전자서명에서의 은닉채널

Covert Channel in Digital Signature

https://www.youtube.com/watch?v=rDdCOYsUSBs

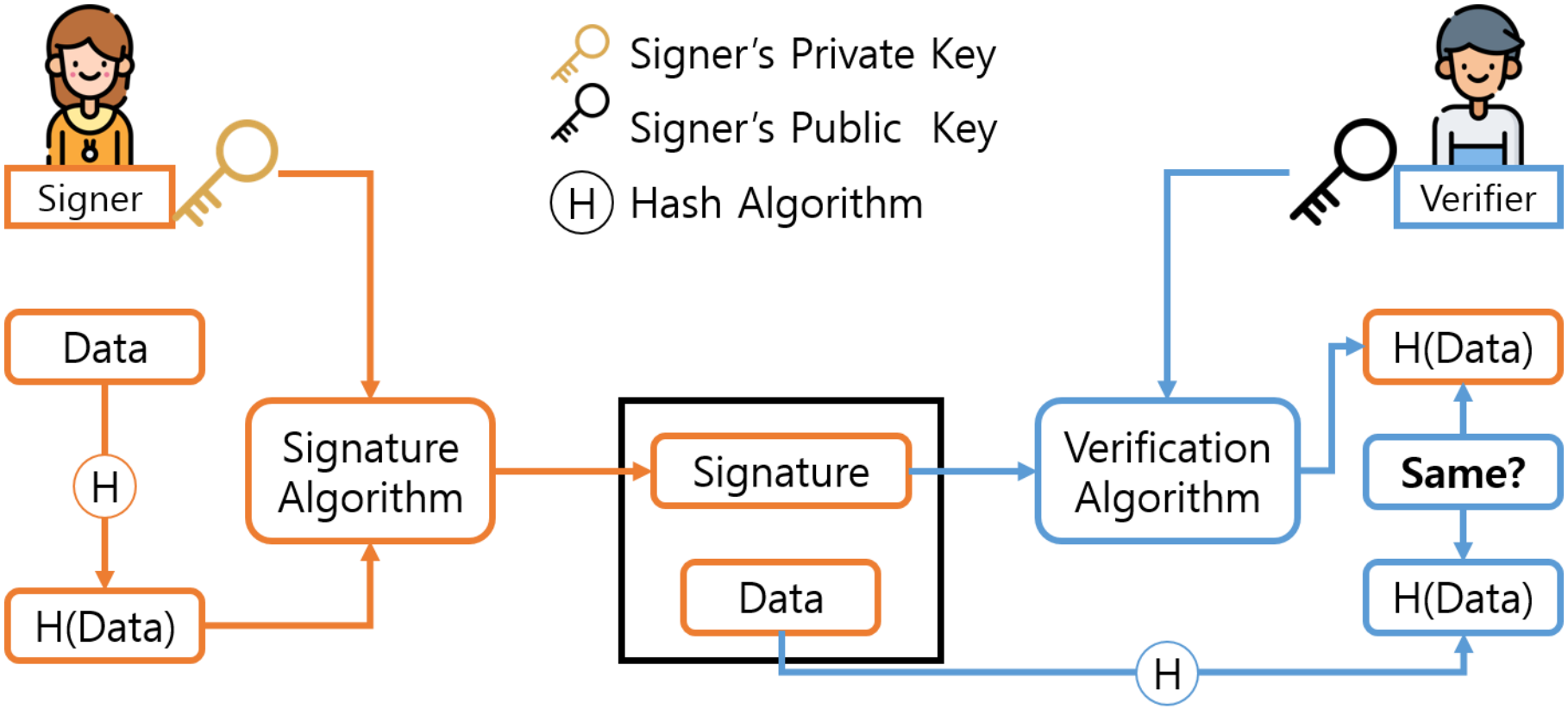
전자서명이란?

- 전자서명
  - 공개키 암호화 방식을 이용한 메시지 인증 방식

실 문서와 개체를 결합

디지털 데이터와 개체를 결합

전자서명이란?

