6장 정수론과 공개키 암호

6.1 소수

6.2 페르마와 오일러 정리

6.3 이산대수

6.1 소수

- 소수(Prime Number) : 하나의 양정수가 오직 1이나 자신으로만 나누어 떨어진다는 말과 동일
  - 가장 작은 소수는 2이다.
  - 1은 소수가 아니다.
  - 소수는 무한하다.
- 소수 판정법
  - 결정적 소수 판정 알고리즘
    - 임의의 변수를 사용하여 소수라는 결론을 얻음
    - 판정하는데 시간이 오래 걸려 현실적이지 못함
  - 확률적 소수 판정 알고리즘
    - 적당한 확률 이상으로 소수임을 판정하는 방법
    - 높은 확률을 가지고 빠르게 판정하는 것이 장점

6.1 소수

❑ 소수(Prime Number)

- ❖ 공개키 암호 알고리즘의 기초적 요소
- ❖ 정수 p > 1이 약수로 ±1과 ±p만을 가질경우 p는 소수

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43....

- ❖ 어떠한 정수 a > 1 은 다음과 같이 유일한 방법으로 인수분해 될 수 있음

$$a = p_1^{a_1} \times p_2^{a_2} \times \dots \times p_t^{a_t}$$

- ❖ 여기서 $P_1 < P_2 < \dots < P_t$는 소수이며 여기서 ai는 양의 정수

$$91 = 7 \times 13$$
$$3600 = 16 \times 9 \times 25 = 2^4 \times 3^2 \times 5^2$$
$$11011 = 7 \times 121 \times 13 = 7 \times 11^2 \times 13$$

6.1 소수

$$a = p_1^{a_1} \times p_2^{a_2} \times \ldots \times p_t^{a_t}$$

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43....

정수 12는 $\{a_1 = 2, a_2 = 1\}$로 나타낼 수 있다
정수 18은 $\{a_1 = 1, a_2 = 2\}$
정수 91은 $\{a_4 = 1, a_6 = 1\}$

- $12 = 4 * 3 = 2^2 * 3^1$
- $18 = 2 * 9 = 2^1 * 3^2$
- $91 = 7 * 13 = 7^1 * 13^1$

6.1 소수

$$a = p_1^{a_1} \times p_2^{a_2} \times \ldots \times p_t^{a_t}$$

2, 3, 5, 7, 11, 13, 17, 19, 23, 29, 31, 37, 41, 43....

❖ 두 수의 곱은 해당하는 멱 지수를 더한 값이다.

➢ 모든 $p \in P$에 대해서 $k = mn \rightarrow k_p = m_p + n_p$

$k = 12 \times 18 = 216$

$= (2^2 \times 3^1) \times (2^1 \times 3^2) = 216$

$k_2 = 2 + 1 = 3;\ k_3 = 1 + 2 = 3$

$12 \times 18 = 2^{k2} \times 3^{k3} = 2^3 \times 3^3 = 8 \times 27 = 216$

6.1 소수

❖ 소수들의 관점에서 a|b의 의미

➢ P^k의 형태를 가지는 어떠한 정수는 단지 그것보다 작거나 같은 지수를 가지는 소수 P^j , 단 j <= k에 의해 나누어 질 수 있음

$$a = \prod_{p \in P} p^{a_p}, b = \prod_{p \in P} p^{b_p}$$

➢ 모든 p에 대해 a|b → $a_p \le b_p$

a = 12; b = 36 ; 12|36

12 = $2^2 \times 3$

36 = $2^2 \times 3^2$

$a_2 = 2 = b_2$

$a_3 = 1 \le 2 = b_3$;

따라서 $a_p \le b_p$ 는 모든 소수에 대하여 만족

예) 12|48 9|18 12|72 18|60

6.1 소수

❖ 최대 공약수 결정

➢ 소수들로 정수를 표현하면 두 양의 정수들의 최대 공약수를 결정하기 쉬움

$$300 = 2^2 \times 3^1 \times 5^2$$
$$18 = 2^1 \times 3^2 \times 5^0$$
$$\gcd(18, 300) = 2^1 \times 3^1 \times 5^0 = 6$$

