execution time of the program, as well as can extend the scale of the problem that can be solved.

In this paper, we perform the actual cryptographic algorithms through parallel processing and evaluate its efficiency. Length of the key, which is stable criterion of cryptographic algorithm, judged according to the amount of complete enumeration computation. So we present a detailed procedure of DES key search cryptographic algorithms for executing of enumeration computation in parallel processing environment. And then, we did the simulation through applying to clustering system. As a result, we can measure the safety and solidity of cryptographic algorithm.

Keywords : Parallel computing, cryptographic, DES Key Search, enumeration computation

1. 서 론

병렬 컴퓨팅이란 대규모의 문제를 해결하기 위해 분산된 다수의 계산 자원을 동시에 사용하는 것을 말하며, 이를 위해 여러 개의 프로세서를 가지는 단일 컴퓨터, 네트워크로 연결된 다수의 컴퓨터(Workstation Cluster) 또는 이 둘의 결합된 형태로 구성된다. 문제를 병렬로 처리하는 주된 목적은 무엇보다 프로그램의 실제 실행 시간(wall-clock time)을 줄이고자 하는 것이다. 사용자는 여러 프로세서에게 작업을 분산시켜 동시에 계산을 수행하게 함으로써 프로그램의

※ 교신저자(Corresponding Author): JunWeon Yoon
접수일:2013년 03월 11일, 수정일:2013년 03월 24일
완료일:2013년 03월 25일
* KISTI 국가슈퍼컴퓨팅연구소 선임연구원
Tel: +82-42-869-0581, Fax: +82-42-869-0569
email: jwyoon@kisti.re.kr
** KISTI 국가슈퍼컴퓨팅연구소 선임연구원
*** KISTI 국가슈퍼컴퓨팅연구소 책임연구원
**** 남서울대학교 멀티미디어학과
▣ 본 연구는 KISTI 2013년도 “국가슈퍼컴퓨팅 인프라 구축 및 운영” 연구비 지원에 의해 수행되었음.

실행시간을 단축시킬 수 있으며 또한 해결할 수 있는 문제의 규모를 키울 수 있다[1].

고성능 프로세서의 지속적인 개발은 전송속도와 소형화에 대한 물리적인 한계와 비용 상승에 대한 경제적 제한 때문에 어려움이 있다. 따라서 보다 강력하고 보다 빠른 프로세서를 기반으로 하는 고성능 단일 프로세서 시스템의 개발이 어느 정도 한계에 이르렀다. 현재 그다지 비싸지 않은 가격의 범용적인 프로세서들의 성능 또한 상당한 수준에 이르러 있기에 이러한 범용 프로세서들을 여러 개 병렬로 묶어 동시에 사용하게 함으로써 상대적으로 낮은 비용으로 사용자가 원하는 만큼의 성능이득 효과를 누릴 수 있다.

한편, 암호 키 전수조사 공격은 Exhaustive Key Search 또는 Brute-Force Search로 불리는 것으로 올바른 키를 찾을 때까지 가능한 모든 키를 조사해 보는 암호 분석의 가장 기초적인 기술이다. 올바른 키(correct key)를 식별하기 위해서는 평문과 대응하는 암호문 쌍이 필요하지만, 평문이 의미 있는 문자로 구성된 경우 암호문 단독으로도 올바른 키의 식별이 가능하다. 키 전수조사 공격은 임의의 암호 알고리즘에 대해서 적용이 가능하며, 키 스케줄 또는 암호 알고리즘 자체의 약점을 이용할 경우 공격에 대한 효율성을 향상 시킬 수 있다.

본 논문에서는 블록암호 DES 키 탐색 알고리즘을 분석을 위해 클러스터링 시스템을 이용하여 시뮬레이션을 수행하였다.

