A formula for disaster: a unified approach to elliptic curve special-point-based attacks

Vladimir Sedlacek[1,2] Jesus-Javier Chi-Dominguez[3,4] Jan Jancar[1] Billy Bob Brumley[3]

[1]Centre for Research on Cryptography and Security, Masaryk University, Brno, Czech Republic

[2]Ca'Foscari University, Venice, Italy

[3]Tampere University, Finland

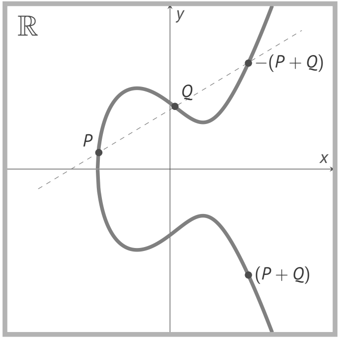
[4]Cryptography Research Centre, Technology Innovation Institute, Abu Dhabi, UAE

Asiacrypt 2021, December 7

What is ECC?

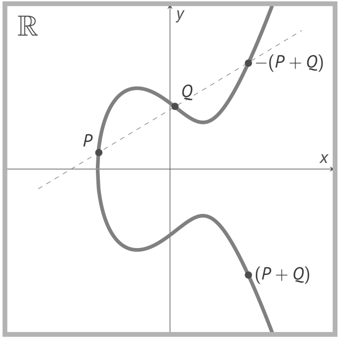
Elliptic Curve Cryptography

Based on the discrete log: given P and $[k]P$, recover the private key k



Elliptic Curve Cryptography

Based on the discrete log: given P and $[k]P$, recover the private key k

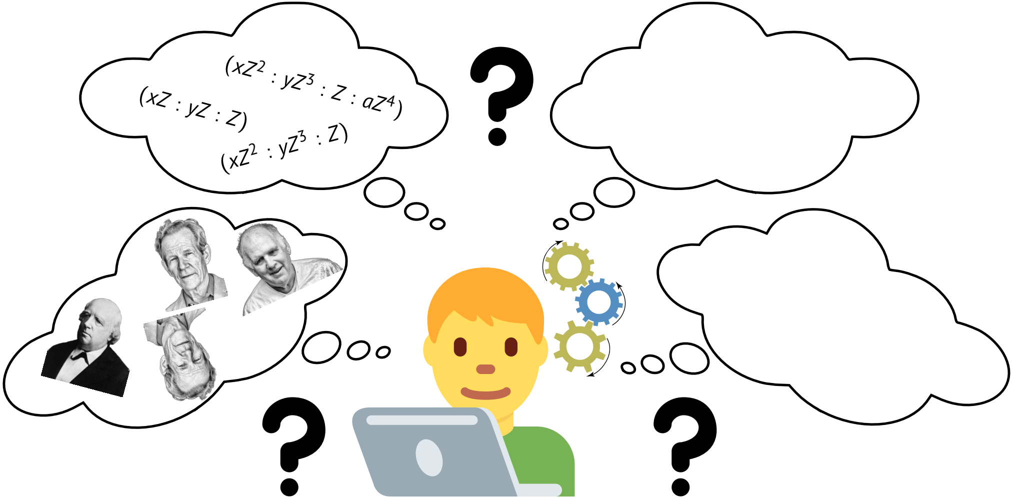


Textbook affine addition: $P = (X_1 : Y_1 : 1), Q = (X_2 : Y_2 : 1) \implies P + Q = (X_3 : Y_3 : 1)$,
where $X_3 = \lambda^2 - X_1 - X_2$, $Y_3 = \lambda(X_1 - X_3) - Y_1$, $\lambda = \frac{Y_1 - Y_2}{X_1 - X_2}$.