➢ 만약 k=gcd(a, b) 면, 모든 p에 대해 $k_p = \min(a_p, b_p)$

➢ 예) gcd(60, 90)

➢ 예) gcd(72, 60)

6.2 페르마와 오일러 정리

❑ 페르마 정리

❖ p가 소수이라면 a는 p에 의하여 나누어지지 않는 양의 정수이다.

❖ $a^{p-1} \equiv 1 \bmod p$ $(3^6 = 1 \bmod 7)$

➢ (p-1) 숫자들 {a mod p, 2a mod p, ..., (p-1)a mod p}은 어떠한 순서에서의 숫자 {1, 2, . . . p-1}들이다. 이러한 숫자들을 같이 곱하면

- $a \times 2a \times \ldots \times (p-1)a \equiv [(1 \times 2 \times \ldots \times (p-1)] \bmod p$
  - ➔ $a \times 2a \times \ldots \times (p-1)a = a^{p-1}(p-1)!$
  - ➔ $[(1 \times 2 \times \ldots \times (p-1)] \bmod p \equiv (p-1)! \bmod p$
- $a^{p-1}(p-1)! \equiv (p-1)! \bmod p \Rightarrow a^{p-1} \equiv 1 \bmod p$

a = 7, p = 19 => $7^{18} \equiv 1 \bmod 19$

$7^2 = 49 \equiv 11 \bmod 19$

$7^4 \equiv 121 \equiv 7 \bmod 19$

$7^8 \equiv 49 \equiv 11 \bmod 19$

$7^{16} \equiv 121 \equiv 7 \bmod 19$

$a^{p-1} = 7^{18} = 7^{16} \times 7^2 \equiv 7 \times 11 \equiv 1 \bmod 19$

$a * X_i \bmod n$

- (p-1) 숫자들 {a mod p, 2a mod p, ..., (p-1)a mod p}은 어떠한 순서에서의 숫자 {1, 2, . . . p-1}들이다

- a=5과 n=8일 경우
  - 5*1 mod 8 , 5*2 mod 8 , 5*3 mod 8 , 5*4 mod 8 ,
  - 5*5 mod 8 , 5*6 mod 8 , 5*7 mod 8

Zn	0	1	2	3	4	5	6	7
5 의 곱	0	5	10	15	20	25	30	35
나 머 지	0	5	2	7	4	1	6	3

- $a \times 2a \times \ldots \times (p-1)a \equiv [(1 \times 2 \times \ldots \times (p-1)] \bmod p$

6.2 페르마와 오일러 정리

- 페르마 정리의 다른 형태($a^{p-1} \equiv 1 \bmod p$)
- p가 소수이고 a가 양의 정수이라면
  - $a^p \equiv a \bmod p$

$p = 5, a = 3$
$a^p = 3^5 = 243 \equiv 3 \pmod 5 \equiv a \pmod p$

$p = 5, a = 10$
$a^p = 10^5 = 100000 \equiv 10 \pmod 5 \equiv 0 \pmod 5$
$\equiv a \pmod p$

Pierre de Fermat

- 1601년 8월 17일 (프랑스) - 1665년 1월 12일

경력 :

- 1648년 툴루즈 지방의회 칙선의원
- 변호사
- 툴루즈 청원위원
- 미적분학의 발전에 영향
- 페르마의 원리 발견

페르마의 마지막 정리(Fermat's Last Theorem)

- $X^2+Y^2=Z^2$ (피타고라스 정리)
- $X^n+Y^n=Z^n$ (n>=3 이상일때 , 이를 만족하는 해는 존재하지 않는다)
- 1908년 볼프스켄의 유지에 따라 왕립 과학원이 상금을 검
  - 2007년 9월 13일까지 기한
  - 10만 마르크
  - 15만 제곱까지 증명(컴퓨터 이용)
  - 영국의 수학자 앤드루 와일즈(볼프스켈 상, 1994)

