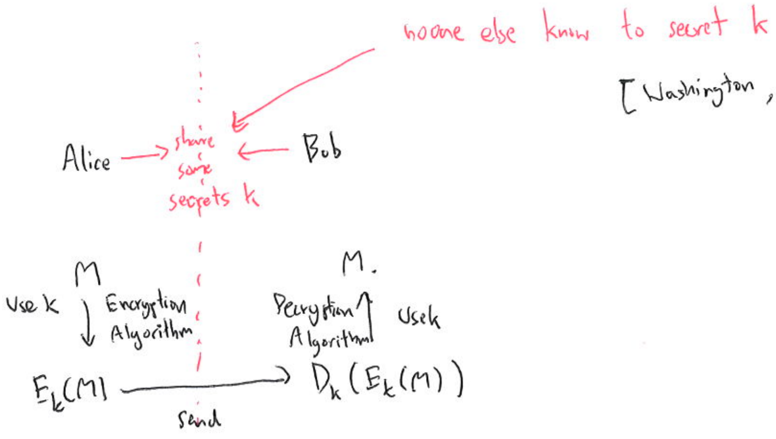
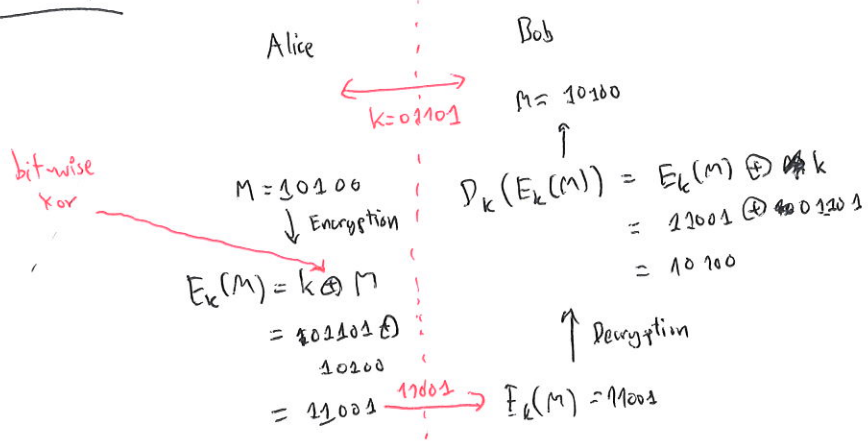
Private Key Cryptography

[Washington, Chapter 6]