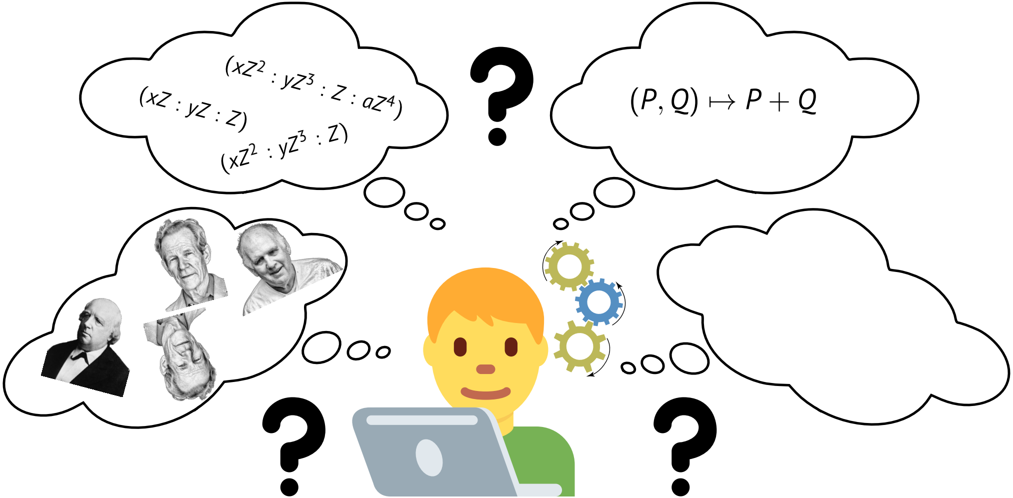
Implementing ECC

Implementing ECC

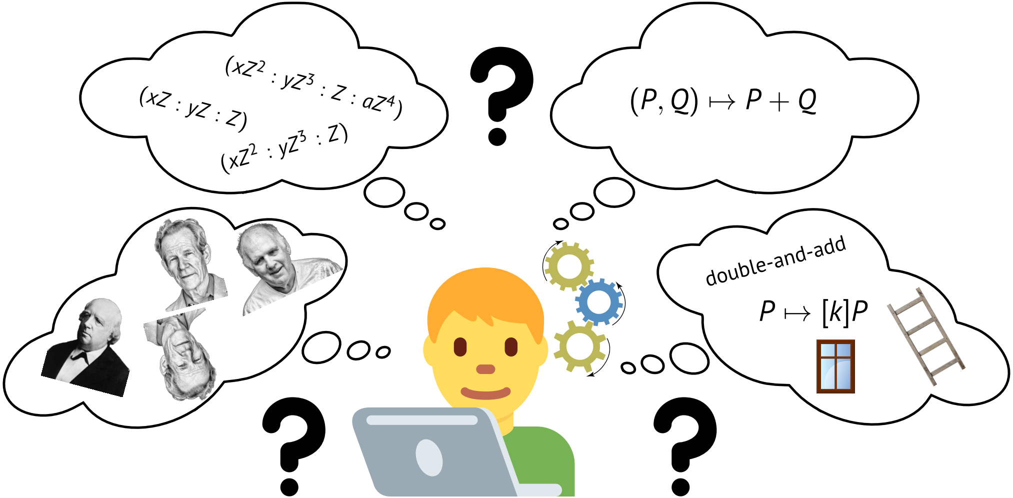
Implementing ECC



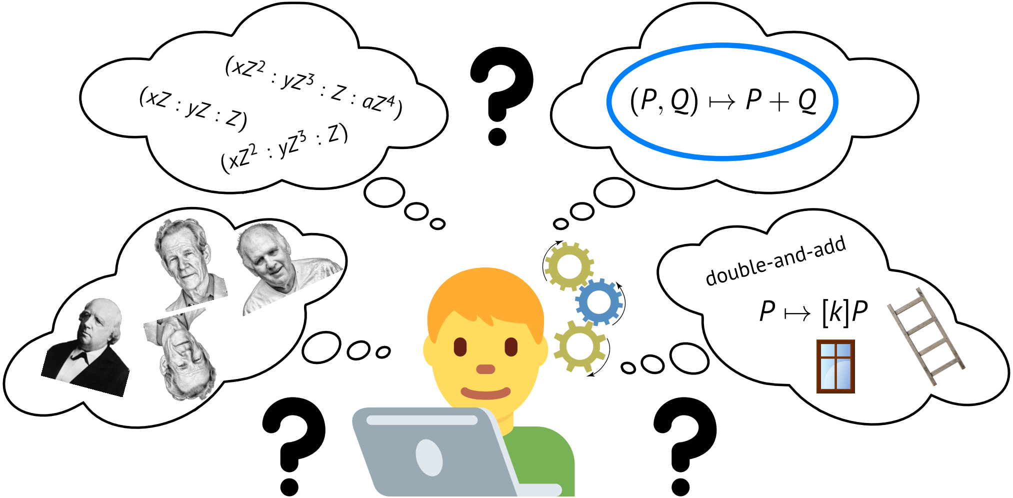
Implementing ECC



Implementing ECC



Implementing ECC



Formula example: add-2007-bl

Input: $P = (X_1 : Y_1 : Z_1), \quad Q = (X_2 : Y_2 : Z_2)$

Output: $P + Q = (X_3 : Y_3 : Z_3)$

$$
\begin{aligned}
U_1 &= X_1 \cdot Z_2 \\
U_2 &= X_2 \cdot Z_1 \\
S_1 &= Y_1 \cdot Z_2 \\
S_2 &= Y_2 \cdot Z_1 \\
ZZ &= Z_1 \cdot Z_2 \\
T &= U_1 + U_2 \\
TT &= T^2 \\
M &= S_1 + S_2 \\
t_0 &= ZZ^2 \\
t_1 &= a \cdot t_0 \\
t_2 &= U_1 \cdot U_2
\end{aligned}
$$

$$
\begin{aligned}
t_3 &= TT - t_2 \\
R &= t_3 + t_1 \\
F &= ZZ \cdot M \\
L &= M \cdot F \\
LL &= L^2 \\
t_4 &= T + L \\
t_5 &= t_4^2 \\
t_6 &= t_5 - TT \\
G &= t_6 - LL \\
t_7 &= R^2 \\
t_8 &= 2 \cdot t_7
\end{aligned}
$$

$$
\begin{aligned}
W &= t_8 - G \\
t_9 &= F \cdot W \\
X_3 &= 2 \cdot t_9 \\
t_{10} &= 2 \cdot W \\
t_{11} &= G - t_{10} \\
t_{12} &= 2 \cdot LL \\
t_{13} &= R \cdot t_{11} \\
Y_3 &= t_{13} - t_{12} \\
t_{14} &= F^2 \\
t_{15} &= F \cdot t_{14} \\
Z_3 &= 4 \cdot t_{15}
\end{aligned}
$$

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✗)

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✘)
- Hidden assumptions $\rightarrow$ unexpected behavior

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✘)