6.2 페르마와 오일러 정리

❑ 오일러의 정리

- ❖ 오일러 정리는 서로 소인 모든 a와 n에 대한 관계 표현
- ❖ $a^{\varphi(n)} \equiv 1 \bmod n$

$a = 3;\ n = 10;\quad \phi(10) = 4;\quad 3^4 = 81 \equiv 1 \bmod 10$
$a = 2;\ n = 11;\quad \phi(11) = 10;\ 2^{10} = 1024 \equiv 1 \bmod 11$

❑ 오일러 Totient 함수: φ(n)

- ❖ n보다 작고 n과 서로소인 양의 정수의 개수
  - ❖ φ(10)= 원소의 갯수{1,3,7,9} = 4개
  - ❖ φ(11)= no. of {1, 2, 3,... ...10} = 10개
- ❖ P가 소수일 때는 $\varphi(p) = p - 1$ 이 성립

예) a=3일때 n=7은?? $3^6 = 1 \bmod 7$

예) a=4일때 n=5은??

6.2 페르마와 오일러 정리

- 소수 p와 q에 대해 n = pq 일때,
  - φ(n) = φ(pq) = φ(p) × φ(q) = (p-1) × (q-1)
  - Zn상에서의 나머지들의 집합 {0, 1, 2, 3, ..., (pq-1)}에 대하여 고려
    - n에 대하여 서로소가 아닌 나머지들의 집합은 {p, 2p, 3p, ..., (q-1)p}, 집합 {q 2q, 3q, ..., (p-1)q} 그리고 0임
  - 따라서 φ(n) = pq - [(q-1) + (p-1) + 1] = pq - (p+q) + 1

    = (p-1) ×(q-1) = φ(p) × φ(q)

$\phi(21) = 12 = \phi(3) \times \phi(7) = (3-1) \times (7-1) = 2 \times 6 = 12$개
==> {1,2,4,5,8,10,11,13,16,17,19,20}의 정수 12개

예) φ(15)

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
φ(n)															
n	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
φ(n)															

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
φ(n)	1	1	2	2	4	2	6	4	6	4	10	4	12	6	8
n	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30
φ(n)	8	16	6	18	8	12	10	22	8	20	12	18	12	28	8

6.2 페르마와 오일러 정리

- 오일러 정리의 다른 형태

$$a^{\varphi(n)+1} \equiv a \mod n$$

$$(a^{\varphi(n)} \equiv 1 \mod n)$$

- RSA 알고리즘의 유용성을 증명하는 유용한 오일러의 정리의 결과
- 두 개의 소수 p와 q가 주어지고, 0<m<n인 정수 n=pq와 m이 주어진다고 가정
  - $m^{\varphi(n)+1} = m^{(p-1)(q-1)+1} \equiv m \mod n$

오일러 (Leonhard Euler)

- 1707.4.15~1783.9.18 (스위스)
- 스위스의 수학자·물리학자. 수학·천문학·물리학뿐만 아니라, 의학·식물학·화학 등 많은 분야에 걸쳐 광범위하게 연구하였다. 수학분야에서 미적분학을 발전시키고, 변분학을 창시하였으며, 대수학·정수론·기하학 등 여러 방면에 걸쳐 큰 업적을 남겼다

- 논문는 800여 편이나 되고 전집만도 45권의 분량 (저서)
- 삼각 함수의 기호 사인(Sin), 코사인(Cos), 탄젠트(Tan)
- 자연 대수의 밑수 e를 처음으로 쓰기 시작

6.3 이산대수

❖ 수론에서, 수치적으로 해결이 쉽지 않은 주요 문제들
- ✓ 이산 대수 문제 (Discrete Logarithm Problem, DLP)
- ✓ 타원 곡선 상의 이산 대수 문제 (Elliptic Curve Discrete LogarithmProblem, ECDLP)
- ✓ 정수 인수분해 문제 (Integer Factorization Problem, IFP)

