ECTester: Reverse-engineering side-channel countermeasures of ECC implementations

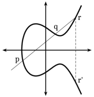
Vojtech Suchanek, Jan Jancar, Jan Kvapil, Petr Svenda, Łukasz Chmielewski

CROCS

CHES 2025

Vibe of the talk

- Implementations of elliptic curve cryptography (ECC)

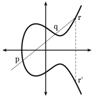


$$E : y^2 = x^3 + ax + b$$
$$[k]P = P + \cdots + P$$

ECDH secret: Hash($[k]P$)

Vibe of the talk

- Implementations of elliptic curve cryptography (ECC)



$$E : y^2 = x^3 + ax + b$$
$$[k]P = P + \cdots + P$$

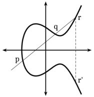
ECDH secret: Hash($[k]P$)

- ECTester: toolkit for a black-box testing of ECC implementations

&$#%& $\longrightarrow$ 

Vibe of the talk

- Implementations of elliptic curve cryptography (ECC)



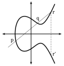
$$E : y^2 = x^3 + ax + b$$
$$[k]P = P + \cdots + P$$

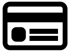
ECDH secret: Hash($[k]P$)

- ECTester: toolkit for a black-box testing of ECC implementations

&$#%& $\longrightarrow$ 

- Lying to JavaCards to reverse-engineer (RE) randomization techniques protecting a secret scalar.



$\longrightarrow$  $\longrightarrow$ 

&$#%&

Previously at CHES 2024 ...

pyecsca: Reverse engineering black-box elliptic curve cryptography via side-channel analysis. CHES 2024.

Motivation:

- Side-channel attacks on ECC often assume a white-box attacker
- Real-world implementations of ECC are usually black-box (smartcards, HSM, TPM, or crypto-wallets)

Previously at CHES 2024 ...

pyecsca: Reverse engineering black-box elliptic curve cryptography via side-channel analysis. CHES 2024.

Results:

- Analysis of 18 open-source libraries showed a variety of implementation decisions
  - Elliptic curve E
  - Coordinate representation of points $P \in E$
  - Addition formulas for $P + Q$
  - Scalar multiplier for $[k]P$

  ...

Previously at CHES 2024 ...

pyecsca: Reverse engineering black-box elliptic curve cryptography via side-channel analysis. CHES 2024.

Results:

- Analysis of 18 open-source libraries showed a variety of implementation decisions
- Enumeration of the space of ECC implementations yielded $> 139\,489$ possibilities $\Rightarrow$ hard to guess!
- **pyecsca** toolkit for automatic RE of the scalar multiplier and the coordinate system

Previously at CHES 2024 ...

pyecsca: Reverse engineering black-box elliptic curve cryptography via side-channel analysis. CHES 2024.

Limitations:

- Assumes that we can set the domain parameters
- Scalar randomization breaks the RE methods
- Not demonstrated on real-world black-box devices

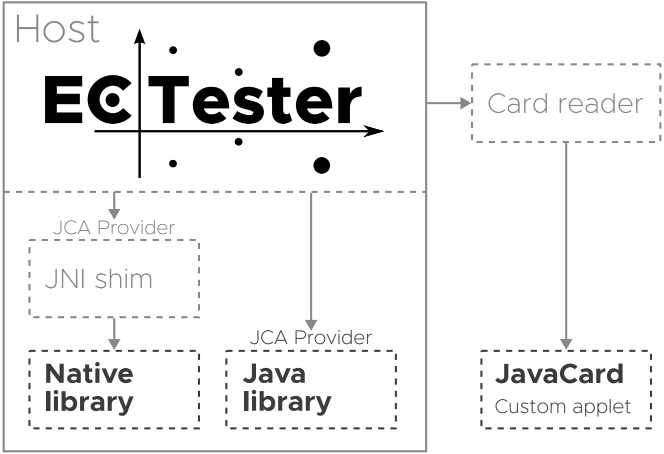
Currently at CHES 2025

ECTester: Reverse-engineering side-channel countermeasures of ECC implementations. CHES 2025.