- Hidden assumptions $\rightarrow$ unexpected behavior
- Issues in the past:

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✗)
- Hidden assumptions $\rightarrow$ unexpected behavior
- Issues in the past:
  - NSS, OpenSSL, BoringSSL, Python fastecdsa

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✗)
- Hidden assumptions $\rightarrow$ unexpected behavior
- Issues in the past:
  - NSS, OpenSSL, BoringSSL, Python fastecdsa
  - Incorrect results and/or leakage

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✗)
- Hidden assumptions $\rightarrow$ unexpected behavior
- Issues in the past:
  - NSS, OpenSSL, BoringSSL, Python fastecdsa
  - Incorrect results and/or leakage
- Explicit-Formulas Database (EFD) [Bernstein, Lange] helps...

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✗)
- Hidden assumptions $\rightarrow$ unexpected behavior
- Issues in the past:
  - NSS, OpenSSL, BoringSSL, Python fastecdsa
  - Incorrect results and/or leakage
- Explicit-Formulas Database (EFD) [Bernstein, Lange] helps...but still not ideal

Choosing an addition formula

Problems:

- Many formulas with exceptional cases ($P + Q =$ ✘)
- Hidden assumptions $\rightarrow$ unexpected behavior
- Issues in the past:
  - NSS, OpenSSL, BoringSSL, Python fastecdsa
  - Incorrect results and/or leakage
- Explicit-Formulas Database (EFD) [Bernstein, Lange] helps...but still not ideal