One-time pad

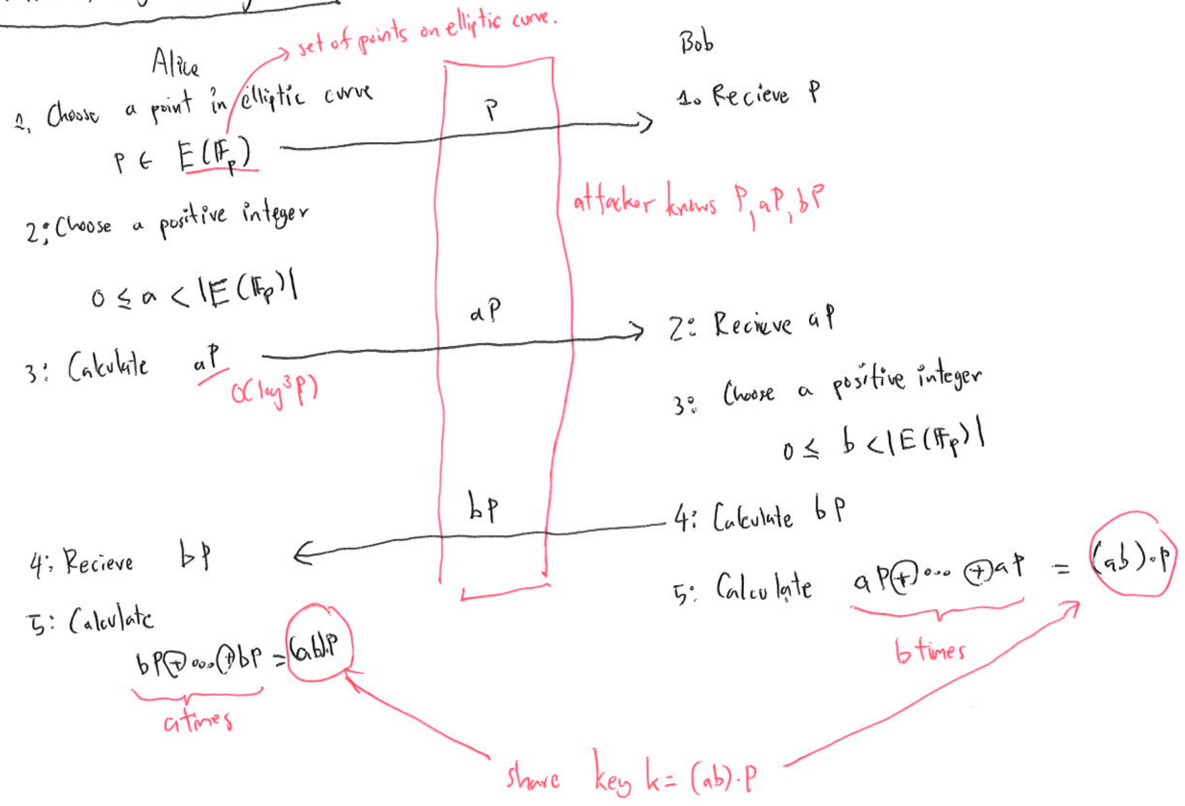


- One-Time Pad is known to be costly and weak.
- Currently, Advanced Encryption Scheme (AES) is the most commonly used cryptosystem.

<u>Problem</u> How can Alice and Bob share the key k to each other? → Key Exchange Protocol.

Diffie-Hellman Key Exchange Protocol

[~~Washington~~, Chapter



Alice

1, Choose a point in elliptic curve $P \in E(\mathbb{F}_p)$

2: Choose a positive integer $0 \leq a < |E(\mathbb{F}_p)|$

3: Calculate aP

4: Recieve bP

5: Calculate $\underbrace{bP \oplus \cdots \oplus bP}_{a\ times} = \boxed{(ab)P}$

Bob

1. Recieve P

2: Recieve aP

3: Choose a positive integer $0 \leq b < |E(\mathbb{F}_p)|$

4: Calculate bP

5: Calculate $\underbrace{aP \oplus \cdots \oplus aP}_{b\ times} = \boxed{(ab)\cdot P}$

Diffie-Hellman Problem

Input: P, aP, bP

Calculate: $(ab)P$

$\xleftrightarrow{\text{equivalent}}$

Discrete Logarithm Problem

Input: P, aP

Calculate: a

Large when $p \approx 2^{256}$

$\downarrow$

very hard problem

not possible to have algorithm faster than $\boxed{O(\sqrt{p})}$

Algorithms for Discrete Logarithm Problem: Baby step, Giant step

Input: P, Q

Find: a such that $a.P = Q$

$N = \sqrt{|E(\mathbb{F}_p)|}$ (maximum value for a)

1: Calculate $S = \{P, 2P, 3P, \dots, N \cdot P\}$ $\rightarrow O(N)$ (baby step)

2: Check if any in S is equal to Q. If yes, we have a.