Contributions:

- ECTester: toolkit for testing ECC libraries and JavaCards
- Techniques for RE of scalar randomization countermeasures *without side-channel measurements*
- RE of countermeasures on 13 JavaCards certified under CC or FIPS 140

ECTester



Supported libraries

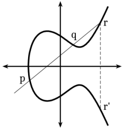
- BoringSSL
- Botan
- BouncyCastle
- Crypto++
- Intel Cryptography Primitives
- libgcrypt
- LibreSSL
- libtomcrypt
- mbedTLS
- Nettle
- OpenSSL
- SunEC

Supported smartcards

Any JavaCard >= 2.2.1

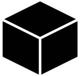
crocs-muni/ECTester

ECTester



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

$\longrightarrow$



$Q = [k]P$

$\longrightarrow \quad Q$

ECTester tests:

• Invalid curve attack	$P \in E$
• Small subgroup attack	$\text{ord}(P) = n$
• Malformed signatures	$r, s \neq 0$
• Composite curve order	n is prime
• Anomalous curves, supersingular curves	$p \notin \{\#E - 1, \#E\}$
...	...

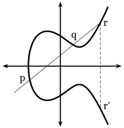
ECTester: testing input validation

Card	ord P	ord G	prime n
NXP 1	✓	✓	✓
NXP 2	✓	✓	✓
NXP 3	✓	✓	✓
NXP 4	✓	✓	✓
NXP 6	✓	✓	✓
NXP 9	✓	✓	✓
Infineon 1	✓	✓	✓
Infineon 2	✓	✓	✗
Athena	✗	✗	✗
G&D	✗	✗	✓
TaiSYS	✓	✓	✓
Feitian 1	✓	✓	✓
Feitian 2	✓	✓	✓

Library	ord P	ord G	prime n
BoringSSL	✓	✓	✓
Botan	✓	✓	✓
BouncyCastle	✓	✓	✓
Crypto++	✓	✓	✓
Intel Crypto	✓	✗	✓
libgcrypt	✗	✗	✗
LibreSSL	✓	✗	✓
libtomcrypt	✓	✗	✗
mbedTLS	✓	✓	✓
Nettle	✗	✗	✗
OpenSSL	✓	✗	✓
SunEC	✗	✗	✗

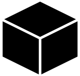
✓= success, ✗= fail to pass validation with invalid parameters

Reverse-engineering using invalid parameters



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

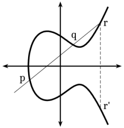
$\longrightarrow$



$$Q = [k]P$$

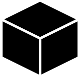
$\longrightarrow$ Q

Reverse-engineering using invalid parameters



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

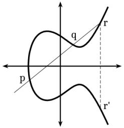
$\longrightarrow$



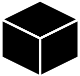
$\longrightarrow$ Q

$$k \mapsto \rho(k)$$
$$Q = [\rho(k)]P = [k]P$$

Reverse-engineering using invalid parameters



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

$\longrightarrow$  $\longrightarrow$ Q

$$k \mapsto \rho(k)$$
$$Q = [\rho(k)]P = [k]P$$

- The scalar randomization $k \mapsto \rho(k)$ prohibits side-channel attacks and RE
- Can we RE the randomization algorithm ρ
- Can we recover the random mask used for $\rho(k)$?

Usual suspects

Group scalar randomization

function $\textsc{Mult}(P, k)$
  $r \xleftarrow{\$} \{0, 1, \ldots, 2^b\}$
  return $[k + rn]P$

Additive splitting

function $\textsc{Mult}(P, k)$
  $r \xleftarrow{\$} \mathbb{Z}_n^*$
  return $[k - r]P + [r]P$

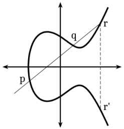
Euclidean splitting

function $\textsc{Mult}(P, k)$
  $r \xleftarrow{\$} \{0, \ldots, 2^{\lfloor \log_2(n)/2 \rfloor}\}$
  $S \leftarrow [r]P$
  $k_1 \leftarrow k \mod r$
  $k_2 \leftarrow \left\lfloor \frac{k}{r} \right\rfloor$
  return $[k_1]P + [k_2]S$

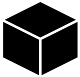
Multiplicative splitting

function $\textsc{Mult}(P, k)$
  $r \xleftarrow{\$} \{0, 1, \ldots, 2^b\}$
  $S \leftarrow [r]P$
  return $[kr^{-1} \mod n]S$

Test $3n$: **reverse-engineering** ρ



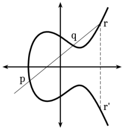
$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

$\longrightarrow$  $\longrightarrow$ Q

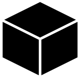
$$k \mapsto \rho(k)$$
$$Q = [\rho(k)]P = [k]P$$