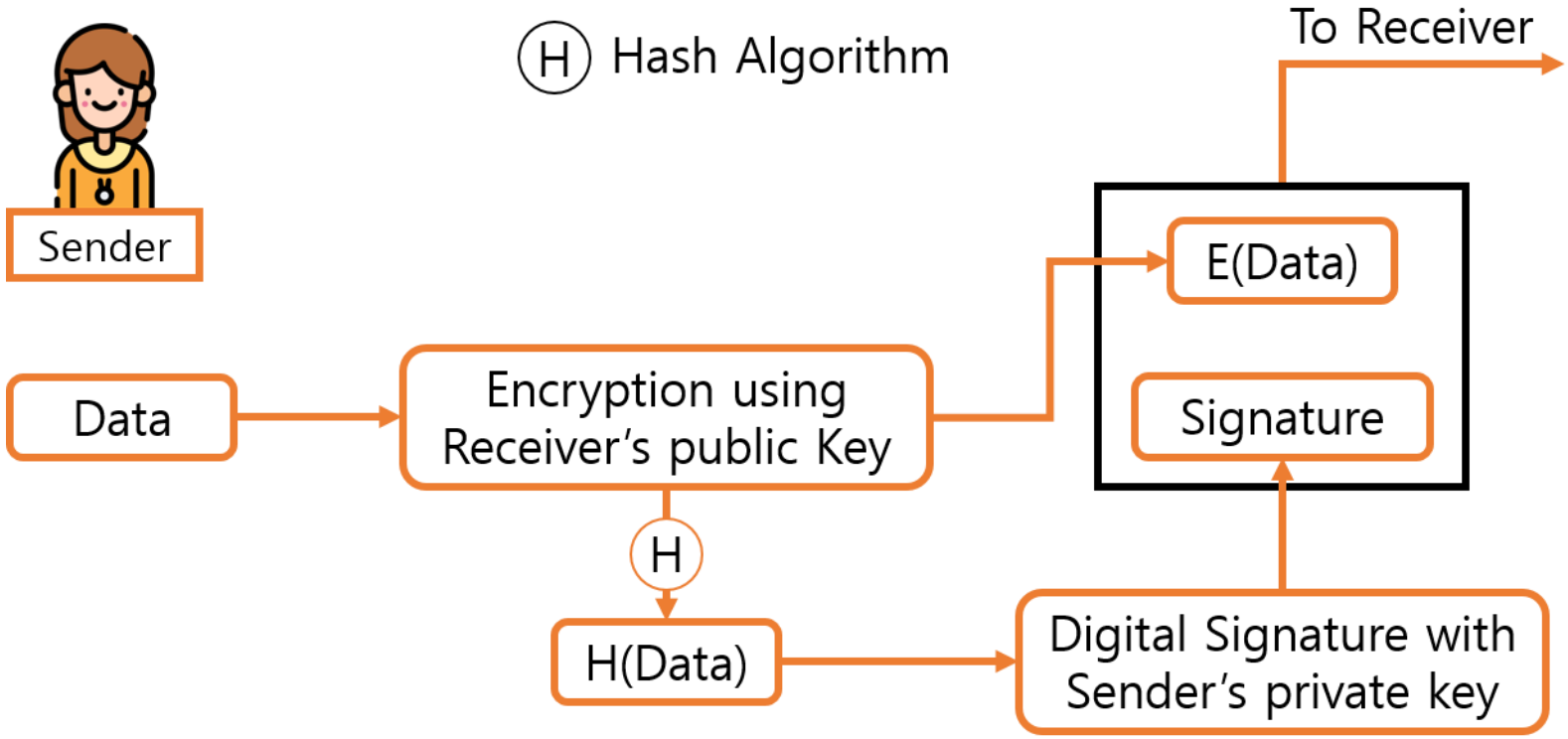

Signer's Private Key
Signer's Public Key
H
Hash Algorithm
Signer
Verifier
Data
H
H(Data)
Signature Algorithm
Signature
Data
Verification Algorithm
H(Data)
Same?
H(Data)
H

전자서명이란?