● 이들은, 단방향 함수 (On-way Function)의 일종으로 간주됨

● 대수 문제(logarithm problem)
- ➢ 거듭제곱의 역연산을 대수라고 함
- ➢ 보통의 수학에서 대수를 구하는 계산은 어렵지 않음
- ➢ $7^x = 49$
  - ➢ X가 2 라는것은 금방 알수 있음
  - ➢ 숫자가 커져도 대수를 구하는 계산은 어렵지 않음

6.3 이산대수

❑ 이산대수 문제(discrete logarithm problem)

❖ mod 연산에 있어서의 대수는 이산 대수라고 함

❖ $7^x mod\ 13 = 8$

❖ x 값은?

- $7^0 \bmod 13 = 1$
- $7^1 \bmod 13 = 7$
- $7^2 \bmod 13 = 10$
- $7^3 \bmod 13 = 5$
- $7^4 \bmod 13 = 9$
- $7^5 \bmod 13 = 11$
- $7^6 \bmod 13 = 12$
- $7^7 \bmod 13 = 6$
- $7^8 \bmod 13 = 3$
- $7^9 \bmod 13 = 8$

6.3 이산대수

❑ 이산대수 구하기는 어려움

- ❖ 앞의 예에서는 쉽게 x의 값이 9라는 것을 알 수 있음
- ❖ 하지만 7과 같은 작은 수가 아니고 매우 큰 수라면 해당 이산대수를 구하기는 매우 어려움
- ❖ 특히 숫자가 백 자리 이상으로 커지면 이산대수를 구하는 것은 고속연산을 수행할 수 있는 컴퓨터를 이용하더라도 상당히 어렵고 시간이 굉장히 많이 걸림
- ❖ 현재까지 이산대수를 구하는 고속 알고리즘은 발견되지 않음

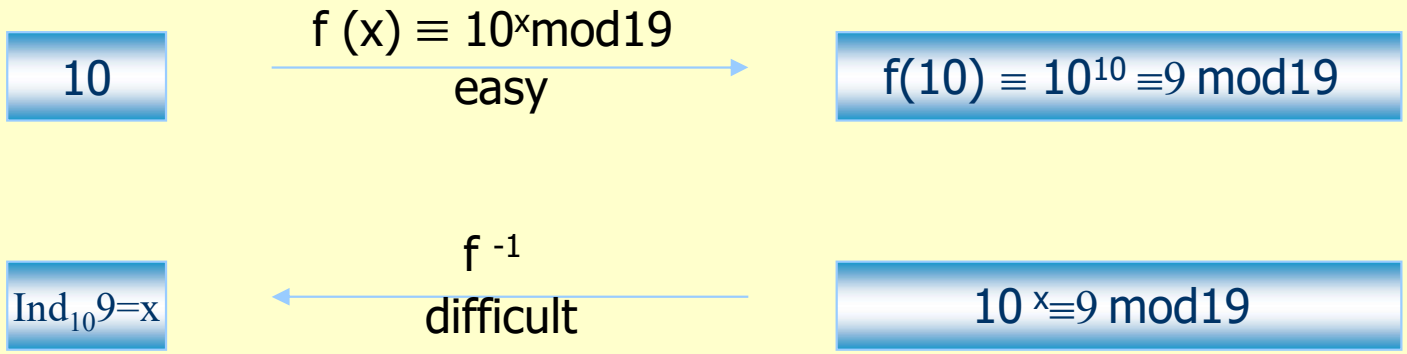
● 유한체 상의 이산 대수 문제 (Discrete Logarithm Problem, DLP)