Test $3n$: Select a curve with $\#E = 3n$, a point P of order $3n$ and claim cofactor$= 1$

Test $3n$: reverse-engineering ρ



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

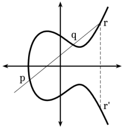
$\longrightarrow$  $\longrightarrow$ Q

$$k \mapsto \rho(k)$$
$$Q = [\rho(k)]P = [k]P$$

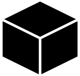
Test $3n$: Select a curve with $\#E = 3n$, a point P of order $3n$ and claim cofactor$= 1$

- GSR: $[\rho(k)]P = [k + rn]P \in \{[k]P, [k+n]P, [k+2n]P\}$ with prob. dist. $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$

Test $3n$: **reverse-engineering** ρ



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

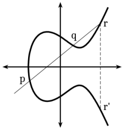
$\longrightarrow$  $\longrightarrow$ Q

$$k \mapsto \rho(k)$$
$$Q = [\rho(k)]P = [k]P$$

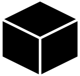
Test $3n$: Select a curve with $\#E = 3n$, a point P of order $3n$ and claim cofactor$= 1$

- GSR: $[\rho(k)]P = [k + rn]P \in \{[k]P, [k+n]P, [k+2n]P\}$ with prob. dist. $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$
- Multiplicative: $\frac{2}{9}, \frac{2}{9}, \frac{5}{9}$

Test $3n$: **reverse-engineering** ρ



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

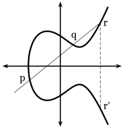
$\longrightarrow$  $\longrightarrow$ Q

$$k \mapsto \rho(k)$$
$$Q = [\rho(k)]P = [k]P$$

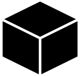
Test $3n$: Select a curve with $\#E = 3n$, a point P of order $3n$ and claim cofactor$= 1$

- GSR: $[\rho(k)]P = [k + rn]P \in \{[k]P, [k+n]P, [k+2n]P\}$ with prob. dist. $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$
- Multiplicative: $\frac{2}{9}, \frac{2}{9}, \frac{5}{9}$
- Euclidean: $1, 0, 0$

Test $3n$**: reverse-engineering** ρ



$E/\mathbb{F}_p : y^2 = x^3 + ax + b$
$P \in E, \quad k < n$
$\#E = n \cdot \text{cofactor}$ $\longrightarrow$



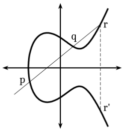
$\longrightarrow \quad Q$

$k \mapsto \rho(k)$
$Q = [\rho(k)]P = [k]P$

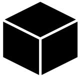
Test $3n$: Select a curve with $\#E = 3n$, a point P of order $3n$ and claim cofactor$= 1$

- GSR: $[\rho(k)]P = [k + rn]P \in \{[k]P, [k+n]P, [k+2n]P\}$ with prob. dist. $\frac{1}{3}, \frac{1}{3}, \frac{1}{3}$
- Multiplicative: $\frac{2}{9}, \frac{2}{9}, \frac{5}{9}$
- Euclidean: $1, 0, 0$
- Additive: $\frac{1}{2}, \frac{1}{2}, 0$

Test $n + \epsilon$: recovering the mask



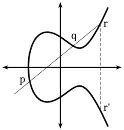
$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

$\longrightarrow$  $\longrightarrow$ Q

$$k \mapsto k + rn$$
$$Q = [k + rn]P = [k]P$$

Test $n + \epsilon$: Select a curve with composite n, point P of order n and claim $\#E = n + \epsilon$.

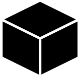
Test $n + \epsilon$: recovering the mask



$E/\mathbb{F}_p : y^2 = x^3 + ax + b$

$P \in E, \quad k < n$

$\#E = n \cdot \text{cofactor}$

$\longrightarrow$  $\longrightarrow \quad Q$

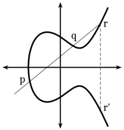
$k \mapsto k + rn$

$Q = [k + rn]P = [k]P$

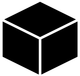
Test $n + \epsilon$: Select a curve with composite n, point P of order n and claim $\#E = n + \epsilon$.