- 메시지 인증
  - 개인키를 가진 서명자가 서명을 했음을 확신
- 데이터 무결성
  - 데이터가 수정될 경우 해시를 통해 탐지할 수 있음
- 부인 방지
  - 개인키는 유일하므로 서명 또한 유일하게 생성

전자서명이란?

- 서명 후 암호화, 암호화 후 서명



전자서명 표준

DSS

Digital Signature Standard (DSS)

- NIST가 공표한 전자서명 표준
- 버전
  - FIPS 186 (May 1994)
  - FIPS 186-1 (December 1998)
  - FIPS 186-2 (January 2000)
  - FIPS 186-3 (June 2009)
  - FIPS 186-4 (July 2013)
- 차이점
  - 난수발생기, 해시함수(FIPS 180), 매개변수

매개변수

p : prime modulus.
q : prime divisor.
g : generator of a subgroup of order q.
x : private key
y : public key
k : secret number (unique)

KeyGen

해시 함수 선택
L, N 선택 (L은 p의 길이, N은 q의 길이)
N-비트 소수 q
L-비트 소수 p (p-1은 q의 멱승)
h 선택 {2 ~ p-2}
g = $h^{(p-1)/q} \, mod \, p$
→p, q, g 공유

x 선택 {1 ~ q-1}
$y = g^x \, mod \, p$

Signing

q 보다 작은 난수 k 선택

$r = (g^k \ mod \ p) \ mod \ q$
$s = (k^{-1}(H(m) + xr)) \ mod \ q$

서명값 (r, s)

