2018/01/31, 2018 보안 기초 세미나

Network Security Essentials

- Chapter_3 공개키 암호와 메시지 인증 -

권순홍 (soonhong@pel.smuc.ac.kr)

상명대학교 프로토콜공학연구실

목 차

- 공개키 암호 원리
- 공개키 암호 알고리즘
  - RSA 알고리즘
  - Diffie-Hellman 알고리즘
  - 기타 알고리즘
    - 디지털 서명
    - 타원 곡선 알고리즘

공개키 암호 원리

- 공개키 암호(Public-key encryption)
  - 정의
    - 암호화/복호화할 때 서로 다른 두 개의 키를 사용하는 암호 방식
      - 공개키(Public key)
        - 외부에 공개 되는 키
      - 개인키(Private key)
        - 외부에 공개해서는 안 되는 키
  - 특징
    - 수학적 함수에 의해 만들어짐
    - 기밀성,키분배,인증 분야에서 성능이 뛰어남
    - 공개키로 암호화한 암호문은 공개키와 대응되는 개인키만이 복호화 가능

공개키 암호 원리

- 공개키 암호 구조
  - 핵심 요소
    - 평문(Plaintext)
      - 알고리즘의 입력으로 사용되며, 사람이 읽을 수 있는 메시지,데이터
    - 암호 알고리즘(Encryption algorithm)
      - 평문을 여러가지 형태로 변환시키는 알고리즘
    - 공개키와 개인키(Public and Private key)
      - 암호 알고리즘에 의한 변환은 입력으로 사용되는 공개키/개인키에 의해 이루어짐
    - 암호문(Ciphertext)
      - 출력으로 나오는 암호화된 메시지
    - 복호 알고리즘(Decryption algorithm)
      - 평문을 암호화할 때 사용한 키에 대응하는 키를 이용하여 원래의 평문으로 변환하는 알고리즘

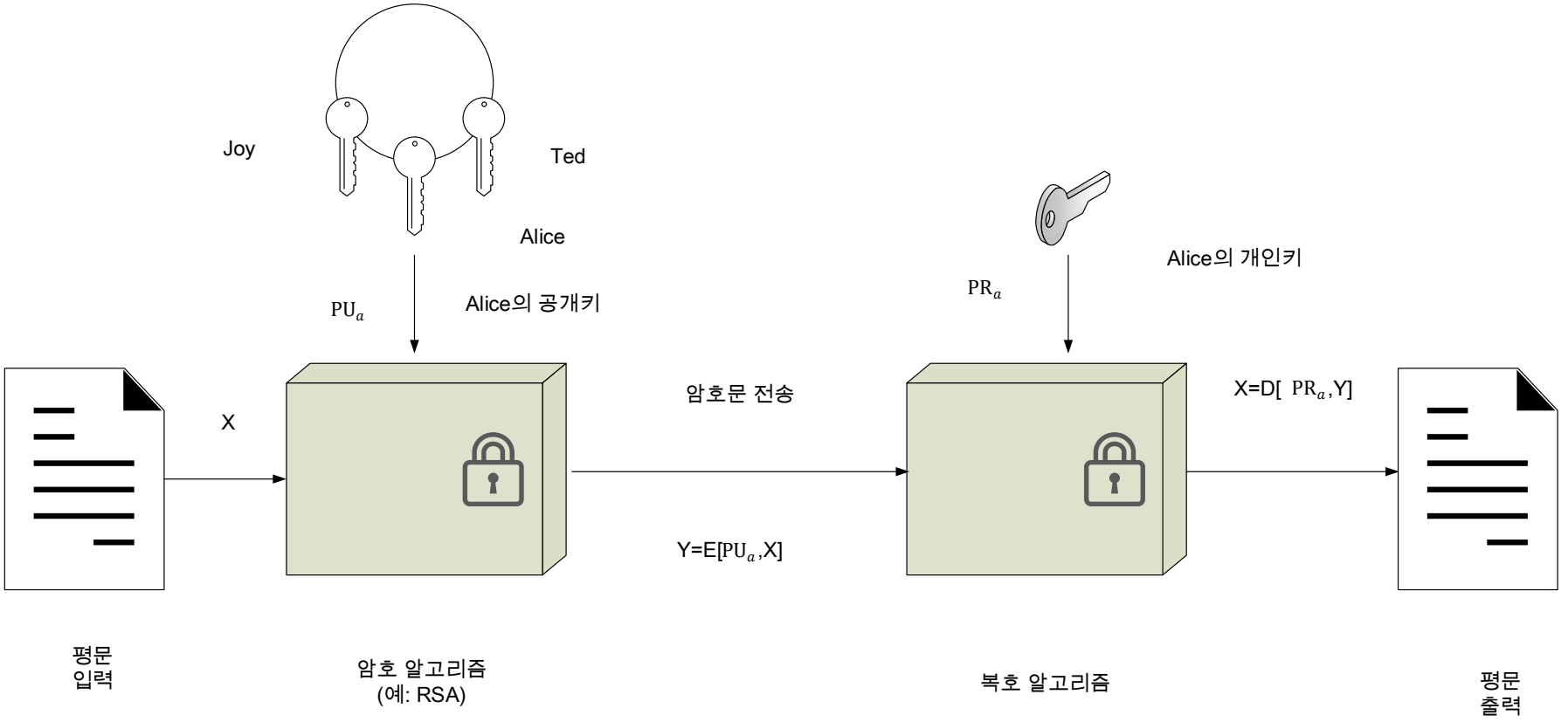
공개키 암호 원리

- 공개키 암호
  - 대칭키와 공개키 암호 방식 비교

구분	대칭키 암호방식	공개키 암호 방식
키	대칭키(비밀키)	비대칭키(공개키,개인키)
키의 상호관계	암호화키=복호화키	암호화키≠복호화키
암호화키/복호화키	비밀/비밀	공개/비밀
암호 알고리즘	공개	공개
키의 개수	n*(n-1)/2	2*n
장점	계산속도 빠름 알고리즘이 다양	암호화키 사전 공유 불필요 통신 대상의 추가가 용이 인증 기능 제공
단점	키 분배 및 관리의 어려움	계산속도 느림
대표적인 예	DES,AES,IDEA	RSA,ECC

