Realize your vision

T_SM

New secure scalar multiplication algorithm

CHES 2018 Rump Session

Samsung SDS
Security Research Team

삼성SDS SAMSUNG

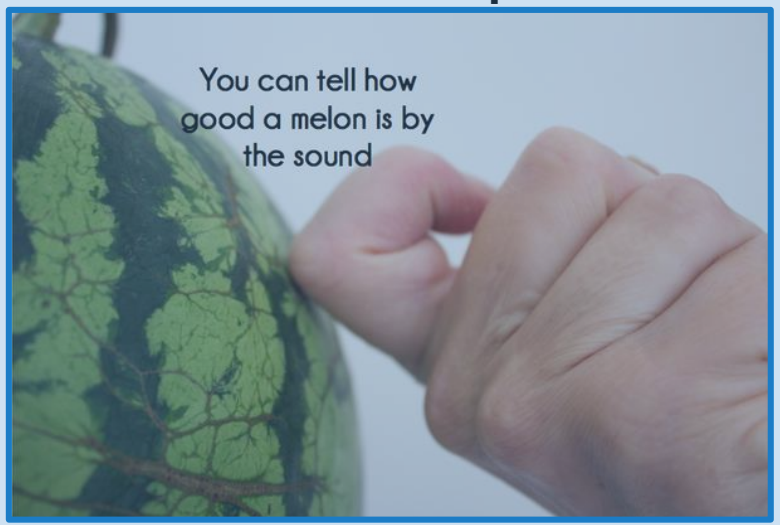
“Side Channel Attacks”

use **implementation-specific characteristics** to recover the secret in a target cryptosystem

Real World
How to pick a ripe watermelon?

- Appearance
  - Surface color: become dull from light tone
  - Belly: turns from white to creamy white or yellow

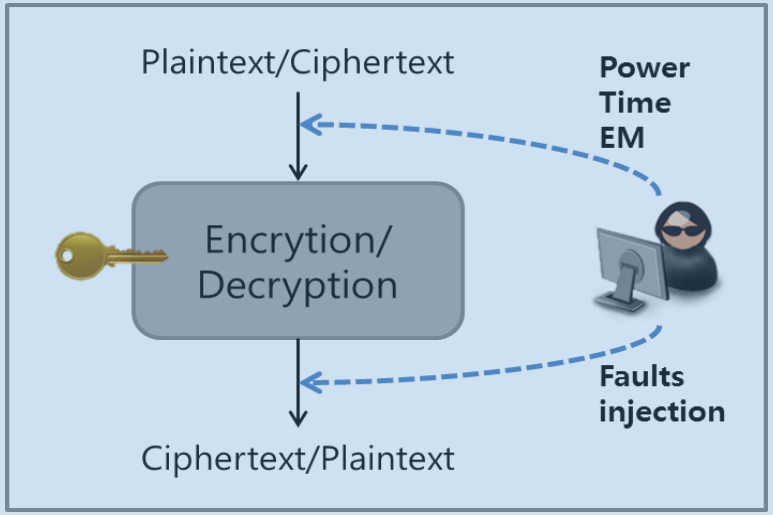
- **Sound [Side channel information]**
  - Make a **hollow and low-pitched** sound



Crypto World
How to pick a SECRET in Crypto systems?

- **Black-box model**
  - Uses only input/output information

- **Gray-box model [Side channel attacks]**
  - Uses side information such as **time, power, EM** etc



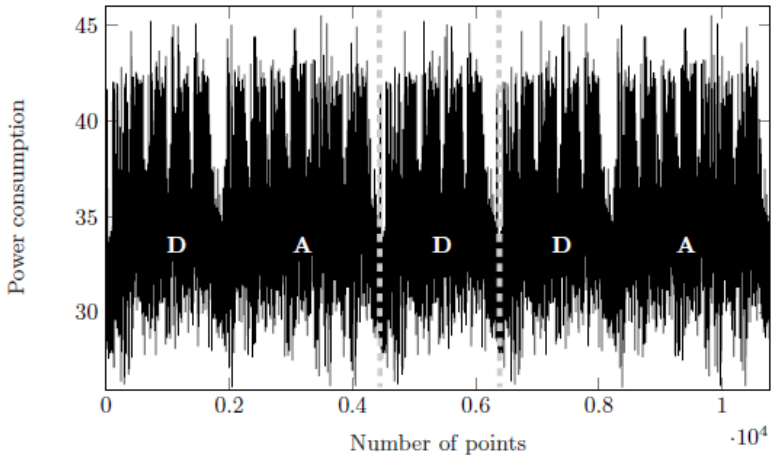
“More Powerful Attacks”

Especially, **single-trace attacks** are more practical than multi-trace attacks against **the Scalar Multiplication**

Simple Power Attack(SPA) [`99]

Exploits **the patterns of secret scalar bit-dependent conditional branches** from a single-trace