- ➢ $y = g^x \bmod p$. g,x,p를 알고서, y를 구하기(계산하기)는 쉽지만,
- ➢ y,g,p를 알고서, x를 구하는 문제의 어려움
- ➢ 즉, 지수 계산의 역 연산인 로그 계산의 어려움 ($x = \log_g y$)

6.3 이산대수

❑ 이산대수 문제(discrete logarithm problem)

❖ 큰 수 n을 법으로 하는 지수승 $y \equiv k^x \bmod n$은 계산하기 쉽지만, 주어진 y와 k에 대하여 식 $y \equiv k^x \bmod n$을 만족하는 x를 구하기 어려운 점을 이용하는 이론



❖ Diffie- Hellman, ElGamal, Massey-Omura, ECC

6.3 이산대수

- 이산대수는 Diffie-Hellman의 키 교환과 전자서명 알고리즘(DSA: Digital Signature Algorithm)을 포함하는 공개키 암호 알고리즘에서 중요한 개념
- Modulo n상에서 정수의 멱
  - $a^{\varphi(n)} \equiv 1 \bmod n$ (오일러의 정리)
  - φ(n) : n보다 작은 양의 정수이고 n과는 서로 소
  - 보다 일반적인 표현으로 바꾸면,

    $a^m \equiv 1 \bmod n$ ..... (*)
  - 만약 a와 n이 서로소이라면, 등식 (*)은 m = φ(n)을 만족하는 정수 m이 적어도 하나 존재한다.
- 등식 (*)을 만족하는 가장 작은 양의 멱 지수 m은 다음과 같다.
  - a (mod n)에 대한 위수(order) ; m
  - a가 (mod n)에 속한 멱 지수
  - a에 의해 생성된 주기의 길이

6.3 이산대수

- modulo 19상에서 7에 대한 멱 지수 예
- $7^1 =$ $7 \bmod 19$

  $7^2 = 49 = 2 \times 19 + 11 = 11 \bmod 19$

  $7^3 = 343 = 18 \times 19 + 1 = 1 \bmod 19$

  $7^4 = 2401 = 126 \times 19 + 7 = 7 \bmod 19$

  $7^5 = 16807 = 884 \times 19 + 11 = 11 \bmod 19$

- 순서가 반복되는 것을 발견
  - $7^3 = 1 \pmod{19}$, $7^{3+j} = 7^3 7^j = 7^j \pmod{19}$
  - 3만큼의 차이가 나는 7에 대한 어떠한 두 멱 지수는 각각 (mod 19)에 있어서 합동(주기가 존재)
  - 주기의 길이는 $7^m = 1 \pmod{19}$를 만족하는 가장 작은 양의 멱 지수 m이다.

6.3 이산대수

- 다음 표는 양수 a<19에 대해 modulo 19상에서 a에 대한 모든 멱 지수를 표현

1. 모든 순열은 1에서 끝이 난다.
2. 순열의 길이는 φ(19) = 18로 구분된다. 즉, 순열에 대한 전체 숫자는 표의 각 행에서 나타난다.
3. 어떠한 순열들은 길이가 18이다. 이러한 경우에 있어서, 밑수 a는 modulo 19상에서 0이 아닌 정수들의 집합을 생성한다. 그와 같은 각각의 정수를 modulus 19의 **원시 근(primitive root)**라고 부른다.

(2,3,10,13,14,15)