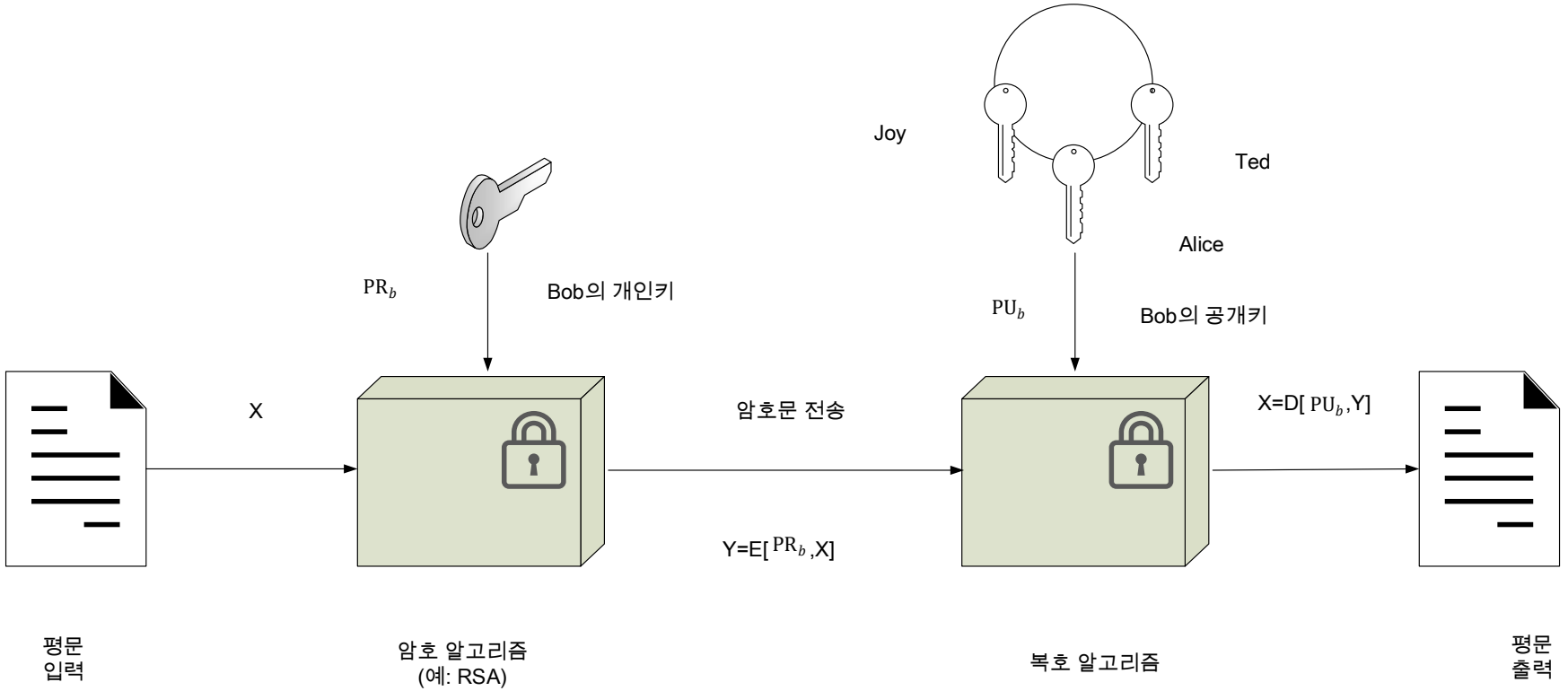
공개키 암호 원리

- 공개키 암호
  - 구조
    - 공개키에 의한 암호화



공개키 암호 원리

- 공개키 암호
  - 구조
    - 개인키에 의한 암호화



공개키 암호 원리

- 공개키 암호
  - 응용
    - 암호화/복호화(Encryption/decryption)
      - 송신자는 수신자의 공개키를 이용해 메시지 암호화
    - 디지털 서명(Digital signature)
      - 송신자는 자신의 개인키로 메시지에 서명
    - 키 교환(Key exchange)
      - 키 합의(Key Agreement)
        - 양측의 개인키와 공개키를 사용하여 비밀키 생성
          - e.g., Diffie-Hellman알고리즘
      - 키 전송(Key Transport)
        - 하나의 주체가 비밀키를 생성후 다른 사람의 공개키로 암호화 한 후 전송
          - e.g., RSA알고리즘