2. 관련 연구

2.1 병렬 및 분산 컴퓨팅 기술

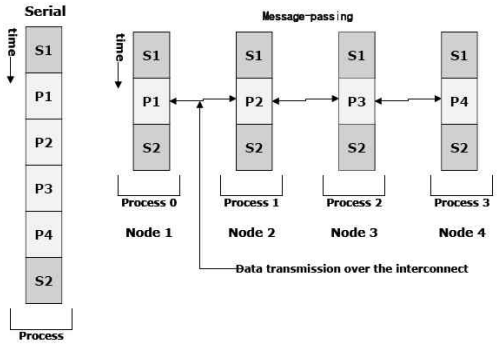
▪ **클러스터링 시스템:** 슈퍼컴퓨터의 한 종류인 클러스터는 노드로 불리는 여러 대의 컴퓨터를 통신 네트워크(Communication Network)로 병렬 연결시킨 장비로서, 한대의 컴퓨터가 할일을 여러 대의 컴퓨터로 분할해서 작업을 수행할 수 있다[2]. 슈퍼컴퓨터는 메모리 접근 방식에 따라 공유 메모리 시스템, 분산 메모리 시스템, 분산 공유 메모리 시스템으로 분류 할 수 있는데 클러스터는 이 중 분산 공유 메모리 시스템에 속한다. 각 노드가 Dual Core 또는 그 이상의 프로세서를 가지는 클러스터의 각각의 노드는 UMA 방식의 SMP 시스템으로 구성되어 있다[3].

▪ **병렬컴퓨팅:** 문제 해결을 위해 다수의 계산 자원을 동시에 사용하는 것을 말하며 이를 위해 여러 개의 프로세서를 가지는 단일 컴퓨터, 네트워크로 연결된 다수의 컴퓨터(Workstation Cluster) 또는 이 둘의 결합된 형태로 구성되는 시스템 등의 병렬 컴퓨팅 계산 자원이 필요하다. 문제를 병렬로 처리하는 주된 목적은 무엇보다 프로그램의 실제 실행시간(wall-clock time)을 줄이고자 하는 것이다[4]. 사용자는 여러 프로세서에게 작업을 분담시켜 동시에 계산을 수행하게 함으로써 프로그램의 실행시간을 단축시킬 수 있으며 또한 해결할 수 있는 문제의 규모를 키울 수 있다.

고성능 프로세서의 지속적인 개발은 전송속도와 소형화에 대한 물리적인 한계와 비용 상승에 대한 경제적 제한 때문에 어려움이 있다. 따라서 보다 강력하고 보다 빠른 프로세서를 기반으로 하는 고성능 단일 프로세서 시스템의 개발이 어느 정도 한계에 이르렀다. 현재 그다지 비싸지 않은 가격의 범용적인 프로세서들의 성능 또한 상당한 수준에 이르러 있기에 이러한 범용 프로세서들을 여러 개 병렬로 묶어 동시에 사용하게 함으로써 상대적으로 낮은 비용으로 사용자가 원하는 만큼의 성능이득 효과를 누릴 수 있다.[5]

▪ **메시지 패싱(Message Passing):** 병렬 처리에서 프로세스들 간에 데이터를 공유하기 위해 데이터를 송, 수신하여 통신하는 방식을 메시지 패싱이라고 한다[6]. 메시지 패싱은 병렬화를 위한 작업할당, 데이터분배, 통신의 운용 등 모든 것을 프로그래머가 담당하게 되고, 연결된 네트워크를 통하여 한 프로세스 메모리의 데이터를 다른 프로세스의 메모리로 복사하는 방식으로 데이터를 전달한다. 프로그래밍이 어렵지만 유용성이 좋기 때문에 다양한 하드웨어 플랫폼에서 구현이 가능하다. 이런 기능을 갖는 함수(서브루틴)들을 모아놓은 메시지 패싱 라이브러리로 MPI(Message Passing Interface), PVM(Parallel Virtual Machine), HPF(High Performance Fortran)등이 있다.

(그림 1) 메시지 패싱



(Figure 1) Message Passing

(그림 1)은 각 노드마다 하나의 프로세스가 실행되며 프로세스 내의 병렬 영역들은 실행되는 동안 서로 데이터를 주고받기 위해 다른 프로세스들과 통신을 하는 메시지 패싱 병렬 프로그래밍 모델을 나타내고 있다 . 그림에서는 인접한 프로세스 사이의 통신만이 표현되어 있지만 실제로 각 프로세스는 다른 어떤 프로세스와도 통신이 가능하다. 메시지 패싱 모델을 이용한 병렬화 과정에서는 통신으로 인한 부하, 작업 분배의 불균형, 동기화 등의 문제로 인한 성능 저하를 고려하여야한다.