- $Q = [k + r(n + \epsilon)]P = [k + r\epsilon]P$

Test $n+\epsilon$: recovering the mask



$E/\mathbb{F}_p : y^2 = x^3 + ax + b$
$P \in E, \quad k < n$
$\#E = n \cdot \text{cofactor}$

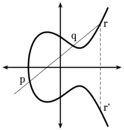
$\longrightarrow$  $\longrightarrow$ Q

$k \mapsto k + rn$
$Q = [k + rn]P = [k]P$

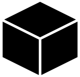
Test $n+\epsilon$: Select a curve with composite n, point P of order n and claim $\#E = n + \epsilon$.

- $Q = [k + r(n+\epsilon)]P = [k + r\epsilon]P$
- Solve the discrete logarithm problem for P, Q to find $d = k + r\epsilon$

Test $n + \epsilon$: recovering the mask



$$E/\mathbb{F}_p : y^2 = x^3 + ax + b$$
$$P \in E, \quad k < n$$
$$\#E = n \cdot \text{cofactor}$$

$\longrightarrow$  $\longrightarrow$ Q

$$k \mapsto k + rn$$
$$Q = [k + rn]P = [k]P$$

Test $n + \epsilon$: Select a curve with composite n, point P of order n and claim $\#E = n + \epsilon$.

- $Q = [k + r(n + \epsilon)]P = [k + r\epsilon]P$
- Solve the discrete logarithm problem for P, Q to find $d = k + r\epsilon$
- Simply compute $r = \frac{d-k}{\epsilon}$

RE results

Card	Target	$3n$	Composite	$k=10$	EPA	ρ	Mask
NXP 1	Derive	$0.34, 0.33, 0.32$	100%	86%	¿0	GSR	✗
	Sign	$0.31, 0.31, 0.38$	83%	-		GSR	160
	Keygen	$0.32, 0.33, 0.35$	100%	-		GSR	160
NXP 3	Derive	$0.33, 0.32, 0.35$	100%	98%		GSR	32
	Sign	$0.31, 0.30, 0.39$	85%	-		GSR	160
	Keygen	✗	✗	-		✗	✗
NXP 4	Derive	$0.22, 0.56, 0.22$	82%	100%		Mult	64
	Sign	$0.23, 0.23, 0.54$	-	-		Mult	?
	Keygen	✗	✗	-		✗	✗
NXP 6	Derive	$0, 0, 1$	100%	100%	▭	Euc.?	2
	Sign	$0, 0.52, 0.48$	71%	-		Euc.?	2
	Keygen	$0, 0.51, 0.49$	100%	-		Euc.?	2
...	...	...	...	...	...	...	...

Impact, limitations and discussion

* Scalar randomization countermeasures are no longer secret under our RE techniques

Impact, limitations and discussion

- Scalar randomization countermeasures are no longer secret under our RE techniques
- We can mount the learning phase of profiled attacks via Test $n + \epsilon$

Impact, limitations and discussion

- Scalar randomization countermeasures are no longer secret under our RE techniques
- We can mount the learning phase of profiled attacks via Test $n + \epsilon$
- Path to RE of the scalar multiplier with scalar randomization

Impact, limitations and discussion

- Scalar randomization countermeasures are no longer secret under our RE techniques
- We can mount the learning phase of profiled attacks via Test $n + \epsilon$
- Path to RE of the scalar multiplier with scalar randomization
- ⚠ Cards with proper validation or internal fault detection

Impact, limitations and discussion

- ✸ Scalar randomization countermeasures are no longer secret under our RE techniques
- ✸ We can mount the learning phase of profiled attacks via Test $n+\epsilon$
- ✸ Path to RE of the scalar multiplier with scalar randomization
- ⚠ Cards with proper validation or internal fault detection
- ⚠ Combination of countermeasures

Impact, limitations and discussion

- Scalar randomization countermeasures are no longer secret under our RE techniques
- We can mount the learning phase of profiled attacks via Test $n + \epsilon$
- Path to RE of the scalar multiplier with scalar randomization
- Cards with proper validation or internal fault detection
- Combination of countermeasures
- Proper validation comes with a high cost

Impact, limitations and discussion

- Scalar randomization countermeasures are no longer secret under our RE techniques
- We can mount the learning phase of profiled attacks via Test $n + \epsilon$
- Path to RE of the scalar multiplier with scalar randomization
- Cards with proper validation or internal fault detection
- Combination of countermeasures
- Proper validation comes with a high cost
- Restrict API to standard named curves

Stay tuned for episode 3!

Thank you!

Explore our tools

Come grab a sticker after the talk!

pyecsca

pyecsca.org

ECTester

crocs-muni/ECTester

CROCS

crocs.fi.muni.cz