공개키 암호 원리

- 공개키 암호
  - 공개키 암호 시스템의 응용 표

알고리즘	암호/복호	디지털 서명	키 교환
RSA	Y	Y	Y
Diffie-Hellman	N	N	Y
DSS	N	Y	N
타원 곡선	Y	Y	Y

공개키 암호 원리

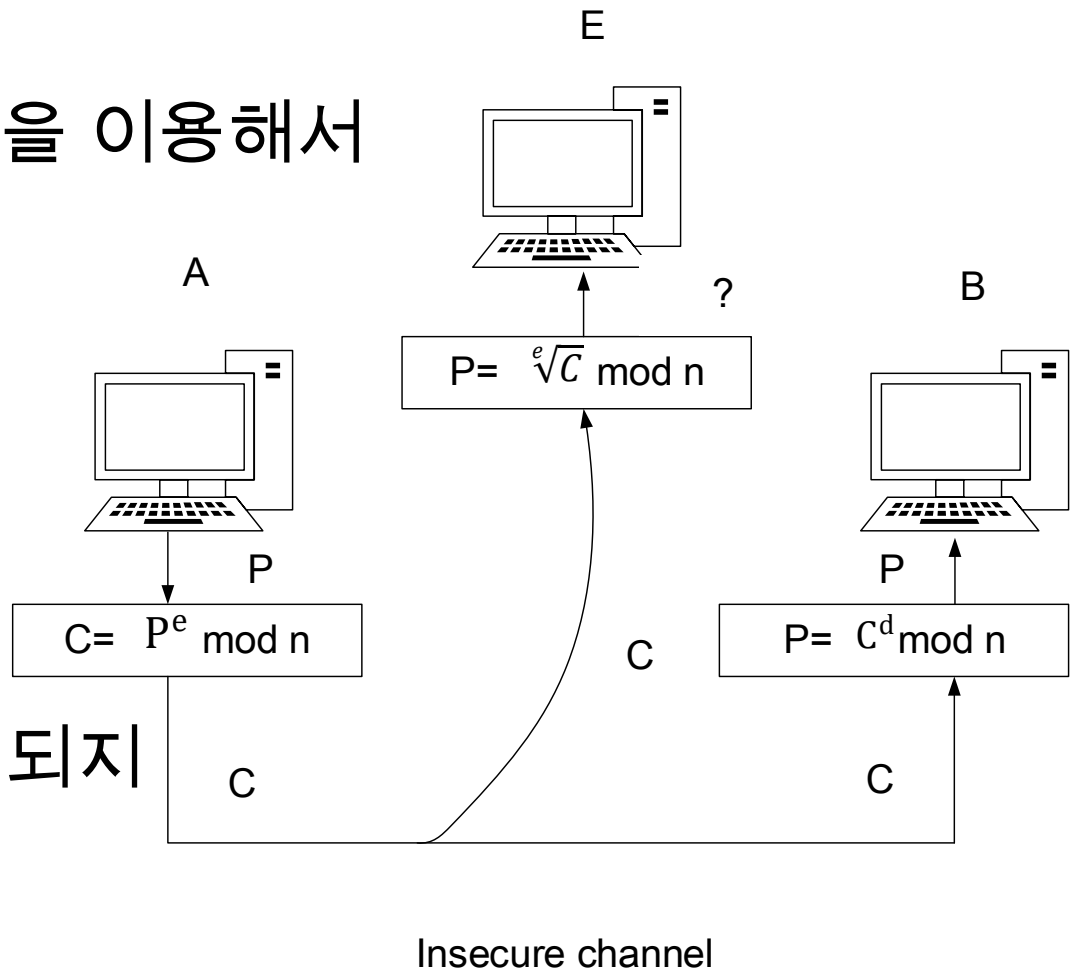
- 공개키 암호
  - 요건
    - 한 쌍의 키(공개키,개인키)를 생성하는 것은 계산적으로 쉬워야 함
    - 송신자는 암호문을 계산적으로 쉽게 구할 수 있어야함
      - $C = E(PU, M)$
    - 수신자는 원문으로 복호화하는 것이 계산적으로 쉬워야함
    - 공격자가 개인키를 알아내는 것이 계산적으로 불가능 해야함
    - 공격자가 원문을 알아내는 것이 계산적으로 불가능 해야함
    - 2개의 키 중 어느 하나를 암호화에 사용하면 다른하나 는 복호화에 사용할 수 있음
      - $M = D[PU, E(PR, M)] = D[PR, E(PU, M)]$

공개키 암호 알고리즘

- RSA (Rivest Shamir Adleman)알고리즘
  - 개요
    - 1977년 MIT에서 Ron Rivest와 Adi Shamir 그리고 Len Adleman이 만들어서 1978년 최초로 출판한 알고리즘
  - 특징
    - 소인수분해 문제의 어려움을기반으로 한 알고리즘
      - 모듈로 지수계산 사용
        - 모듈로 로그는 모듈로 값을 소인수분해하는 것 만큼 어려움
    - 해독하는 데 시간이 오래 걸림
    - 평문 메시지와 암호문,키 값을 모두 숫자로 취급함

공개키 암호 알고리즘

- RSA
  - 연산의 복잡도
    - RSA는 모듈로 지수계산을 이용해서 암호화/복호화를 함
    - 공격하려는 제 3자는 $\sqrt[e]{C} \bmod n$을 계산해야함
    - 모듈로 n에 대한 e 거듭제곱근을 구하는 속도가 다항식 정도의 복잡도를 가진다면 일방향 함수가 되지 못함



공개키 암호 알고리즘

- RSA 알고리즘
  - 표기법

p,q(p,q 각각 512비트)	키를 생성하기 위해 선택하는 소수값
n(1024비트)	p*q의 합성수
M	평문
C	암호문
e	공개키의 인자값(공개 값)
d	개인키의 인자 값(비밀 값)
PU={e,n}	공개키
PR={d,n}	개인키