▪ **MPI(Message Passing Interface):** MPI는 “Message Passing Interface”의 약어로서 프로세스들 사이의 통신을 위해 코드에서 호출해 사용하는 서브루틴(Fortran) 또는 함수(C) 들의 라이브러리이다. MPI 는 Fortran 또는 C 로 작성된 메시지 패싱 프로그램들에게 순차 프로그램들처럼 다양한 아키텍처들에 대한 풍부한 소스코드 이식성 (source-code portability)을 제공하고자 하는 표준화 작업의 결과이다. 지난 1994 년 봄에 40 개의 서로 다른 기구들을 대표하는 약 60 명의 메시지 패싱 시스템 전문가들로 구성된 MPIF(MPI Forum) 는 MPI-1 표준을 내놓았고, 1997 년에 기존의 MPI-1에 병렬 I/O, C++ 와 Fortran90 지원, 동적 프로세스 관리 등의 도구를 추가한 MPI-2를 발표하였다. 현재 MPI 는 집합 연산, 사용자 정의 데이터 타입과 토폴로지, 다양한 방식의 통신 등의 다양한 기능들과 이기종 병렬 아키텍처 (heterogeneous parallel architecture) 에 대한 지원 등을 제공한다[7].

2.2 암호학 분야 블록암호 DES 키 탐색

암호 키 전수조사 공격은 Exhaustive Key Search 또는 Brute-Force Search로 불리는 것으로 올바른 키를 찾을 때까지 가능한 모든 키를 조사해 보는 암호 분석의 가장 기본적인 기술이다. 올바른 키(correct key)를 식별하기 위해서는 평문과 대응하는 암호문 쌍이 필요하지만, 평문이 의미 있는 문자로 구성된 경우 암호문 단독으로도 올바른 키의 식별이 가능하다. 키 전수조사 공격은 임의의 암호 알고리즘에 대해서 적용이 가능하며, 키 스케줄 또는 암호 알고리즘 자체의 약점을 이용할 경우 공격에 대한 효율성을 향상 시킬 수 있다.

컴퓨터 계산 능력이 향상될수록 키 전수조사 공격의 효율성은 고정된 길이의 키에 대해서 점차 실용적인 공격이 된다. 1970년대 미국의 표준 암호 알고리즘 DES(Data Encryption Standard)가 개발될 당시에 이 알고리즘은 막대한 비용의 하드웨어를 투자하지 않는 한 키 전수조사 공격량 관점에서 안전하도록 설계되었다. 그러나 시간이 흐를수록 DES에 대한 키 전수조사 공격량은 잠재적인 공격자들에게 실질적으로 공격 가능한 계산량이 되었다[8].

키 전수조사 공격은 표준적인 데스크톱 워크스테이션 또는 개인용 컴퓨터 상에서 소프트웨어를 구동시킴으로써 수행이 가능하다. DES의 56-비트 키를 이러한 방법으로 공격할 경우에 현재에도 수십 년 또는 수백 년 걸리는 것이 일반적이다. 그러나 인터넷의 발달은 키 공간을 분할하여 수 천대의 분산된 컴퓨터를 통하여 계산하게 함으로써 56-비트 키 전수조사 공격량을 현실적인 것으로 만들었다.

1997년 2월에는 RSA사에서 개최한 DES ChallengeI에서 78,000대의 컴퓨터를 이용하여 96일 만에, 1998년 7월에는 DES Challenge II에서 250,000달러의 전용 칩을 제작하여 56시간 만에, 1999년 1월 18일 DES Challenge III에서 1만여대의 컴퓨터와 전용 칩을 이용하여 DES암호를 22시간 15분 만에 해독해 내었다. 이러한 기록은 56-비트 키를 사용하는 DES 알고리즘을 하루 내에 해독 가능하다는 것을 의미하므로 더 이상 높은 안전성을 요구하는 용도로 DES를 사용할 수 없다는 것을 의미한다.

