Multiplicação Escalar

Ricardo Dahab & Julio López

Laboratório de Criptografia Aplicada
IC-UNICAMP
Campinas, 21 de fevereiro 2004

Roteiro

- Definição de multiplicação escalar kP
- Métodos para calcular kP:
  - Quando P é um ponto elíptico geral
    (Binário, Janela, López-Dahab)
  - Quando P é um ponto elíptico conhecido (fixo)
    (Janela, Comb)
- Métodos para calcular $kP + sQ$
  (Shamir, Möller)

Introdução

- Dados um inteiro k e um ponto P sobre a curva $E_{a,b}$ definida sobre o corpo finito $\mathbb{F}_q$, a operação

$$kP = \underbrace{P + P + \ldots + P}_{k \text{ vezes}}$$

  é chamada *multiplicação escalar* ou *multiplicação de pontos.*

- A operação kP ou $[k]P$ domina o tempo de execução dos esquemas criptográficos baseados em curvas elípticas.

- Assumimos que $\#E_{a,b}(\mathbb{F}_q) = n \cdot h$, onde n é primo e h é pequeno ($n \approx q$), o ponto P tem ordem n ($nP = \mathcal{O}$), e o multiplicador k é um número aleatório no intervalo $[1, n-1]$, cuja representação binária é $(k_{t-1}k_{t-2}\ldots k_1 k_0)_2$, onde $t \approx \lceil \log_2 q \rceil$.

Classificação dos métodos para calcular kP

- kP: P um ponto elíptico conhecido (ou fixo)
  (P gerador de um subgrupo de ordem n) (off-line)

- kP: P um ponto elíptico arbitrário (on-line)

- $kP + sQ$: P e Q pontos elípticos
  (geralmente P fixo e Q arbitrário)

Técnicas básicas para calcular kP P ponto arbitrário

- $kP = \sum_{i=0}^{t-1} k_i 2^i P,\ k_i \in \{0,1\},\quad \{P\}$
- $kP = \sum_{i=0}^{t} k_i 2^i P,\ k_i \in \{-1,0,1\},\quad \{P\}$
- $kP = \sum_{i=0}^{t_k} k_i 2^i P,\ k_i \in \{0, \pm 1, \pm 3, \ldots, 2^{w-1}-1\}$
  precomputação: $\{P, 3P, 5P, \ldots, (2^{w-1}-1)P\}$

Técnicas básicas para calcular kP
P ponto fixo

- $kP = \sum_{i=0}^{t-1} k_i Q_i,\ k_i \in \{0,1\},\ Q_i = 2^i P$

- $kP = \sum_{i=1}^{2^w-1} iQ_i,\ \ Q_i = \sum_{j:K_j=i} 2^{wj} P$
  $k = (K_{d-1}, \ldots, K_1, K_2)_{2^w},\ d = \lceil t/w \rceil$

- $kP = \sum_{i=0}^{e-1} 2^i \sum_{j=0}^{v-1} (2^{ej} \sum_{l=0}^{w-1} [K_j^l]_i\ 2^{dl} P)$
  onde $[K_j^l] = (k_{s+e-1}, \ldots, k_{s+1}, k_s),\ s = dl + ej,\ e = \lceil d/v \rceil$.

Método Binário
exemplo 1: esquerda-direita:

$$k = (\ldots, k_4, k_3, k_2, k_1, k_0)_2$$

$$\begin{aligned} kP &= k_4 2^4 P + k_3 2^3 P + k_2 2^2 P + k_1 2P + k_0 P \\ &= 2[2[2[2k_4 P + k_3 P] + k_2 P] + k_1 P] + k_0 P \end{aligned}$$

$$\begin{aligned} Q &\leftarrow k_4 P \\ Q &\leftarrow 2Q + k_3 P \\ Q &\leftarrow 2Q + k_2 P \\ Q &\leftarrow 2Q + k_1 P \\ kP = Q &\leftarrow 2Q + k_0 P \end{aligned}$$

Método Binário
exemplo 2: direita-esquerda:

$$k = (\ldots, k_4, k_3, k_2, k_1, k_0)_2$$

$$kP = k_0P + k_1[2P] + k_2[2^2P] + k_3[2^3P] + k_4[2^4P]$$

$$\begin{aligned}
Q &\leftarrow k_0P, & P &\leftarrow 2P \\
Q &\leftarrow Q + k_1P, & P &\leftarrow 2P \\
Q &\leftarrow Q + k_2P, & P &\leftarrow 2P \\
Q &\leftarrow Q + k_3P, & P &\leftarrow 2P \\
kP = Q &\leftarrow Q + k_4P
\end{aligned}$$

Método Binário (R-to-L)
algoritmo 3.26

Entrada: $k = (k_{t-1}, \ldots, k_1, k_0)_2,\ P \in E_{a,b}(\mathbb{F}_q)$.

Saída: kP

1. $Q \leftarrow \mathcal{O}$
2. **for** i **from** 0 **to** $t-1$ **do**
   - 2.1 **if** $k_i = 1$ **then** $Q \leftarrow Q + P$
   - 2.2 $P \leftarrow 2P$
3. **return** (Q)

Método Binário (L-to-R)
algoritmo 3.27

Entrada: $k = (k_{t-1}, \ldots, k_1, k_0)_2$, $P \in E_{a,b}(\mathbb{F}_q)$.