" ——————— " $Q \oplus \neg(N P)$. ~~If~~ $\Rightarrow \underline{b}P = Q \oplus (\neg NP)$

↳ the entry that we hit.

$$bP \oplus NP = Q \oplus \cancel{(\neg NP) \oplus (NP)}$$

$$Q = \underbrace{(b+N)}_{a} \cdot P$$

Check if any in S is equal to $Q \oplus \neg(2N \cdot P)$. $\Rightarrow bP = Q \oplus (\neg 2N \cdot P)$

$$bP \oplus 2N \cdot P = Q \oplus \cancel{(\neg 2N \cdot P)} \oplus \cancel{(2N \cdot P)}$$

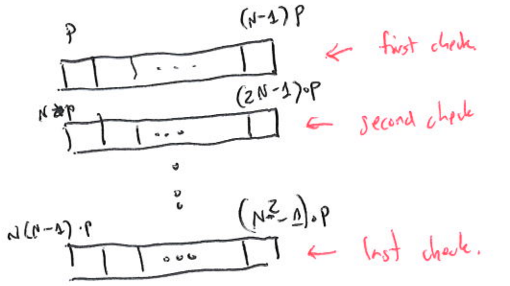
$$\underbrace{(b+2N)}_{a} \cdot P = Q$$

⋮

Check if any in S is equal to $Q \oplus \neg((N-1)N \cdot P)$ $\Rightarrow Q = ((N-1)N + b) \cdot P$.

Any of these cases must be satisfied.

$O(N)$



← first check

← second check

← last check.

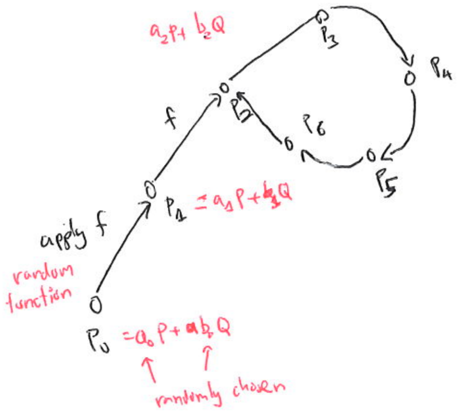
Time Complexity $O(N) = O(\sqrt{|E(\mathbb{F}_p)|})$ ← may be doable when $p \approx 2^{128}$

Memory Complexity $O(N) = O(\sqrt{|E(\mathbb{F}_p)|})$ ← not doable even when $p \approx 2^{64}$

[Washington, Chapter 5]

Pollard's ρ Method

Idea



$$P_2 = P_7$$

$$a_2 P \oplus b_2 Q = a_7 P \oplus b_7 Q$$

$$a_2 P \oplus \cancel{b_2 Q} \oplus \cancel{\neg b_2 Q} \oplus \neg a_7 P$$

$$= \cancel{a_7 P} \oplus b_7 Q \oplus \neg b_2 Q \oplus \cancel{\neg a_7 P}$$

$$(a_2 - a_7) \cdot P = (b_7 - b_2)Q$$

We can solve this equation to obtain $P = a \cdot Q$ (explained later)

- We will be back to the visited point after $O(\sqrt{|E(\mathbb{F}_p)|})$ applications of f.
- But, how to check if a particular point → store all the points will consume $O(\sqrt{|E(\mathbb{F}_p)|})$ ☹

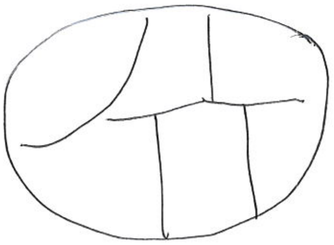
Idea Use fast and slow kangarous

fast: move 2 hops, apply f for 2 times

slow: move 1 hop, apply f for 2 times

What is function f ?

- To garantee that the visited point is found in $O(\sqrt{|E(\mathbb{F}_p)|})$, f has to be random.
- It is hard to have the random function, as we need to have the same output for the same input. (then, we have to store $f(P)$ for many point P.)