DES Challenge의 영향으로 미국의 NIST에서

는 128-비트 크기 이상의 키 길이를 갖는 차세대 알고리즘을 제정하기 위한 AES 프로젝트를 수행한 결과 벨지움의 Rijndael이 선정된 결과를 낳게 된다. 현재, 차세대 미국 표준 암호 알고리즘인 AES는 2001년 11월부터 미 연방 표준 FIPS-197로 제정되어 사용되고 있으며, 블록 크기는 128-비트, 키 길이는 128-비트, 192-비트, 256-비트가 가능한 대칭키 블록 암호 알고리즘이다[9].

DES에서 AES로 전환되는 과정에서 가장 결정적인 역할을 한 것은 키 전수조사 공격량 관점의 안전성이다. 그러므로 암호 알고리즘의 안전성을 현실적으로 측정하는 가장 강력한 도구는 키 전수조사 공격법이라 할 수 있다. 키 전수조사 공격법은 암호 알고리즘의 안전성을 측정하기 위한 방법 중에서 가장 현실적이고 기본적인 것이다. 미국의 표준 암호 알고리즘인 DES가 차세대 표준 암호 알고리즘인 AES로 대체된 결정적인 이유도 키 전수조사 공격량 관점의 안전성 때문이다. 클러스터 컴퓨팅을 통하여 분산된 수 천대의 계산 자원 협력이 가능한 현재의 컴퓨터 네트워크 환경에서는 키 전수조사 공격량을 예측하는 것이 예전에 비해서 단순하지 않다.

3. 블록암호 알고리즘 분석

키 전수조사에 참여하는 컴퓨터의 양에 따라서 계산량의 추이를 실증적으로 예측하는 것은 암호 알고리즘의 안전성 강도를 측정하는 데에 가장 기본적이고 중요한 요소가 된다. 가장 효율적인 키 전수조사 방법이 사용될 경우의 계산량을 측정함으로써 암호 알고리즘에 대한 안전성 강도를 실질적으로 예측할 수 있는 것이다. 더 나아가 고성능인 컴퓨터를 병렬로 활용할 수 있는 대수와 계산량의 추이를 연구하는 것을 통해 분산 환경의 유휴 컴퓨터 자원을 이용한 공격량을 예측할 수 있을 것이다.

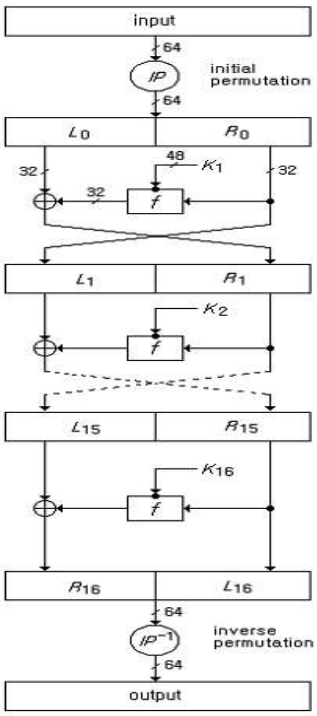
현재, 암호학계에서는 10년 또는 15년 동안의 안전성을 보장 받기 위해서 적어도 80-비트 길이의 키를 사용할 것을 권장하고 있다. 그리고 보다 좋은 안전성을 위해서는 128-비트 이상의 키 길이가 적용되어야 함을 지적하고 있다. 이러한 안전성 요소들을 실증적으로 밝히기 위해서도 컴퓨터 연계 활용 기술을 이용한 키 전수조사 공격 방법은 매우 적절하고 유용한 연구 방법인 것으로 사료된다.

본 논문에서는 분산 네트워크 환경에서 블록암호 알고리즘의 키 전수조사 계산량 관점의 안전성 분석을 수행하였다. 그 과정은 다음과 같다.