Saída: kP

1. $Q \leftarrow \mathcal{O}$
2. **for** i **from** $t-1$ **to** 0 **do**
   - 2.1 $Q \leftarrow 2Q$
   - 2.2 **if** $k_i = 1$ **then** $Q \leftarrow Q + P$
3. **return** (Q)

Análise do Método Binário

- O algoritmo binário requer, em média, aproximadamente m duplicações de pontos e $m/2$ somas de pontos, onde $m = \lceil \log_2 q \rceil$ e $|k| \approx m$.

$$T(m) = \frac{m}{2}S + mD$$

- Em termos de coordenadas (afins ou projetivas) temos:

Coordenadas	$\mathbb{F}_{2^m}$	$\mathbb{F}_p$
Afins	$3mM$ $+1.5mI$	$2.5mQ + 3mM$ $+1.5mI$
Projetivas	$8.5mM$ $+(2M + 1I)$	$8mM + 5.5mQ$ $+(1I + 3M + 1Q)$

Forma Não Adjacente de um Inteiro

Definição: A *forma não adjacente* (FNA) de um inteiro positivo k é uma expressão da forma $k = \sum_{i=0}^{l-1}$ onde $k_i \in \{-1, 0, 1\}, k_{l-1} \neq 0$, e nenhum par de dígitos consequtivos k_i são não-zero. O comprimento da FNA é l.

$$\begin{aligned} k &= (1122334455)_{10} \\ &= (10000010\ 11100101\ 01110110\ 11110111)_2 \\ &= (1000010\bar{1}\ 00\bar{1}010\bar{1}0\ \bar{1}000\bar{1}00\bar{1}\ 0000\bar{1}00\bar{1}) \end{aligned}$$

Forma Não Adjacente de um Inteiro/2

Motivação: A operação $-P(x,y)$ pode-se calcular eficientemente em curvas elípticas:

- Se $P=(x,y) \in E(\mathbb{F}_{2^m})$ então $P=(x,x+y)$.
- Se $P=(x,y) \in E(\mathbb{F}_p)$ então $-P=(x,-y)$.

$$kP = \sum_{i=0}^{l-1} 2^i k_i P$$

$$k_i P = \begin{cases} P & \text{se } k_i = 1 \\ \mathcal{O} & \text{se } k_i = 0 \\ -P & \text{se } k_i = \bar{1} \end{cases}$$

Propriedades das FNAs

Teorema: Seja k um inteiro positivo, então temos:

1. O inteiro k tem uma única FNA denotada FNA(k).
2. A FNA(k) tem o mínimo número de dígitos não-zero de qualquer representação com sinal de k.
3. O comprimento da FNA(k) é pelo menos um dígito mais do que o comprimento da representação binária de k.
4. Se o comprimento da FNA(k) é l, então $2^l/3 < k < 2^{l+1}/3$.
5. A densidade média dos dígitos não-zero entre todas as FNA de comprimento l é aproximadamente $1/3$.

Algoritmo para calcular a FNA(k) algoritmo 3.30

Entrada: Um inteiro positivo k

Saída: FNA(k)

1. $i \leftarrow 0$
2. **while** $k \geq 1$ **do**
   - 2.1 **if** k é ímpar **then** $k_i \leftarrow 2 - (k \bmod 4)$, $k \leftarrow k - k_i$
   - 2.2 **else** $k_i \leftarrow 0$
   - 2.3 $k \leftarrow k/2, i \leftarrow i + 1$
3. **return** $(k_{i-1}, k_{i-2}, \ldots, k_1, k_0)$.

Método Binário (FNA)
algoritmo 3.31

Entrada: Um inteiro positivo k e um ponto $P \in E(\mathbb{F}_q)$.

Saída: kP

1. Use o algoritmo 3.30 para calcular a FNA(k)
2. $Q \leftarrow \mathcal{O}$
3. **for** i **from** $l-1$ **to** 0 **do**

    3.1 $Q \leftarrow 2Q$

    3.2 **if** $k_i = 1$ **then** $Q \leftarrow Q + P$

    3.3 **if** $k_i = \bar{1}$ **then** $Q \leftarrow Q - P$

3. **return** (Q).

Análise do Método Binário na FNA

- O método na FNA requer, em média, aproximadamente m duplicações de pontos e $m/3$ somas de pontos, onde $m = \lceil \log_2 q \rceil$ e $|k| \approx m$.

$$T(m) = \frac{m}{3}S + mD$$

- Em termos de coordenadas (afins ou projetivas) temos:

Coordenadas	$\mathbb{F}_{2^m}$	$\mathbb{F}_p$
Afins	$2.66mM$ $+1.33mI$	$2.33mQ + 2.66mM$ $+1.33mI$
Projetivas	$7mM$ $+(2M + 1I)$	$6.66mM + 5mQ$ $+(1I + 3M + 1Q)$

Método da Janela

Definição: Seja $w \geq 2$ um inteiro positivo. A FNA de largura w de um inteiro k é uma expressão da forma $k = \sum_{i=0}^{l-1} k_i 2^i$ onde cada coeficiente não-zero k_i é ímpar, $|k_i| < 2^{w-1}$, $k_{l-1} \neq 0$, e pelo menos um dos w dígitos consequtivos é não-zero. O comprimento da FNA de largura w é l.