Divide $E(\mathbb{F}_p)$ into 20 pieces $S_1, \cdots, S_{20}$

For $P_i \in S_j$, $f(P_i) = P_i \oplus a_j' P \oplus b_j' Q$

random number

Algorithm

random numbers; slow kangaroo; fast kangaroo

```
1: $P_0 \leftarrow a_0 P + b_0 Q$, $R \leftarrow P_0$ and $S \leftarrow P_0$
2: Do
       $R \leftarrow f(R)$, $Q \leftarrow f(f(R))$
   Until $R = S$
3: Suppose that $R = a_i P \oplus b_i Q$
                $S = a_{2i} P \oplus b_{2i} Q$
       Solve $a_i P \oplus b_i Q = a_{2i} P \oplus b_{2i} Q$ to have
           $Q = \bigcirc P$
```

Example $p = 1093$ $A = B = 1$ $E(\mathbb{F}_{1093}) = \{(x,y) \in \mathbb{F}_{1093}^2 : y \otimes y = x \otimes x \otimes x \oplus x \oplus 1\}$

$|E(\mathbb{F}_{1093})| = 1067$

Input: $P = (0,1)$, $Q = (413, 957)$.

[We want to find a such that $(413, 957) = a(0,1)$]

$$f((x,y)) = \begin{cases} (x,y) \oplus 4P \oplus 3Q & \text{when } x \equiv 0 \bmod 3 \\ (x,y) \oplus 9P \oplus 17Q & \text{when } x \equiv 1 \bmod 3 \\ (x,y) \oplus 19P \oplus 6Q & \text{when } x \equiv 2 \bmod 3 \end{cases}$$

$1: \; P_0 = 3P \oplus 5Q = (326, 69) = R = S$

→ random number

Iteration 1

$R = f(R) = (326, 69) \oplus 19(0,1) \oplus 6(413, 957) = (727, 589)$

$S = f(f(S)) = f((727, 589)) = (727, 589) \oplus 9(0,1) \oplus 17(413, 957)$

$= (560, 365)$

$R = (3P + 5Q) \oplus 19P \oplus 6Q = 22P \oplus 11Q$

$S = (22P \oplus 11Q) \oplus 9P \oplus 17Q = 31P \oplus 28Q$

Iteration 2

$R = (727, 589) = 31P \oplus 28Q$

$S = (473, 903) = 69P \oplus 40Q$

$\vdots$

Iteration 53

$R = (71, 338) = 620P \oplus 557Q$ $S = (71, 338) = 1217P \oplus 1131Q$

$620P \oplus 557Q = 1217P \oplus 1131Q$

[$|E(\mathbb{F}_{1093})| = 1067$
$1067P = e$ for any P.]

$620P \oplus 557Q = \underset{e}{\cancel{1067}P} \oplus 150P \oplus \underset{e}{\cancel{1067}Q} \oplus 64Q$

$620P \oplus 557Q = 150P \oplus 64Q$

$620P \oplus \underbrace{557Q \oplus 510Q}_{1067Q = e} \oplus -150P = \cancel{150P} \oplus 6\overset{4}{\cancel{7}}Q \oplus 510Q \oplus -\cancel{150}P$

$k \cdot 470P = \cancel{574Q}\, k \cdot 574Q$

$k \cdot 470P = [1067n + 1]Q$

Bonus Question

Solve Diophantine's equation $1067n + 1 = 574k$ when k, n are integers,

Ans ~~k=2764, 9~~ ~~2~~353 $k = 303$ $n = 163$

$\cancel{767 \cdot 470}$ $(303 \cdot 470)P = [1067 \cdot 163 + 1]Q$

$142410\,P = Q$

$[133 \cdot 1067 + 499]P = Q$

$499P = Q$

$a = 499.$

□

Complexity

	Baby step, Giant Step	Pollard's ρ method
Memory	$O(\sqrt{p})$	$O(\log p)$ $\therefore O(1)$ elliptic points, each point has $O(\log p)$
Computation Time	$O(\sqrt{p})$	$O(\sqrt{p})$ with high probability
More precise computation time	$c_2\sqrt{p} + o(\sqrt{p})$ small c_2	$c_1 \sqrt{p} + o(\sqrt{p})$ very large c_1

[Bernstein and Lange, ANT'13]

ongoing : How to reduce c_2 and c_1 ?