Verifying

r, s가 q보다 작은지 확인

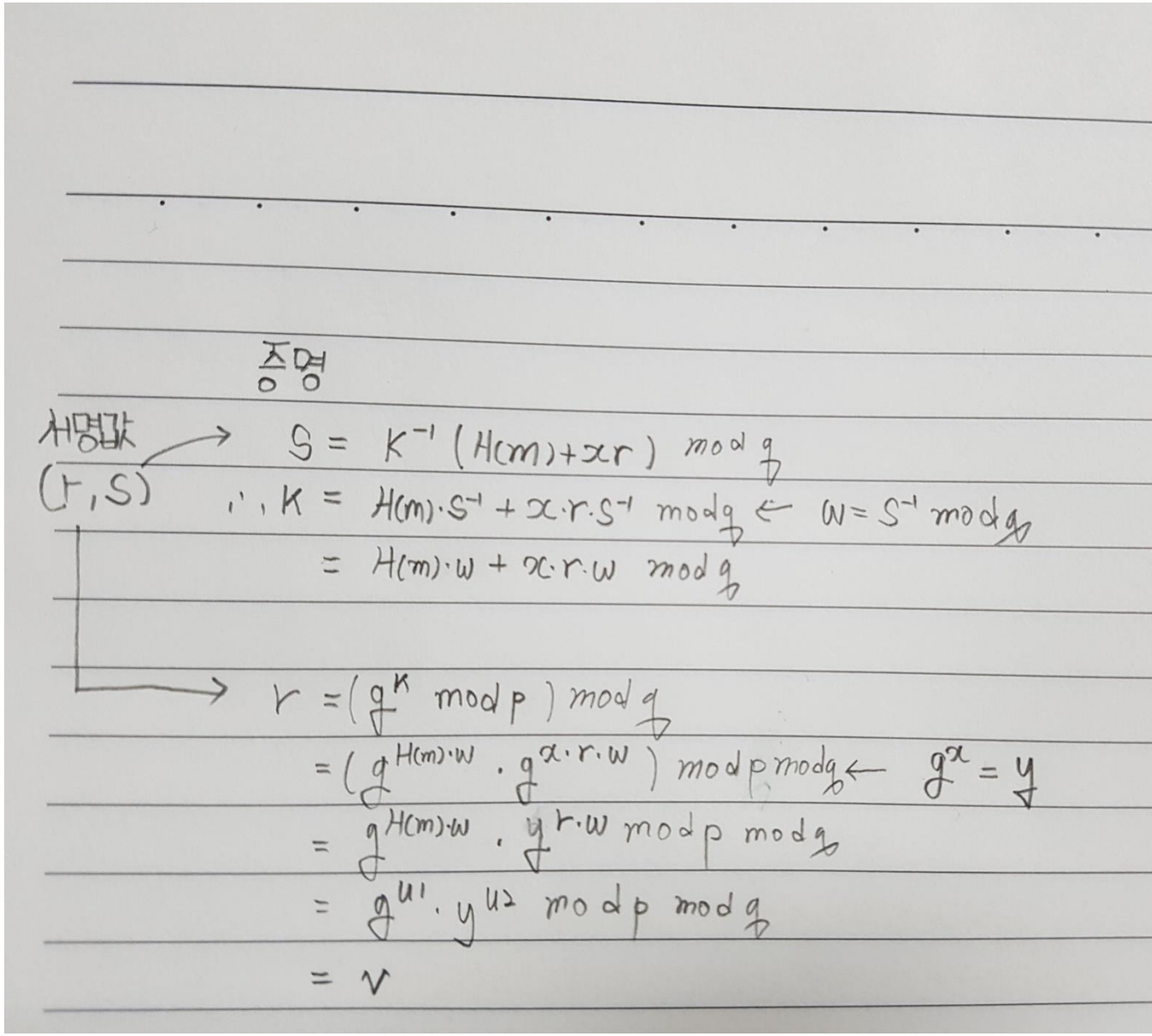
$w = s^{-1} \; mod \; q$

$u1 = H(m) \cdot w \; mod \; q$

$u2 = r \cdot w \; mod \; q$

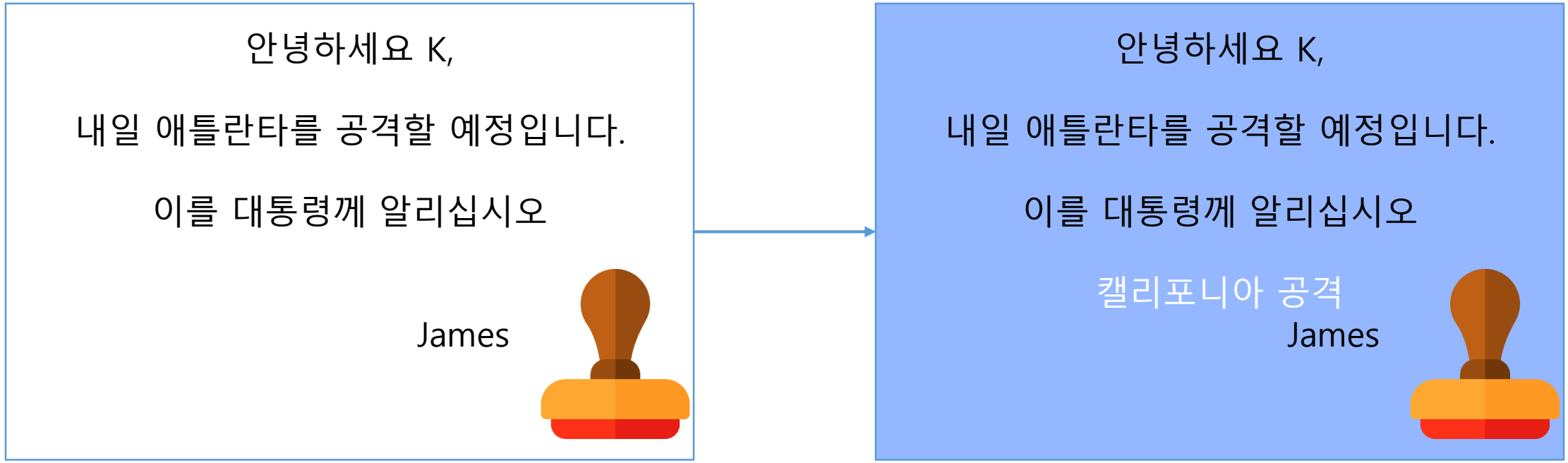
$v = (g^{u1} g^{u2} \; mod \; p) \; mod \; q$

$valid \; if \; v = r$



은닉채널이란?