6.3 이산대수

❑ Modulo 19의 정수 멱

a^1	a^2	a^3	a^4	a^5	a^6	a^7	a^8	a^9	a^{10}	a^{11}	a^{12}	a^{13}	a^{14}	a^{15}	a^{16}	a^{17}	a^{18}
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	4	8	16	13	7	14	9	18	17	15	11	3	6	12	5	10	1
3	9	8	5	15	7	2	6	18	16	10	11	14	4	12	17	13	1
4	16	7	9	17	11	6	5	1	4	16	7	9	17	11	6	5	1
5	6	11	17	9	7	16	4	1	5	6	11	17	9	7	16	4	1
6	17	7	4	5	11	9	16	1	6	17	7	4	5	11	9	16	1
7	11	1	7	11	1	7	11	1	7	11	1	7	11	1	7	11	1
8	7	18	11	12	1	8	7	18	11	12	1	8	7	18	11	12	1
9	5	7	6	16	11	4	17	1	9	5	7	6	16	11	4	17	1
10	5	12	6	3	11	15	17	18	9	14	7	13	16	8	4	2	1
11	7	1	11	7	1	11	7	1	11	7	1	11	7	1	11	7	1
12	11	18	7	8	1	12	11	18	7	8	1	12	11	18	7	8	1
13	17	12	4	14	11	10	16	18	6	2	7	15	5	8	9	3	1
14	6	8	17	10	7	3	4	18	5	13	11	2	9	12	16	15	1
15	16	12	9	2	11	13	5	18	4	3	7	10	17	8	6	14	1
16	9	11	5	4	7	17	6	1	16	9	11	5	4	7	17	6	1
17	4	11	16	6	7	5	9	1	17	4	11	16	6	7	5	9	1
18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1	18	1

6.3 이산대수

- mod φ(n)에 속할 수 있는 숫자들 가운데 가장 높은 지수를 φ(n)이라고 말할 수 있다.
- a가 n의 원시근이라면, 그 때 그것의 멱승은 다음과 같다.
  - $a, a^2, \ldots, a^{\varphi(n)}$
- 이것은 (mod n)에 의하여 구별이 되며, n에 모두 서로 소이다.
- 다르게 표현해서,
  - 소수 p에 대해, 만약 a가 p의 원시 근이라면,
  - $a, a^2, \ldots, a^{p-1}$은 (mod p)와 구별이 된다.
  - 소수 19에 대해, 원시근은 2, 3, 10, 13, 14 그리고 15이다.
  - 모든 정수들이 원시근을 가지는 것은 아니다

6.3 이산대수

- 임의의 소수 P 에 대하여 근의 멱승이 1부터 P-1까지의 모든 정수를 생성하는 원시근을 α라 하자.
- α의 멱승에 대한 P모듈로 계산 결과: $\alpha \bmod p$, $\alpha^2 \bmod p$, ..., $\alpha^{p-1} \bmod p$; 1부터 P-1까지의 각각 다른 정수를 생성.

- 소수 p=7, 원시근 α =3 일 때, $b = \alpha^i \bmod p$ 에서
  - $3^1 \bmod 7 = 3$,
  - $3^2 \bmod 7 = 2$,
  - $3^3 \bmod 7 = 6$,
  - $3^4 \bmod 7 = 4$,
  - $3^5 \bmod 7 = 5$,
  - $3^6 \bmod 7 = 1$,

계산결과 b는 1부터 6까지의 각각 다른 정수를 구성

$P^n \mod n$

❑ p=5과 n=8일 경우

Zn	0	1	2	3	4	5	6	7
5의 멱승	1	5	25	125	625	3125	15625	78125
나 머 지	1	5	1	5	1	5	1	5

❑ p=5과 n=7일 경우

Zn	0	1	2	3	4	5	6
5의 멱승	1	5	25	125	625	3125	15625
나 머 지	1	5	4	6	2	3	1

6.3 이산대수

❑ 이산대수의 계산

❖ $y=g^x \bmod p$

- g, x 그리고 p 가 주어진다면 y를 구하는 것은 쉬움
  - 최악의 경우, 반복적인 곱셈 과정을 x번 수행
- 그러나 y,g 그리고 p가 주어진다고 하더라도 x를 계산하는 것은 매우 어려움
  - 이 계산의 어려움은 RSA알고리즘을 풀기 위해 요구되는 인수분해의 경우와 동일

❖ 대수의 속성

- $\log_x(1) = 0$, $\log_x(x) = 1$, $\log_x(yz) = \log_x(y) + \log_x(z)$
- $\log_x(y^r) = r \times \log_x(y)$