Goal: classify all exceptional cases in EFD formulas

EFD addition formulas

Model	Coordinates	(x, y) representation	Number of formulas
short Weierstrass	projective	$(xZ : yZ : Z)$	21
	Jacobian	$(xZ^2 : yZ^3 : Z)$	36
	modified	$(xZ^2 : yZ^3 : Z : aZ^4)$	4
	w12 with $b = 0$	$(xZ : yZ^2 : Z)$	2
	xyzz	$(xZ^2 : yZ^3 : Z^2 : Z^3)$	6
	xz	$(xZ : Z)$	22
Montgomery	xz	$(xZ : Z)$	8
twisted Edwards	projective	$(xZ : yZ : Z)$	3
	extended	$(xZ : yZ : xyZ : Z)$	18
	inverted	$\left(\frac{Z}{x} : \frac{Z}{y} : Z\right)$	3
Edwards	projective	$(xZ : yZ : Z)$	12
	inverted	$\left(\frac{Z}{x} : \frac{Z}{y} : Z\right)$	6
	yz	$\left(yZ\sqrt{d} : Z\right)$	6
	yzsquared	$\left(y^2Z\sqrt{d} : Z\right)$	6

Classifying exceptional points

- Both fuzzing and manual analysis

Classifying exceptional points

- Both fuzzing and manual analysis
- All 111 addition and 42 differential addition EFD formulas analyzed

Classifying exceptional points

- Both fuzzing and manual analysis
- All 111 addition and 42 differential addition EFD formulas analyzed
- All exceptional points completely classified

Classifying exceptional points

- Both fuzzing and manual analysis
- All 111 addition and 42 differential addition EFD formulas analyzed
- All exceptional points completely classified
- New family of exceptional points found for `add-2007-bl`:

$$P = (X_1 : Y_1 : 1) \text{ and } Q = (X_2 : -Y_1 : 1) \text{ with } X_1 \neq X_2,$$

Formula example: `add-2007-bl` (exceptions)

Input: $P = (X_1 : Y_1 : Z_1), \quad Q = (X_2 : Y_2 : Z_2)$

Output: $P + Q = (X_3 : Y_3 : Z_3)$

$$U_1 = X_1 \cdot Z_2$$
$$U_2 = X_2 \cdot Z_1$$
$$S_1 = Y_1 \cdot Z_2$$
$$S_2 = Y_2 \cdot Z_1$$
$$ZZ = Z_1 \cdot Z_2$$
$$T = U_1 + U_2$$
$$TT = T^2$$
$$M = S_1 + S_2$$
$$t_0 = ZZ^2$$
$$t_1 = a \cdot t_0$$
$$t_2 = U_1 \cdot U_2$$

$$t_3 = TT - t_2$$
$$R = t_3 + t_1$$
$$F = ZZ \cdot M$$
$$L = M \cdot F$$
$$LL = L^2$$
$$t_4 = T + L$$
$$t_5 = t_4^2$$
$$t_6 = t_5 - TT$$
$$G = t_6 - LL$$
$$t_7 = R^2$$
$$t_8 = 2 \cdot t_7$$

$$W = t_8 - G$$
$$t_9 = F \cdot W$$
$$X_3 = 2 \cdot t_9$$
$$t_{10} = 2 \cdot W$$
$$t_{11} = G - t_{10}$$
$$t_{12} = 2 \cdot LL$$
$$t_{13} = R \cdot t_{11}$$
$$Y_3 = t_{13} - t_{12}$$
$$t_{14} = F^2$$
$$t_{15} = F \cdot t_{14}$$
$$Z_3 = 4 \cdot t_{15}$$

Formula example: `add-2007-bl` (exceptions)

Input: $P = (X_1 : Y_1 : Z_1), \quad Q = (X_2 : Y_2 : Z_2)$

Output: $P + Q = (X_3 : Y_3 : Z_3)$

$$U_1 = X_1 \cdot Z_2$$
$$U_2 = X_2 \cdot Z_1$$
$$S_1 = Y_1 \cdot Z_2$$
$$S_2 = Y_2 \cdot Z_1$$
$$ZZ = Z_1 \cdot Z_2$$
$$T = U_1 + U_2$$
$$TT = T^2$$
$$M = S_1 + S_2$$
$$t_0 = ZZ^2$$
$$t_1 = a \cdot t_0$$
$$t_2 = U_1 \cdot U_2$$

$$t_3 = TT - t_2$$
$$R = t_3 + t_1$$
$$F = ZZ \cdot M$$
$$L = M \cdot F$$
$$LL = L^2$$
$$t_4 = T + L$$
$$t_5 = t_4^2$$
$$t_6 = t_5 - TT$$
$$G = t_6 - LL$$
$$t_7 = R^2$$
$$t_8 = 2 \cdot t_7$$

$$W = t_8 - G$$
$$t_9 = F \cdot W$$
$$X_3 = 2 \cdot t_9$$
$$t_{10} = 2 \cdot W$$
$$t_{11} = G - t_{10}$$
$$t_{12} = 2 \cdot LL$$
$$t_{13} = R \cdot t_{11}$$
$$Y_3 = t_{13} - t_{12}$$
$$t_{14} = F^2$$
$$t_{15} = F \cdot t_{14}$$
$$Z_3 = 4 \cdot Z_2^3 \cdot Z_1^3 \cdot (Y_2 \cdot Z_1 + Y_1 \cdot Z_2)^3$$