공개키 암호 알고리즘

- RSA 알고리즘
  - 키 생성
    - B는 키 생성 후 순서쌍 (e,n) 자신의 공개 키로 선언
    - B는 d를 자신의 개인 키로 하고 비밀로 함
    - 안전을 위해 권장되는 각 소수 p와q의 크기는 512비트
      - 10진수로 약 154자리
      - 512비트 소수 사용시 모듈로 n은 1024비트를 가짐

공개키 암호 알고리즘

- RSA 알고리즘
  - 키 생성 알고리즘
    - RSA에서 순서쌍 (e,n)은 공개 키 정수 d는 개인 키

p,q	키를 생성하기 위해 선택하는 소수값
n	p*q의 합성수
M	평문
C	암호문
e	공개키의 인자값(공개 값)
d	개인키의 인자 값(비밀 값)
PU={e,n}	공개키
PR={d,n}	개인키

```
RSA_Key_Generation
{
        큰 소수 p와 q를 선택   //p≠q
        n←p*q                  //n은 p*q의 합성수
        ϕ(n)←(p-1)*(q-1)
        e 선택                 //1<e< ϕ(n)이면서  ϕ(n)와 서로소
        d= e^-1 mod  ϕ(n)       //d는 모듈로 ϕ (n)으로 e의 역원
        Public_key←(e,n)
        Private_key←d
        return Public_key and Private_key
}
```

공개키 암호 알고리즘

- RSA 알고리즘
  - 키 생성 예시
    1. 두 소수 $p = 17$과 $q = 11$선택
    2. $n = p \times q = 17 \times 11 = 187$ 계산
    3. $\phi(n) = (p-1) \times (q-1) = 16 \times 10 = 160$
    4. $\phi(n) = 160$보다 작으면서 $\phi(n)$과 서로소인 e선택
       - E=7 선택
    5. $d < 160$이면서 $de \bmod 160 = 1$인 d 결정
       - $23 \times 7 = 161 = 1 \times 160 + 1$

공개키 암호 알고리즘

- RSA 알고리즘
  - 암호화 알고리즘
    - 누구나 이 공개 키를 이용하여 메시지를 암호화후 전송가능
    - 다항식 연산 정도의 복잡도를 가진 알고리즘
    - 평문의 크기가 n보다 작아야함
      - 만약 n보다 크다면 평문은 길이가 n보다 작은 블록으로 나눔

```
RSA_Encryption(P,e,n)                          //P는 Z_n 에서의 평문이고 P<n
{
        C←Fast_Exponentiation(P,e,n)           //  P^e mod n계산
        return C
}
```

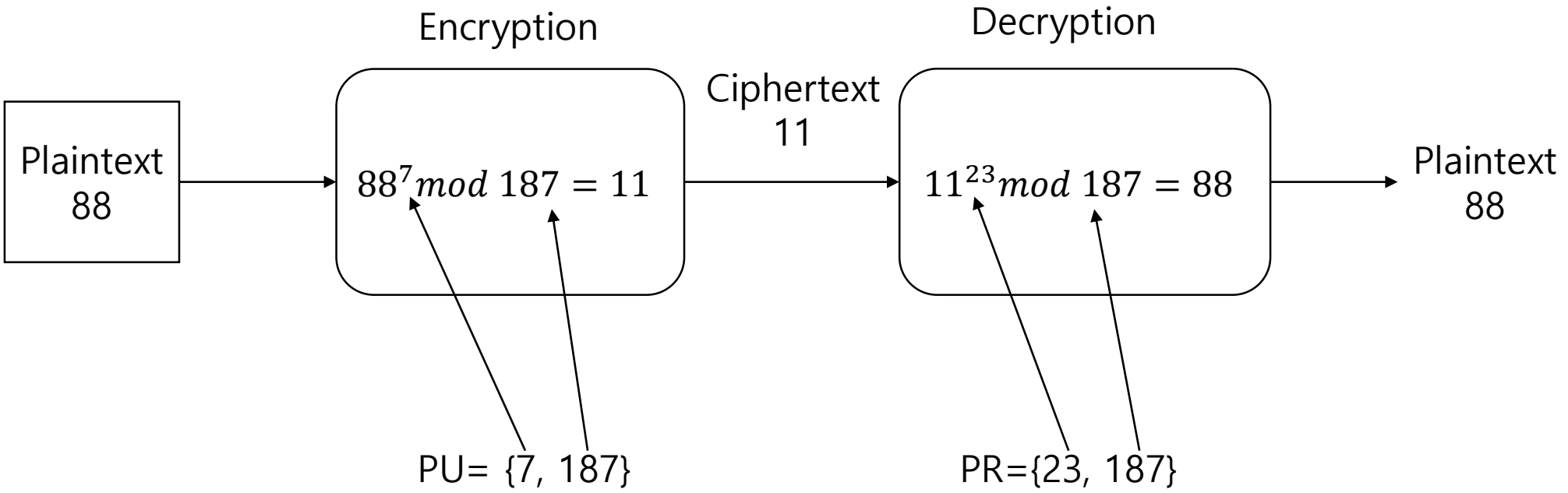
공개키 암호 알고리즘

- RSA 알고리즘
  - 복호화 알고리즘
    - 복호화 알고리즘 이용 시 다항식 연산 정도의 복잡도 계산가능
    - 암호문의 크기 n보다 작아야함

```
RSA_Decryption(C,d,n)                          //C는 Z_n 에서의 평문
{
        P←Fast_Exponentiation(C,d,n)           //  C^d mod n계산
        return P
}
```