4. 클러스터링 시스템 시뮬레이션

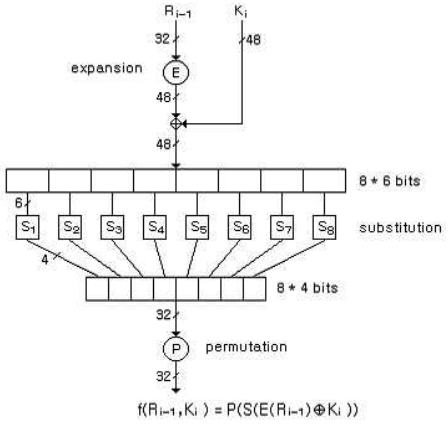
DES는 평문을 64비트로 나눠 56비트의 키를 이용해 다시 64비트의 암호문을 만들어 내는 알고리즘이다(대칭형 블록 암호). DES 알고리즘의 모습은 대체로 다음과 같다. 64비트의 평문이 16라운드를 거쳐 64비트의 암호문을 나오게 하는 Feistel 구조를 가진다. 64비트의 평문과 키 스케줄을 거친 64비트(패리티 비트 포함) 키가 입력되면 64비트의 암호문이 나오게 된다. (그림 2)는 그 과정을 나타내고 있다. 각 라운드의 f함수에는 오른쪽의 32비트 텍스트와 키 스케줄을 거친 키가 들어간다.

(그림 2) DES 동작개요



(Figure 2) DES outline

(그림 3) f - 함수



(Figure 3) f - Function

각 라운드에 입력되는 56비트의 키는 8비트의 parity bits가 포함되어 키 스케줄에 모두 64비트의 키가 들어가게 된다. parity bits는 키 사이즈를 64에서 56 비트로 줄여주며 스케줄을 거친 뒤 16개의 48비트 키가 생성된다.

사용한 DES C-code의 출처는 ETRI Library이고, NIST Special Publication 800-17 "Modes of Operation Validation System"의 125p 에 있는 test vector를 통해 DES 알고리즘을 검증 하였다. DES 알고리즘의 암/복호화 속도는 현재 장비의 1개 프로세서를 기준으로 (3.2GHz) 162.9Mbps이다.

전수조사 공격에 쓰인 평문은 "*God is Love! God love you! Forever*" 이고, padding은 DES Challenge III의 Contest Rules 에 있는 방식을 채택하였다. 이 평문을 다음의 64bit 키 "*0x0468046804680468*" 로 암호화 하였고(이 키는 parity bit를 제거하여 56bit로 알고리즘에 입력된다), CBC mode를 사용하기 위해 64bit의 Initial vector *"0x0123456789abcdef"* 를 사용하였다. 그 결과 다음의 5개의 암호문 블록을 얻었으며(L/R : 좌우 각 32bit), 이 5개의 암호문 블록을 사용해서 DES의 키 전수 조사를 실시하였다.

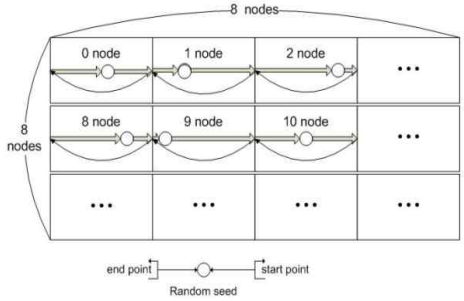
C[1].L=0x452c436a;	*C[1].R=0x04d1bfeb;*
C[2].L=0xa02cab3b;	*C[2].R=0x9ab0ea09;*
C[3].L=0x5e825132;	*C[3].R=0x59919fa4;*
C[4].L=0xc8dbb4a4;	*C[4].R=0xcee61e14;*
C[5].L=0x6fb50f1e;	*C[5].R=0xdc56f2d7;*

▪ ASCII 체크: 평문(plain text)이 알파벳과 특수문자 즉, 실제로 컴퓨터상에서 가독할 수 있게 표현되어지는 ASCII값으로만 이루어져 있다고 가정하면 plain text의 하나의 문자는 0x7f 이하의 값을 갖게 된다. 따라서 ASCII값을 갖는 문자를 0x80값과 "&" 연산을 실행해서 0x00값이 나오게 된다. 즉, 임의의 문자를 0x80값과 "&" 연산을 실행해서 0x00값이 나오면 그 문자는 ASCII값을 갖는다고 할 수 있다. 이러한 연산을 본 연구에 적용해 복호화 된 평문 블록과 0x80808080값을 연산시키는 것을 ASCII체크라 정의한다.