Formula example: `add-2007-bl` (exceptions)

Input: $P = (X_1 : Y_1 : 1), \quad Q = (X_2 : -Y_1 : 1), \quad X_1 \neq X_2$

Output: $P + Q = (X_3 : Y_3 : Z_3)$

$$U_1 = X_1 \cdot Z_2$$
$$U_2 = X_2 \cdot Z_1$$
$$S_1 = Y_1 \cdot Z_2$$
$$S_2 = Y_2 \cdot Z_1$$
$$ZZ = Z_1 \cdot Z_2$$
$$T = U_1 + U_2$$
$$TT = T^2$$
$$M = S_1 + S_2$$
$$t_0 = ZZ^2$$
$$t_1 = a \cdot t_0$$
$$t_2 = U_1 \cdot U_2$$

$$t_3 = TT - t_2$$
$$R = t_3 + t_1$$
$$F = ZZ \cdot M$$
$$L = M \cdot F$$
$$LL = L^2$$
$$t_4 = T + L$$
$$t_5 = t_4^2$$
$$t_6 = t_5 - TT$$
$$G = t_6 - LL$$
$$t_7 = R^2$$
$$t_8 = 2 \cdot t_7$$

$$W = t_8 - G$$
$$t_9 = F \cdot W$$
$$X_3 = 2 \cdot t_9$$
$$t_{10} = 2 \cdot W$$
$$t_{11} = G - t_{10}$$
$$t_{12} = 2 \cdot LL$$
$$t_{13} = R \cdot t_{11}$$
$$Y_3 = t_{13} - t_{12}$$
$$t_{14} = F^2$$
$$t_{15} = F \cdot t_{14}$$
$$Z_3 = 4 \cdot (Y_1 + Y_2)^3$$

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]
- Other attacks:
  - Refined Power Analysis (RPA) [Goubin]

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]
- Other attacks:
  - Refined Power Analysis (RPA) [Goubin]
  - Zero-Value Point (ZVP) [Akiskita, Takagi]

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]
- Other attacks:
  - Refined Power Analysis (RPA) [Goubin]
  - Zero-Value Point (ZVP) [Akiskita, Takagi]

Goal: unify and generalize these

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]
- Other attacks:
  - Refined Power Analysis (RPA) [Goubin]
  - Zero-Value Point (ZVP) [Akiskita, Takagi]

Goal: unify and generalize these

- Common assumptions:

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]
- Other attacks:
  - Refined Power Analysis (RPA) [Goubin]
  - Zero-Value Point (ZVP) [Akiskita, Takagi]

Goal: unify and generalize these

- Common assumptions:
  - Scalar multiplication side-channel oracle

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]
- Other attacks:
  - Refined Power Analysis (RPA) [Goubin]
  - Zero-Value Point (ZVP) [Akiskita, Takagi]

Goal: unify and generalize these

- Common assumptions:
  - Scalar multiplication side-channel oracle
  - Static private key k (e.g., in ECDH, X25519)

Side-channel attacks on formulas

- Exceptional Procedure Attack (EPA) [Izu, Takagi]
- Other attacks:
  - Refined Power Analysis (RPA) [Goubin]
  - Zero-Value Point (ZVP) [Akiskita, Takagi]

Goal: unify and generalize these

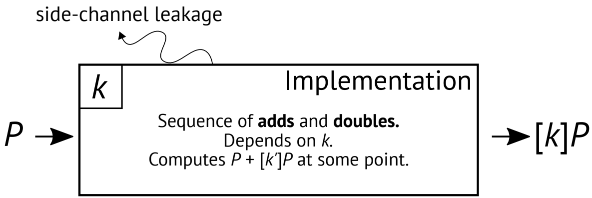
- Common assumptions:
  - Scalar multiplication side-channel oracle
  - Static private key k (e.g., in ECDH, X25519)
  - If k' is a binary prefix of k, then $[k']P$ appears during $[k]P$ computation