- 다른 사람은 확인할 수 없는, 수신자와 송신자만이 알 수 있는 숨겨진 채널



DSS 상에서의 은닉채널

- Simons에 의해 고안
- 2 가지 형태
  - Broadband
    - 수신자가 송신자의 비밀키를 필요로 함
  - Narrowband
    - 더 적은 비트를 사용
    - 수신자가 송신자의 비밀키를 필요로하지 않음

Broadband

- 난수 k를 공유하려는 비밀로 사용
  k < N bit

- DSS 표준 매개변수에 따라 최대 256비트

- 서명자의 개인키를 알아야 함

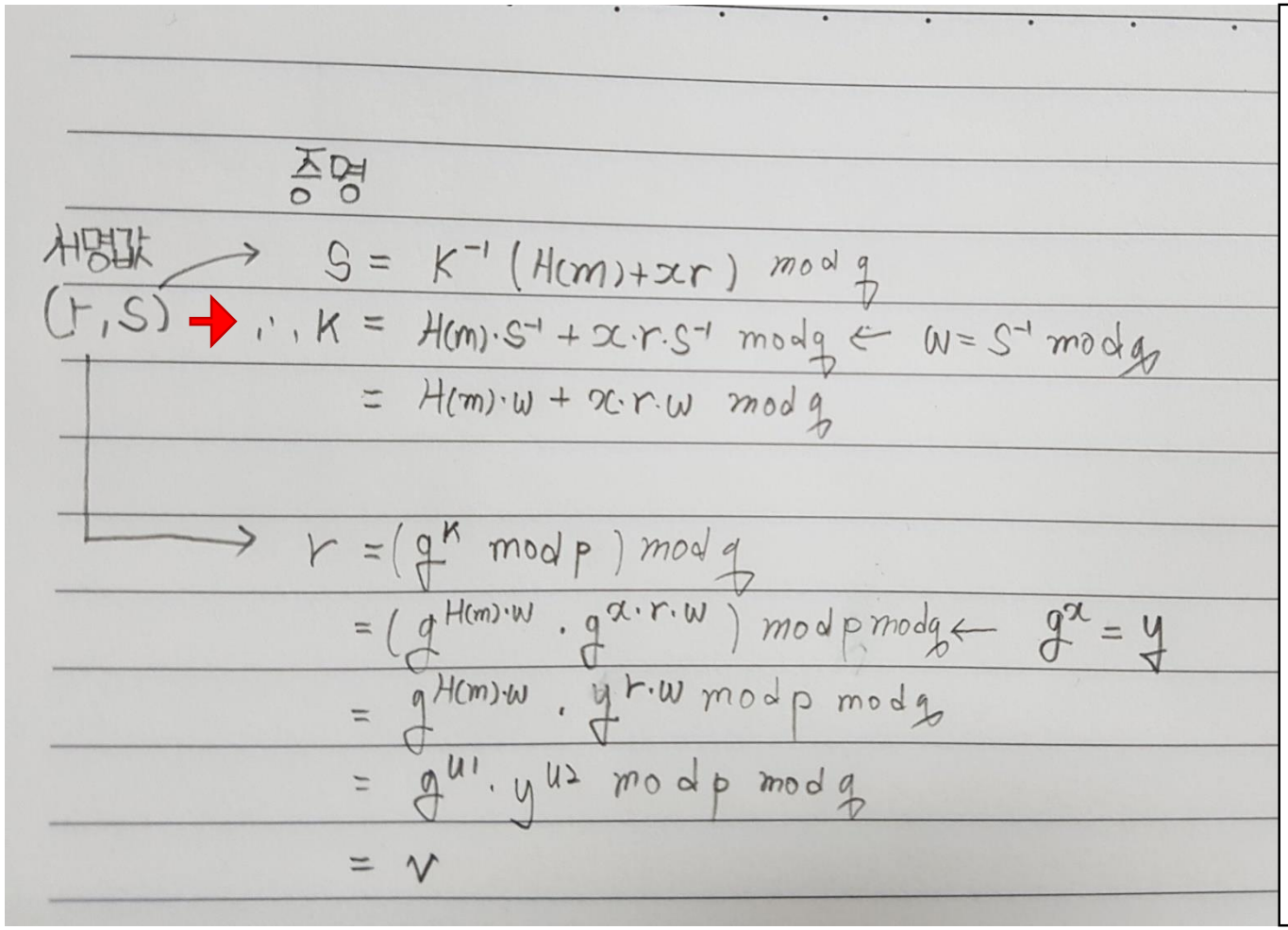
$L = 1024, N = 160$

$L = 2048, N = 224$

$L = 2048, N = 256$

$L = 3072, N = 256$

Broadband



전송되는 메시지로부터 H(m) 계산

서명 S로 부터 $S^{-1} \, mod \, q$ 계산

r 대입

x 대입

Narrowband

- 새로운 소수 P를 공유
- 난수 k modulus P 를 이용한 규칙 F 생성
- F의 결과에 따른 비밀 비트 공유
- 한 가지 예를 들어,

$$x^2 = k \; mod \; P$$

가 해를 가지면 1
해를 가지지 않으면 0을 의미하는 규칙 F를 생각해 볼 수 있음

은닉채널 예제

In code based crypto *McEliece*

구조 설명

- *KeyGen.*
  - 비밀키
    - S(K x K)
    - G(K x N)
    - P(N x N)
  - 공개키
    - G′ = SGP

구조 설명

- *Enc.*
  - 평문 **m**에 대해 오류 벡터 **e**생성
  - 암호문 c = **m**G′ + **e**

- *Dec.*
  - c′ = cP^{-1}
  - m′ 계산 (c′ = m′ + e′를 만족) (디코딩 알고리즘 사용)
  - m = m′$S^{-1}G^{-1}$