▪ 키 생성방법: 키의 구성은 C code의 구조 체를 사용하여 좌우 32bit씩 나누었다. 키 생성방법은 첫 번째 선택으로 총 56bit중 우측 28bit(이하 K.R)를 랜덤으로 생성하고, 좌측 28bit(이하 K.L)를 0x04860486으로 고정시킴으로서 결과적으로 28bit의 길이를 갖는 키에 대한 전수조사가 이루어지게 하였다. 두번째 선택으로 K.R을 랜덤으로 생성하고, 고정되었던 K.L중 하위 3bit만 랜덤하게 생성한 뒤 최하위bit가 parity bit이기 때문에 랜덤하게 생성된 3bit를 1bit left shift시켜서 나머지 고정된 28bit에 결합시킴으로서 31bit 키 길이에 대한 전수조사가 이루어졌다. 세 번째 선택인 35bit에서는 앞의 과정과 비슷하게 상위 21bit를 고정시키고 랜덤하게 생성된 7bit (1bit left shift된)를 결합하였다.

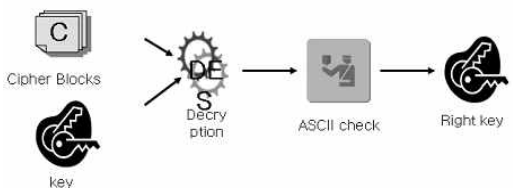
▪ 알고리즘의 병렬화: 키 테이블을 노드개수(0번 ~ 63번 노드)인 64개의 구역으로 분할한 상태에서 각 구역의 '시드 값'을 키 범위 안에 속하도록 랜덤하게 생성한다. 생성한 키로 첫 번째 암호문 블록(C[1])을 복호화 한다. 그 결과로 나온 복호문을 ASCII 체크를 통해 유효한 복호문 인지 검증하게 되고, 통과하게 된다면 사용했던 키로 두 번째 암호문 블록(C[2])를 복호화 해서 ASCII 체크를 한다. 이와 같은 방법을 다섯 번째 암호문 블록(C[5])까지 수행한다. 이렇게 모든 ASCII 체크를 통과한 키는 우리가 찾는 키가 된다. 하지만 복호문이 ASCII 체크를 통과하지 못하면 키 값을 1씩 증가시켜서 복호문을 만들게 된다. 키 값이 테이블의 '마지막 값'까지 증가했으면, 키 값을 '0'부터 '(시드 값-1)'까지 증가시키게 된다. 키를 찾았는지의 여부는 '후보 키 % 0x100000==0'일 때마다 확인을 해서 찾은 노드가 있으면 각 노드에 종료 메시지를 보내 프로그램을 종료하게 된다.

(그림 4) 테이블 구분 방식



(Figure 4) Method of Table Partition

(그림 5) 키 탐색 프레임워크



(Figure 5) Key Search Framework

5. 성능 평가

5.1 키 길이와 전수조사 공격량 상관관계

▪ DES 키 전수조사 결과

이론적으로 키 길이가 1bit 증가하게 되면 조사할 테이블이 두 배로 커짐을 생각할 수 있다. 다음 표를 보면 실험 결과는 아무래도 실험 수치이기 때문에 약간의 오차가 있어 각 bit당 정확히 두 배의 시간이 걸리는 것을 보여주진 못한다. 하지만 그 전 단계나 그 다음 단계의 실험을 포함해서 비교해 보면 각 bit당 약 두 배의 시간이 걸리는 것을 보여준다. 따라서 키 길이가 1bit 증가하게 되면 키를 찾는데 약 두 배의 시간이 더 걸린다고 결론을 내릴 수가 있다.