공개키 암호 알고리즘

- RSA 알고리즘
  - 예시



공개키 암호 알고리즘

- RSA 알고리즘
  - 보안
    - 전수 공격
      - 가능한 모든 개인키를 시도해 보는 공격
      - e와 d의 비트 수가 크면 클수록 알고리즘은 안전
    - 두 개의 소인수의 곱을 인수분해
      - 효과적 인수분해 방법 없음
      - 대부분의 RSA응용에 1024-비트 키(대략 300자리 10진수)를 사용하고 있어서 충분히 안전

공개키 암호 알고리즘

- RSA 알고리즘
  - 소인수 분해(1)
    - 쉬운 소인수분해
      - *http://factordb.com/*

78 Factorize! (?)

Result:		
status (?)	digits	number
FF	2 (show)	78 = 2 · 3 · 13

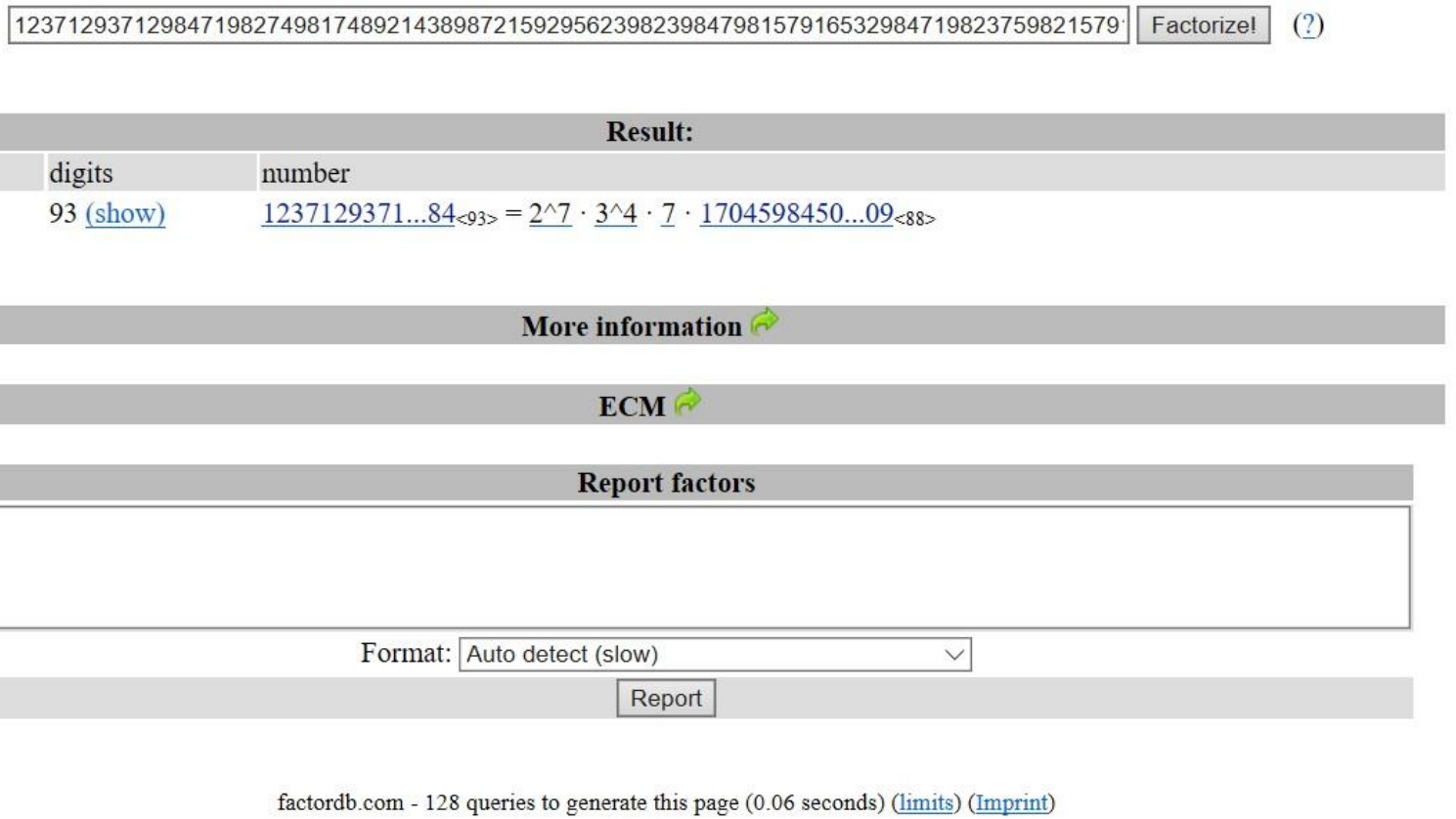
More information

ECM

factordb.com - 3 queries to generate this page (0.01 seconds) (limits) (Imprint)

공개키 암호 알고리즘

- RSA 알고리즘
  - 소인수 분해(2)
    - RSA에서 필요로하는 N값



공개키 암호 알고리즘

- Diffie-Hellman 알고리즘
  - 개요
    - 1976년 Whitfield Diffie와 Martin Hellman에 의해 개발된 키 교환 알고리즘
  - 특징
    - 이산 대수 문제의 어려움을 기반으로 한 알고리즘
    - 공개키를 교환하여 통신 양측이 사용할 비밀키 생성