The unified scenario

Goal: recover k from ECC implementation

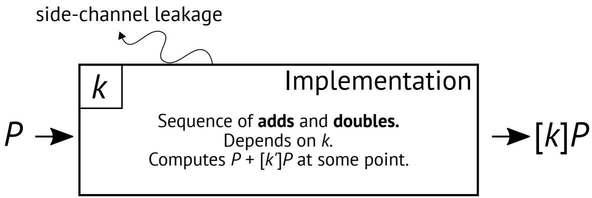
The unified scenario

Goal: recover *k* from ECC implementation



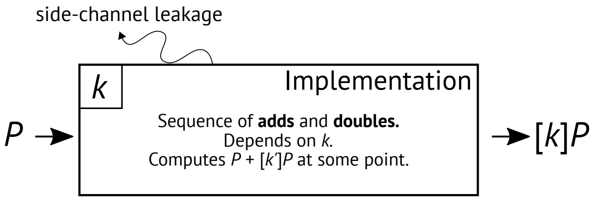
The unified scenario

Goal: recover *k* from ECC implementation



1. Guess k' - a prefix of k

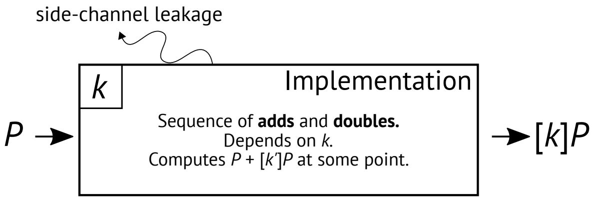
Goal: recover k from ECC implementation



1. Guess k' - a prefix of k
2. Construct a point P based on k' and given formula

The unified scenario

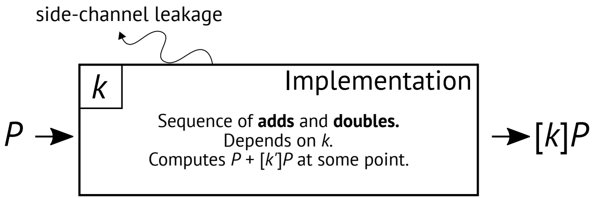
Goal: recover k from ECC implementation



1. Guess k' - a prefix of k
2. Construct a point P based on k' and given formula
3. Input P to the implementation

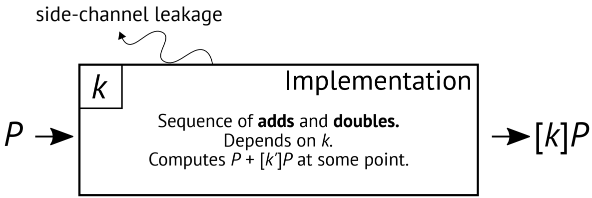
The unified scenario

Goal: recover k from ECC implementation



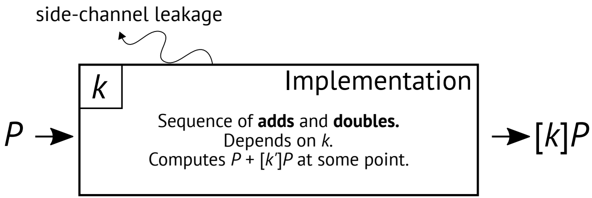
1. Guess k' - a prefix of k
2. Construct a point P based on k' and given formula
3. Input P to the implementation
4. Verify the guess k' using a side channel

Goal: recover k from ECC implementation



1. Guess k' - a prefix of k
2. Construct a point P based on k' and given formula
3. Input P to the implementation
4. Verify the guess k' using a side channel
5. Repeat to sequentially recover all bits of k

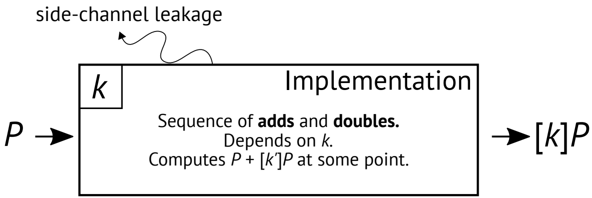
Goal: recover k from ECC implementation



1. Guess k' - a prefix of k
2. Construct a point P s.t. $P + [k']P$ fails
3. Input P to the implementation
4. Verify the guess k' using a side channel: error
5. Repeat to sequentially recover all bits of k