<표 1> 비트당 DES 키 탐색 시간

	28-bit	31-bit	35-bit	37-bit
시간(초)	8.51	70.97	557.94	5085.598
		8.36배	7.86배	9.11배

<Table 1> DES Key Search Time Per Bit

5.2 노드 수에 따른 전수조사 공격량 추이

앞서 키 전수조사에 참여하는 컴퓨터의 양 또한 중요하다고 언급했다. 클러스터링 장비에서 노드 수는 곧 컴퓨터의 양이라 할 수 있다. 계산을 하는 컴퓨터의 양이 변화할 때 전수조사의 공격량은 어떻게 변하는지 실험한 후, 컴퓨터 양과 전수조사의 공격량은 어떠한 관계가 있는지 측정하였다.

▪ DES에 대한 분석

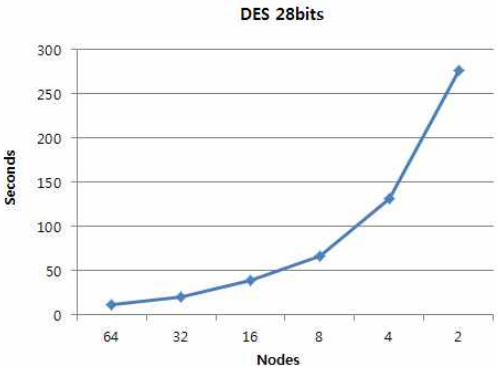
DES에서 전수조사를 할 때 알고리즘을 노드 수 변화를 위한 부분만 수정하여 전체 키 중에서 28bit를 모를 때를 기준으로 실험 하였다. 기존에 64개의 노드를 사용하였기 때문에 64, 32, 16, 8, 4, 2개의 순으로 노드의 개수를 줄여가며 전수조사 공격을 해보았다. 그 결과는 다음과 같다.

<표 2> 키 탐색 시간(노드 개수 변이에 따라)

노드 수	64개	32개	16개	8개	4개	2개
평균시간(초)	11.7	20.5	39.5	66.0	131.6	276.5
		1.75배	1.93배	1.67배	1.99배	2.10배

<Table 2> Key Search Time (variation on the number of Node)

(그림 6)키 탐색 시간(노드 개수 변이에 따라)



(Figure 6) Key Search time (variation on the number of Node)

<표 2>와 (그림 6)을 보면 알 수 있듯이 키를 찾는데 사용하는 노드의 개수를 반으로 줄이면 키를 찾는데 걸리는 시간은 약 두 배씩 증가하는 것을 알 수 있다. 실험 수치이기 때문에 정확히 두

배가 나오는 것은 염두 해 두어야 한다.

5.3 시뮬레이션 결과

블록 암호 알고리즘의 안전성은 키 전수조사에 대한 공격량을 체크함으로써 실증적으로 분석할 수 있다. 지금까지 연구에서는 블록 암호 알고리즘으로 가장 널리 사용되는 미국 표준 알고리즘 DES에 대해서 클러스터 장비를 이용한 키 전수조사 관점의 안전성 분석을 실시하였다. 키 전수조사 실험 결과 클러스터 장비 환경에서 실험 가능한 최대 키 길이는 DES의 경우 37비트 키를 찾기 위한 평균 시간이 약 1시간 24분이라는 기록이 관측되었다. 또한, 실험 결과에서 키의 길이가 1-bit 증가되면 키를 찾는 데 약 두 배의 시간이 걸리고, 키를 찾는 데 사용되는 노드의 개수를 두 배로 증가시키면 키를 찾는 데 걸리는 시간을 반으로 단축할 수 있음을 알 수 있었다.

실험 결과를 바탕으로 예상해 볼 수 있는 것은 다음과 같다. 우리가 DES를 전수 조사 공격할 경우 키 길이 중 37비트를 모를 때 키를 찾는 시간이 1시간 24분이 걸린다고 가정하면 하루 내(정확히 말하면 약22시간 24분)에 키를 찾기 위해서는 4,194,304(=64×216)개의 노드가 필요할 것이다. 즉, 4,194,304개의 컴퓨터가 참여하면 하루 내에 37비트 DES의 전수 조사가 가능하다는 것이다.