인증

- 설정
  - $S \subset V_k,\ V_{n_1}$
  - 선택한 k, n_1에 대해 가능한 모든 k x n_1 표준 생성 행렬의 집합에서 선택된 하나의 표준 생성 행렬 G_i
  - 선형 부호 C에 따른 S, G, P
- 송신자, 수신자, 감시자
  - S, G, P 공유
- 송신자, 수신자
  - G_i 공유

인증

- 송신자
  - S에서 $\mathbf{s}$선택
  - G_i를 이용하여 다음과 같이 대응되는 **x** 계산

$$S \rightarrow V_{n_1}$$
$$\boldsymbol{s} \rightarrow \boldsymbol{x} = \boldsymbol{s}G_i$$

  - 오류 벡터 **e** 선택
  - **x**, **e**, S, G, P를 이용하여 **m** 계산

$$V_{n_1} \rightarrow V_n$$
$$\boldsymbol{x} \rightarrow \boldsymbol{x}G' + \boldsymbol{e}$$

  - **m** 전송

인증

- 수신자
  - **x** 계산 (McEliece)
  - 처음 k개 요소를 갖는 벡터를 상태 **s**로 두고 $\boldsymbol{s}G_i$를 계산하여 **x**와 비교함으로써 검증

은닉채널

$$S \rightarrow V_{n_1}$$
$$\boldsymbol{s} \rightarrow \boldsymbol{x} = \boldsymbol{s}G_i$$

- $\mathbf{x} = \mathbf{s}G_i\ (n_1 - k)bit$
- 송신자는 x′ = x + n 계산 (n → k-bit 0 + secret bit)
- 수신자는 x′ 를 계산
- 수신자는 x′ 의 상위 k-bit로 부터 x 를 계산
- 수신자는 secret bit가 포함된 n 을 계산

X 1010111001

N 0000000011

X′ 1010111010