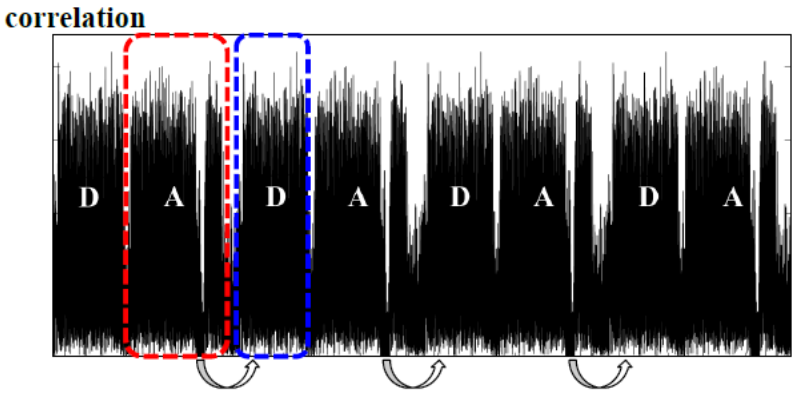
- **Scalar multiplication**
  . Binary method



Collision Attack(CA) [`01]

A kind of higher-order DPA based on **the interrelationships between data**

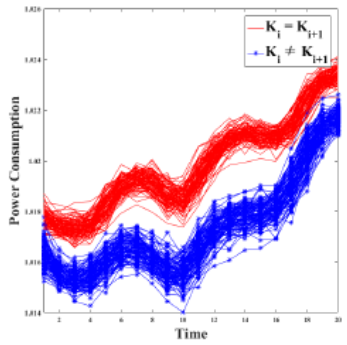
- **Scalar multiplication**
  . D&A Always, Montgomery, Joye’s Add, Coron’s alg.
  . M-ary, Sliding Window, NAF, Moller alg.



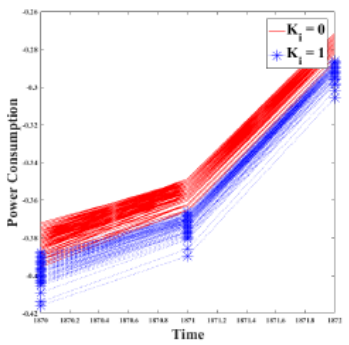
Key-Bit dependent Attack(KBA) [`17]

Uses the leakage which occurs in a **secret scalar bit check phase**

- **Scalar multiplication**
  . All method with scalar bit check phase



(a) Classification according to hamming distance between k_i and k_{i+1} (in case of hardware implementation)



(b) Classification according to hamming weight of k_i (in case of software implementation)

"Scalar Multiplication is the Main operation of ECDSA"

ECDSA* has been used in various protocols, browsers and crypto libraries. **However, there are many attack methods**
(**Browser & OS**: Firefox, Safari, IE, Chrome, Opera, Android, iOS/ **Library**: OpenSSL, boring SSL, GnuTLS, BC, Botan / **Other**: FIDO, Blockchain)

ECDSA scheme

Signature generation phase

- Secret a & message m & base point G
- Choose a **random** integer k with $1 < k < n$

$$R = (k \cdot G) = (x, y)$$
$$s = k^{-1}(m + ax) \bmod n$$
$$Sign = (m, R, s)$$

Signature verification phase

- $Q = a \cdot G$

$$u_1 = s^{-1} \cdot m \bmod n$$
$$u_2 = s^{-1} \cdot x \bmod n$$
$$u_1 \cdot G + u_2 \cdot Q \stackrel{?}{=} R$$

Main Scalar Multiplication methods

Method	SPA	CA	KBA
D&A Always [`99]	Secure	Insecure	Insecure
Montgomery [`02]	Secure	Insecure	Insecure
Joye's Add [`07]	Secure	Insecure	Insecure
Coron's Alg. [`99]	Secure	Insecure	Insecure
Moller window [`01]	Secure	Insecure	Insecure
Width-w NAF [`03]	Secure	Insecure	Insecure
mLSB-set comb [`14]	Secure	Insecure	Insecure
Ours (T_SM) [`18]	**Secure**	**Secure**	**Secure**

* **ECDSA**: Elliptic Curve Digital Signature Algorithm [**NIST FIPS 186-4**]

“The first and unique countermeasure”

SDS Scalar Multiplication method (**T_SM**) counteracts **the powerful side channel attacks**

. T_SM: Sequence Subset-based Scalar Multiplication method

Security

- **Theoretical Security Analysis** was completed
- **Experimental Security Analysis** was completed with **Kookmin Univ.**
- **3rd party Security Evaluation** finished **(ISPEC 2018)**

Efficiency

- **Do not any scalar recoding**
- **Need only point addition operations** (without point doubling operations)
- **Use the same size of secret parameter** (with some public parameters)

Usability

- **Random scalar multiplication** : ECDSA
- **Chosen scalar multiplication** : EdDSA*, FIPS Validation

* **EdDSA**: Edwards-curve Digital Signature Algorithm [**IETF RFC 18032**]

SAMSUNG SDS

www.samsungsds.com