6. 결 론

키 전수조사에 참여하는 컴퓨터의 양에 따라서 계산량의 추이를 실증적으로 예측하는 것은 암호 알고리즘의 안전성 강도를 측정하는 데에 가장 기초적이고 중요한 요소가 된다. 가장 효율적인 키 전수조사 방법이 사용될 경우의 계산량을 측정함으로써 암호 알고리즘에 대한 안전성 강도를 실질적으로 예측할 수 있는 것이다. 분산 환경의 컴퓨터 자원을 이용한 공격량을 예측하기 위해서도 현 단계에서 고성능인 컴퓨터를 병렬로 활용할 수 있는 대수와 계산량의 추이를 연구하는 것이 무엇보다 중요하다. 향후 컴퓨터 계산 능력과 연계 활용 기술은 지속적으로 발전할 것이기 때문에 이러한 관점에서 암호 알고리즘의 안전성을 평가할 수 있는 기술은 정보보호 시스템 설계 시 필수적인 분야가 될 전망이다.

References

[1] Meng, Jiayuan, Raghunathan, Anand, Chakradhar, Srirn at T, Byna, Surendra ,"Exploiting the forgiving nature of applications for scalable parallel execution", Parallel &Distributed Processing(IPDPS), pp.1 - 12, April 2010.

[2] N. Sadashiv, Kumar, S.M.D,"Cluster, grid and cloud computing: A detailed comparison",Computer Science & Education (ICCSE), pp.477 - 482, July 2011.

[3] Gavril Godza, Valentin Cristea,"Comparative Study of COW and SMP Computer Configurations", Parallel Computing in Electrical Engineering(PARELEC), pp. 205-210, September, 2002.

[4] Xiao Qian, Wang Chengguo, Guo Ge,"The Research of Parallel Computing for Large-Scale Finite Element Model of Wheel/Rail Rolling Contact",Computer Science and Information Technology(ICCSIT), pp. 254-257, July, 2010.

[5] Taejung Park, "CUDA-based Object Oriented Programming Techniques for Efficient Parallel Visualization of 3D Content", Journal of Digital Contents Society Vol. 13 No. 2 pp. 169-176, June, 2012.

[6] Fox, Geoffrey C,"Lessons for massively parallel applications on message passing computers", Thirty-Seventh IEEE Computer Society International Conference, CMPCON, pp.103-114, 1992.

[7] Mamidala, Amith R, "Optimizing MPI Collectives Using Efficient Intra-node Communication Techniques over the Blue Gene/P Supercomputer", Parallel and Distributed Processing Workshops and Phd Forum (IPDPSW), pp. 771 - 780, May, 2011.

[8] NIST, FIPS PUB 46-2 : Announcing the Data Encryption Standard (DES)

[9] NIST, FIPS PUB 197 : Announcing the Advanced Encryption Standard (AES)

윤 준 원

2004년 : 고려대학교 대학원 컴퓨터학과(이학석사)
2011년 : 고려대학교 대학원 컴퓨터교육학과 박사 수료
2005년~현재 : KISTI 국가슈퍼컴퓨팅연구소 선임연구원

관심분야 : 그리드 컴퓨팅, 분산 컴퓨팅, 결함포용시스템, 클라우드 컴퓨팅, 슈퍼컴퓨팅

최 장 원

1998년 : 홍익대학교 대학원 전자공학과 (공학석사)
2009년 : 고려대학교 대학원 컴퓨터학과 (이학박사)
1998년~현재 : KISTI 국가슈퍼컴퓨팅연구소 선임연구원

관심분야 : 그리드, 클라우드 컴퓨팅, 네트워크, 정보보호

박 찬 열

1995년 : 고려대학교 대학원 컴퓨터학과(이학석사)
2000년 : 고려대학교 대학원 컴퓨터학과(이학박사)
20002년~현재 : KISTI 국가슈퍼컴퓨팅연구소 책임연구원

관심분야 : 그리드 컴퓨팅, 분산 컴퓨팅, 결함포용시스템, 슈퍼컴퓨팅

공 기 식

1999년 : 고려대학교 컴퓨터학과 (이학사)
2001년 : 고려대학교 컴퓨터학과 (이학석사)
2005년 : 고려대학교 컴퓨터학과 (이학박사)
2009년~현 재: 남서울대학교 멀티미디어학과 조교수

관심분야 : IPv6 이동성 관리, 광대역통합망, 미래인터넷, 분산 컴퓨팅