The unified scenario - ZVP

Goal: recover k from ECC implementation



1. Guess k' - a prefix of k
2. Construct a point P s.t. $P + [k']P$ forces an intermediate $f = 0$
3. Input P to the implementation
4. Verify the guess k' using a side channel: intermediate 0 detection
5. Repeat to sequentially recover all bits of k

Formula example: `add-2007-bl` (ZVP)

Input: $P = (X_1 : Y_1 : Z_1), \quad Q = (X_2 : Y_2 : Z_2)$

Output: $P + Q = (X_3 : Y_3 : Z_3)$

$U_1 = X_1 \cdot Z_2$
$U_2 = X_2 \cdot Z_1$
$S_1 = Y_1 \cdot Z_2$
$S_2 = Y_2 \cdot Z_1$
$ZZ = Z_1 \cdot Z_2$
$T = U_1 + U_2$
$TT = T^2$
$M = S_1 + S_2$
$t_0 = ZZ^2$
$t_1 = a \cdot t_0$
$t_2 = U_1 \cdot U_2$

$t_3 = TT - t_2$
$R = t_3 + t_1$
$F = ZZ \cdot M$
$L = M \cdot F$
$LL = L^2$
$t_4 = T + L$
$t_5 = t_4^2$
$t_6 = t_5 - TT$
$G = t_6 - LL$
$t_7 = R^2$
$t_8 = 2 \cdot t_7$

$W = t_8 - G$
$t_9 = F \cdot W$
$X_3 = 2 \cdot t_9$
$t_{10} = 2 \cdot W$
$t_{11} = G - t_{10}$
$t_{12} = 2 \cdot LL$
$t_{13} = R \cdot t_{11}$
$Y_3 = t_{13} - t_{12}$
$t_{14} = F^2$
$t_{15} = F \cdot t_{14}$
$Z_3 = 4 \cdot t_{15}$

Formula example: `add-2007-bl` (ZVP)

Input: $P = (X_1 : Y_1 : Z_1), \quad Q = (X_2 : Y_2 : Z_2)$

Output: $P + Q = (X_3 : Y_3 : Z_3)$

$U_1 = X_1 \cdot Z_2$
$U_2 = X_2 \cdot Z_1$
$S_1 = Y_1 \cdot Z_2$
$S_2 = Y_2 \cdot Z_1$
$ZZ = Z_1 \cdot Z_2$
$T = U_1 + U_2$
$TT = T^2$
$M = S_1 + S_2$
$t_0 = ZZ^2$
$t_1 = a \cdot t_0$
$t_2 = U_1 \cdot U_2$

$t_3 = TT - t_2$
$R = t_3 + t_1$
$F = ZZ \cdot M$
$L = M \cdot F$
$LL = L^2$
$t_4 = Y_2^2 \cdot Z_1^3 \cdot Z_2 + 2 \cdot Y_1 \cdot Y_2 \cdot Z_1^2 \cdot Z_2^2 + Y_1^2 \cdot Z_1 \cdot Z_2^3 + X_2 \cdot Z_1 + X_1 \cdot Z_2$
$t_5 = t_4^2$
$t_6 = t_5 - TT$
$G = t_6 - LL$
$t_7 = R^2$
$t_8 = 2 \cdot t_7$

$W = t_8 - G$
$t_9 = F \cdot W$
$X_3 = 2 \cdot t_9$
$t_{10} = 2 \cdot W$
$t_{11} = G - t_{10}$
$t_{12} = 2 \cdot LL$
$t_{13} = R \cdot t_{11}$
$Y_3 = t_{13} - t_{12}$
$t_{14} = F^2$
$t_{15} = F \cdot t_{14}$
$Z_3 = 4 \cdot t_{15}$

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Solving DCP $\rightarrow$ constructing oracle $\rightarrow$ private key recovery

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Solving DCP $\to$ constructing oracle $\to$ private key recovery
- For EPA, take $f = Z_3$