공개키 암호 알고리즘

- Diffie-Hellman 알고리즘
  - 이산대수 문제(Discrete logarithms problem)
    - 원시근(Primitive root)α
      - 소수 p의 원시근
        - 자신의 거듭제곱을 이용하면 1부터 p-1까지 정수를 모두 생성할 수 있는 수
        - a mod p, a^2mod p, …, a^{p-1}mod p
    - 이산대수(Discrete logarithm)
      - P보다 작은 임의의 정수 b와 p의 원시근 a에 대해

        b=a^imod p, (0≤i≤p-1)

        지수(exponent) i를 밑수 a로 갖는 b의 이산대수 혹은 지수라 함

        i=d$\log_{a,p}(b)$

공개키 암호 알고리즘

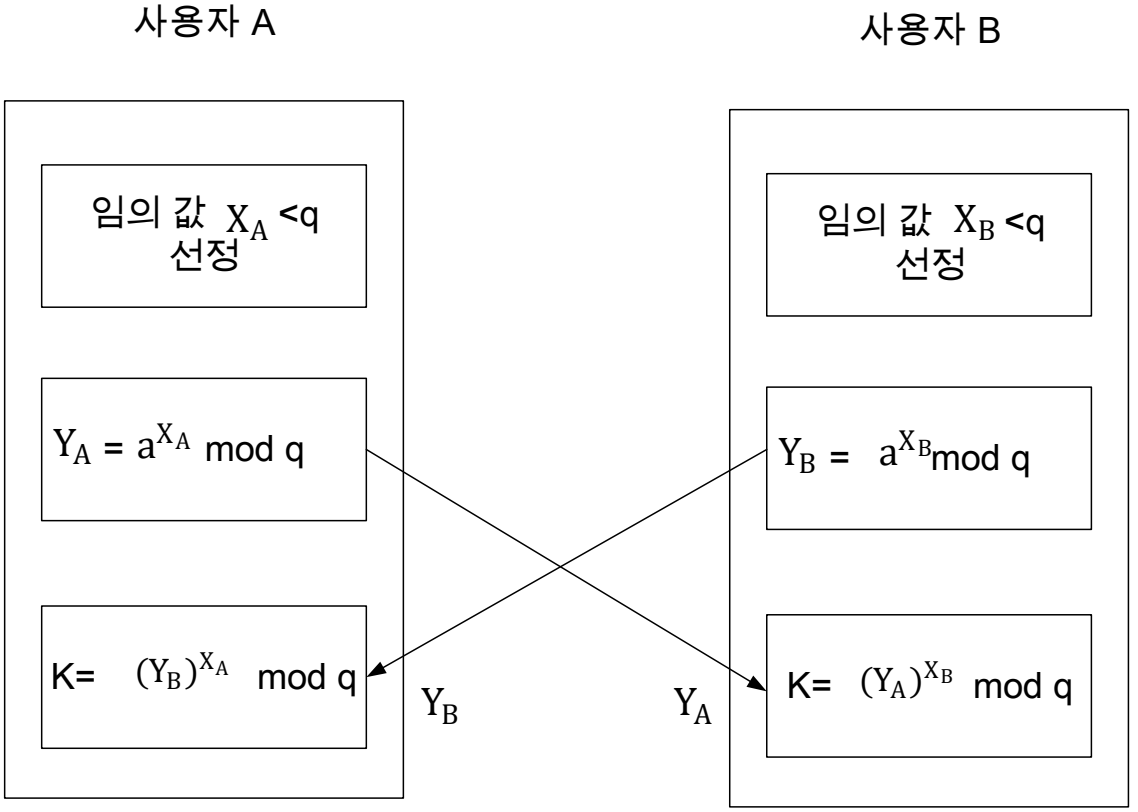
- Diffie-Hellman 알고리즘
  - 이산대수 문제 예
    - 7^x mod 13=6이 되는 x값?
      - 7^0 mod 13=1
      - 7^1 mod 13=7
      - 7^2 mod 13=10
      - 7^3 mod 13=5
      - 7^4 mod 13=9
      - 7^5 mod 13=11
      - 7^6 mod 13=12
      - 7^7 mod 13=6

      X=7

공개키 암호 알고리즘

- Diffie-Hellman 알고리즘
  - 대칭키 생성 과정
    - 공개되는 값

1. 통신 양측은 임의의 개인값 X_A, X_B 선택
2. 공개값 Y_A, Y_B 계산
   $Y_A = a^{X_A} \bmod q$
   $Y_B = a^{X_B} \bmod q$
3. 계산한 공개 값 전송
4. 대칭키 생성
   $K = (Y_B)^{X_A} \bmod q$
   $= (Y_A)^{X_B} \bmod q$