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Solving DCP $\rightarrow$ constructing oracle $\rightarrow$ private key recovery
- For EPA, take $f = Z_3$
- For ZVP, take f an intermediate expression

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Solving DCP $\rightarrow$ constructing oracle $\rightarrow$ private key recovery
- For EPA, take $f = Z_3$
- For ZVP, take f an intermediate expression
- For RPA, take $f = X_3$ or $f = Y_3$

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Hard in general: capturing $Q = [k']P$ does not scale - k' up to ≈ 20 bits

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Hard in general: capturing $Q = [k']P$ does not scale - k' up to ≈ 20 bits
- Solved new cases (e.g., $k' \equiv l/m \pmod{n}$ with $|l|, |m|$ small)

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Hard in general: capturing $Q = [k']P$ does not scale - k' up to ≈ 20 bits
- Solved new cases (e.g., $k' \equiv l/m \pmod{n}$ with $|l|, |m|$ small)
- Easy when f does not depend on Q:

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Hard in general: capturing $Q = [k']P$ does not scale - k' up to ≈ 20 bits
- Solved new cases (e.g., $k' \equiv l/m \pmod{n}$ with $|l|, |m|$ small)
- Easy when f does not depend on Q:
  - RPA

The DCP

The dependent coordinates problem (DCP)

Given $k' \in \mathbb{Z}$, an elliptic curve E over $\mathbb{F}_p$ and a polynomial f, find $P, Q \in E(\mathbb{F}_p)$ such that

$$Q = [k']P \quad \text{and} \quad f(P, Q) = 0.$$

- Hard in general: capturing $Q = [k']P$ does not scale - k' up to ≈ 20 bits
- Solved new cases (e.g., $k' \equiv l/m \pmod{n}$ with $|l|, |m|$ small)
- Easy when f does not depend on Q:
  - RPA
  - Some ZVP cases - new adaptations to window methods, simulated attack against `add-2016-rcb`

Contributions

- An open-source formula-unrolling tool - extension of **pyecsca** (ECC reversing toolkit)

Contributions

- An open-source formula-unrolling tool - extension of **pyecsca** (ECC reversing toolkit)
- Complete EFD addition formula classification, new exceptions found

Contributions

- An open-source formula-unrolling tool - extension of **pyecsca** (ECC reversing toolkit)
- Complete EFD addition formula classification, new exceptions found
- New attack unification framework + new ZVP attack adaptation

Contributions

- An open-source formula-unrolling tool - extension of **pyecsca** (ECC reversing toolkit)
- Complete EFD addition formula classification, new exceptions found
- New attack unification framework + new ZVP attack adaptation
- New DCP cases solved, but remains open

Lessons learned

- Study popular systems in advance, not reactively

Lessons learned

- Study popular systems in advance, not reactively
- Simpler viewpoints help cryptanalysis

Lessons learned

- Study popular systems in advance, not reactively
- Simpler viewpoints help cryptanalysis
- The chosen formula matters

Lessons learned

- Study popular systems in advance, not reactively
- Simpler viewpoints help cryptanalysis
- The chosen formula matters
- We need to be careful about too strong claims

Lessons learned

- Study popular systems in advance, not reactively
- Simpler viewpoints help cryptanalysis
- The chosen formula matters
- We need to be careful about too strong claims
- Be explicit about assumptions, document them!

Additional materials

Thanks for your attention!

Tooling, analysis, demos and more: crocs.fi.muni.cz/public/papers/formulas_asiacrypt21

References

- Daniel J. Bernstein, Tanja Lange; **Explicit-Formulas Database**, 2007. www.hyperelliptic.org/EFD/
- Tetsuya Izu, Tsuyoshi Takagi; **Exceptional Procedure Attack on Elliptic Curve Cryptosystems.** Public Key Cryptography 2003: 224-239
- Louis Goubin; **A Refined Power-Analysis Attack on Elliptic Curve Cryptosystems.** Public Key Cryptography 2003: 199-210
- Toru Akishita, Tsuyoshi Takagi; **Zero-Value Point Attacks on Elliptic Curve Cryptosystem.** ISC 2003: 218-233
- Jan Jancar; **pyecsca**, 2018. neuromancer.sk/pyecsca

Icons and images from **Font Awesome**, **Canva** & **Pixabay**