공개키 암호 알고리즘

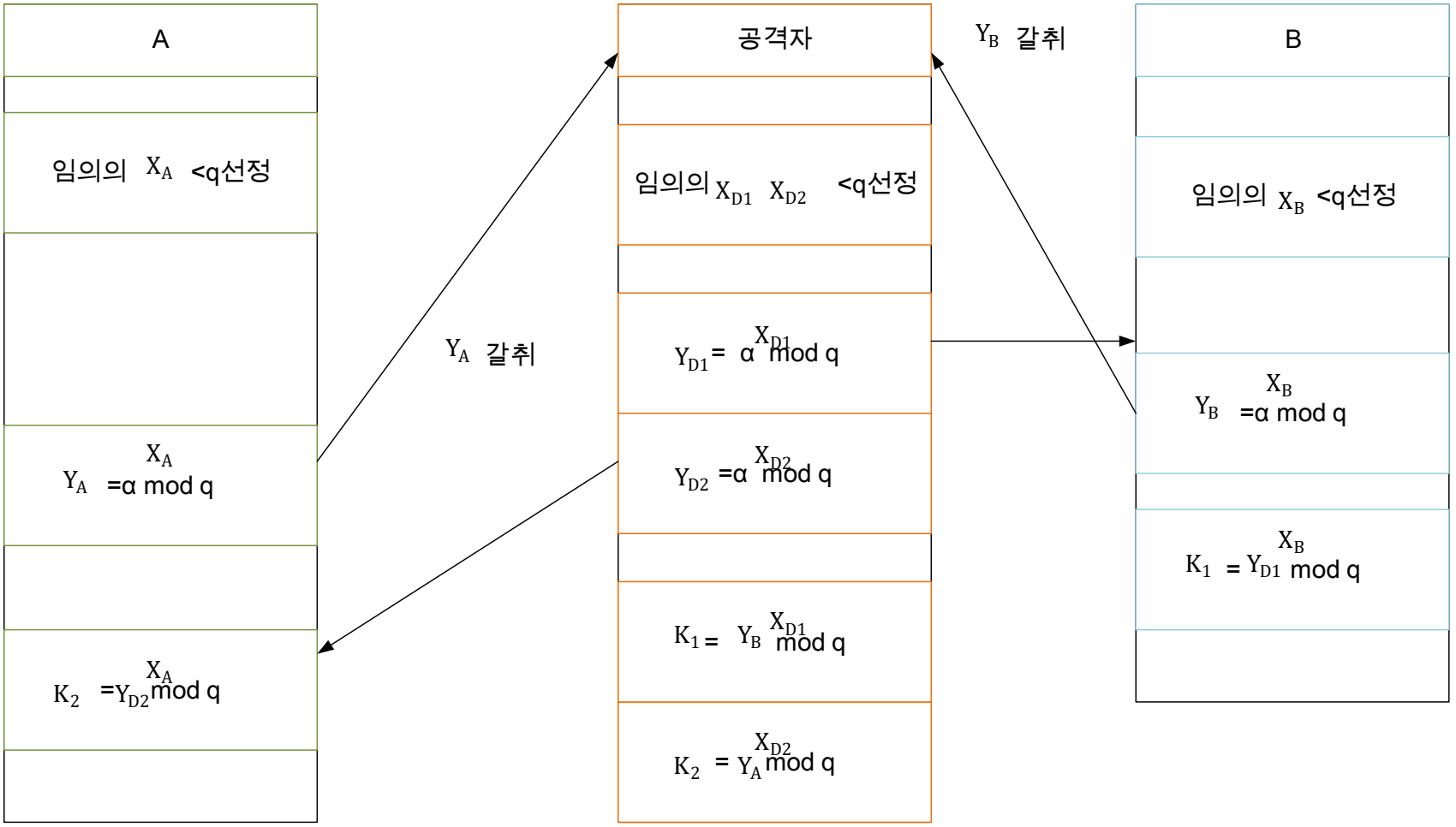
- Diffie-Hellman 알고리즘
  - 동일키 계산
    - 소수 q=353, 353의 원시근 α:3, X_A=97, X_B=233

1. A와 B는 각자 자신의 공개값 계산
   - A는 $Y_A = 3^{97} mod\ 353 = 40$
   - B는 $Y_B = 3^{233} mod\ 353 = 248$

2. 서로의 공개값 교환후 비밀키 생성
   - K=$Y_B^{X_A} mod\ 353 = 248^{97} mod\ 353 = 160$
   - K=$Y_A^{X_B} mod\ 353 = 40^{233} mod\ 353 = 160$

공개키 암호 알고리즘

- Diffie-Hellman 알고리즘
  - 중간자 공격(Man-in-the-middle attack)

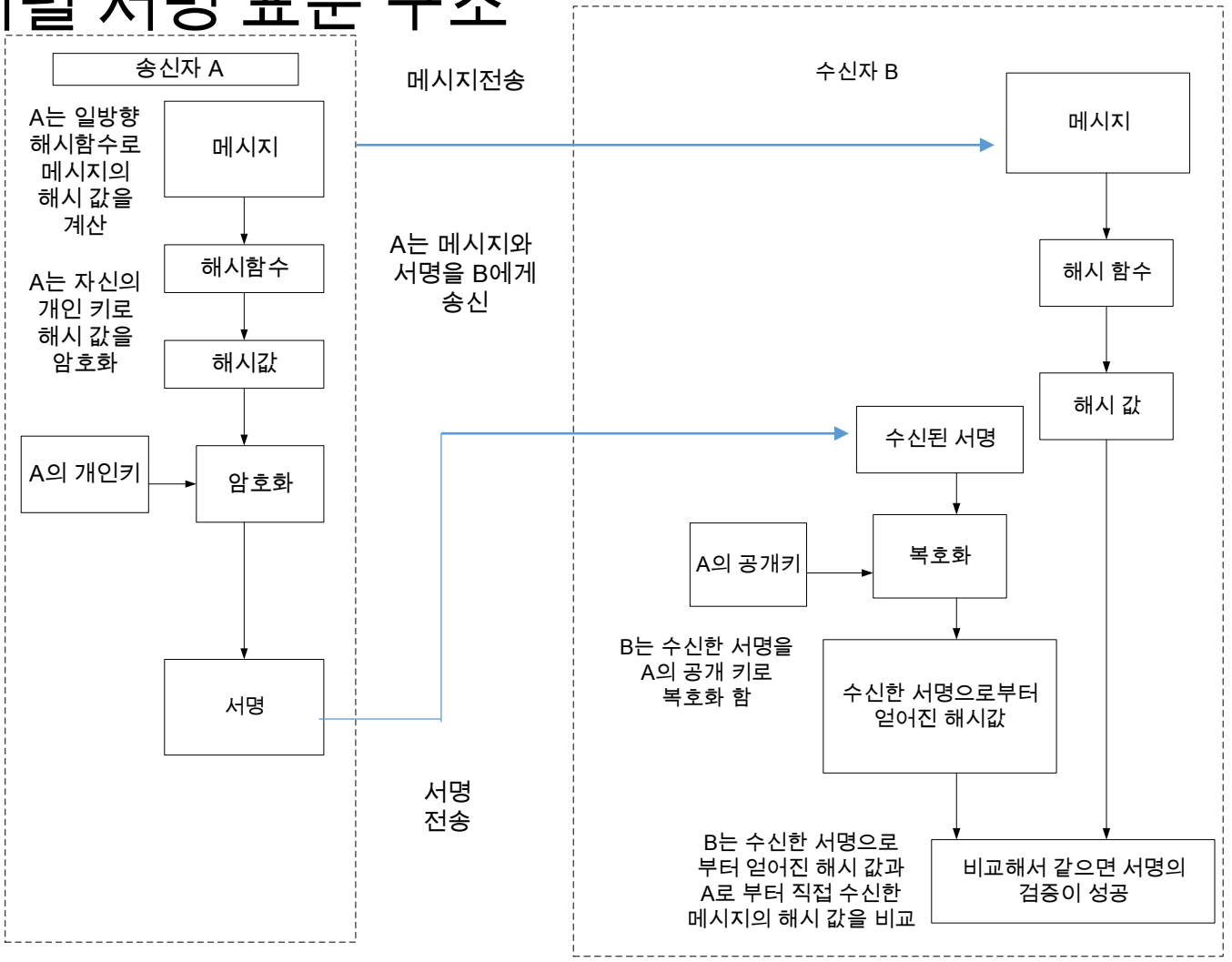


기타 공개키 암호 알고리즘

- 기타 알고리즘
  - 디지털 서명(Digital signature)
    - 송신자의 신원을 증명하는 인증 기법
    - 특징
      - 송신자가 자신의 개인키로 암호화한 메시지를 수신자가 송신자의 공개키로 복호화
      - 디지털 서명만 제공하는 기법으로 인증자(Authenticator) 블록을 생성
        - 인증자
          - 메시지의 기능을 대신하는 작은 블록
        - 인증자를 개인키로 암호화하여 메시지의 출처,무결성,순서를 확인해주는 서명 생성
      - 공개키로 복호화하기 때문에 기밀성을 보장하지 않음

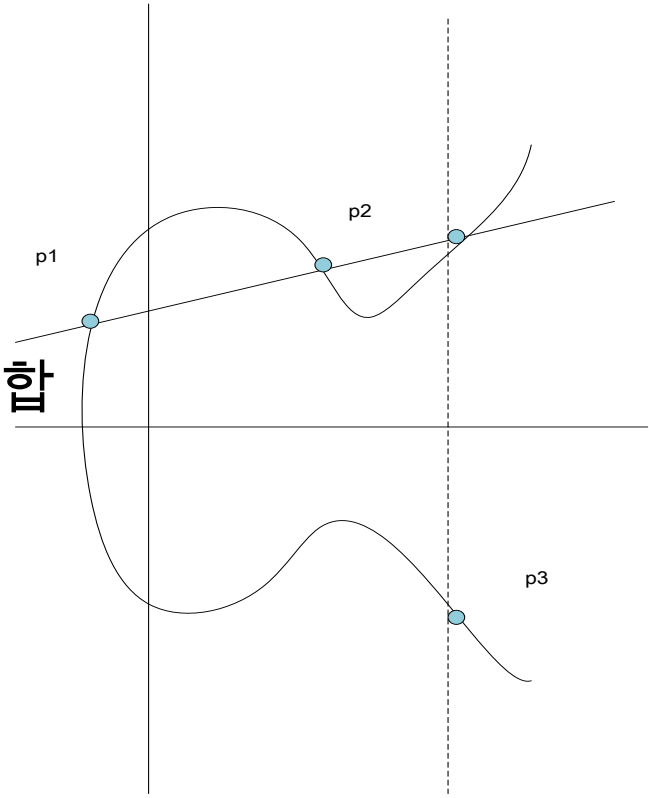
기타 공개키 암호 알고리즘

- 기타 알고리즘
  - 디지털 서명 표준 구조



기타 공개키 암호 알고리즘

- 기타 알고리즘
  - 타원 곡선 암호(ECC,Elliptic Curve Cryptography)
  - 개요
    - 1985년 Neal Koblitz와 Victor Miller가 독립적으로 제안한 타원 곡선 이론 기반 공개키 암호 방식
  - 특징
    - 타원 곡선이라고 알려진 수학적 구조에 기초하여 생성
      - 방정식$y^2 = x^3$+ax+b를 만족하는 점들의 집합
    - RSA보다 짧은 키 길이를 사용하면서 비슷한 수준의 안전성을 제공



감사합니다!