Método da Janela
Exemplo:

$$w = 2, 3, 4, 5, 6 :$$

$$\begin{aligned}
k &= (1122334455)_{10} \\
(k)_2 &= 1000010\ 11100101\ 01110110\ 11110111 \\
NAF_2(k) &= 1000101\bar{1}\ 00\bar{1}010\bar{1}0\ \bar{1}000\bar{1}00\bar{1}\ 0000\bar{1}00\bar{1} \\
NAF_3(k) &= 1000003\ 00\bar{1}00100\ 3000\bar{1}00\bar{1}\ 0000\bar{1}00\bar{1} \\
NAF_4(k) &= 1000010\ 00700005\ 00070007\ 000\bar{1}0007 \\
NAF_5(k) &= 10000\bar{1}500\ 00\bar{9}00000\ 110000009\ 0000000\bar{9} \\
NAF_6(k) &= 10000000\ 002\bar{3}00000\ 110000009\ 0000000\bar{9}
\end{aligned}$$

Propriedades das FNAs de largura w

Teorema: Seja k um inteiro positivo, então temos:

1. O inteiro k tem uma única FNA de largura$w$ denotada $\mathrm{FNA}_w(k)$.
2. $\mathrm{FNA}_2(k) = \mathrm{FNA}(k)$.
3. O comprimento da $\mathrm{FNA}_w(k)$ é pelo menos um dígito mais do que o comprimento da representação binária de k.
4. A densidade média dos dígitos não-zero entre todas as FNA de largura w de comprimento l é aproximadamente $1/(w+1)$.

Algoritmo para calcular a FNA(k) de largura w algoritmo 3.35

Entrada: Um inteiro positivo k e a largura da janela w

Saída: $\text{FNA}_w(k)$

1. $i \leftarrow 0$
2. **while** $k \geq 1$ **do**
   - 2.1 **if** k é ímpar **then** $k_i \leftarrow 2 - (k \bmod 2^w)$, $k \leftarrow k - k_i$
   - 2.2 **else** $k_i \leftarrow 0$
   - 2.3 $k \leftarrow k/2, i \leftarrow i + 1$
3. **return** $(k_{i-1}, k_{i-2}, \ldots, k_1, k_0)$.

A operação $k \bmod 2^w$ define-se como o inteiro u tal que $u \equiv k \pmod{2^w}$ e $-2^{w-1} \leq u < 2^{w-1}$.

Método da Janela ($\mathbf{FNA}_w(k)$) algoritmo 3.36

Entrada: Um inteiro positivo k, a largura w e um ponto $P \in E(\mathbb{F}_q)$.

Saída: kP.

1. Use o algoritmo 3.35 para calcular a $\mathrm{FNA}_w(k)$
2. Calcule $P_i = iP$ para $i \in \{1, 3, 5, \ldots, 2^{w-1} - 1\}$
3. $Q \leftarrow \mathcal{O}$
4. **for** i **from** $l-1$ **to** 0 **do**

   4.1 $Q \leftarrow 2Q$

   4.2 **if** $k_i \neq 0$ **then if** $k_i > 0$ **then** $Q \leftarrow Q + P_{k_i}$

   else $Q \leftarrow Q - P_{k_i}$
5. **return** (Q).

Método da Janela deslizante (Alg. 3.38)

Entrada: Um inteiro $k > 0$, a largura w e um ponto $P \in E(\mathbb{F}_q)$.

Saída: kP.

1. Use o algoritmo 3.30 para calcular a FNA(k).
2. Calcule $P_i = iP$ para $i \in \{1, 3, \ldots, 2(2^w - (-1)^w)/3\}$.
3. $Q \leftarrow \mathcal{O}$, $i \leftarrow l - 1$.
4. **while** $i \geq 0$ **do**
   - 4.1 **if** $k_i = 0$ **then** $t \leftarrow 1, u \leftarrow 0$;
   - 4.2 **else** ache o maior $t \leq w$ tal que $u \leftarrow (k_i, \ldots, k_{i-t+1})$ é ímpar.
   - 4.3 $Q \leftarrow 2^t Q$.
   - 4.4 **if** $u > 0$ **then** $Q \leftarrow Q + P_u$; **else if** $u < 0$ **then** $Q \leftarrow Q - P_{-u}$.
   - 4.5 $i \leftarrow i - t$.
5. **return** (Q).

Análise do Método da Janela

- O tempo de execução esperado do método da janela com largura w é aproximadamente:

$$[1D + (2^{w-2} - 1)S] + [\frac{m}{w+1}S + mD]$$

- O tempo de execução esperado do método da janela deslizante com largura w é aproximadamente:

$$[1D + (\frac{2^w - (-1)^w}{3} - 1)S + [\frac{m}{w + v(w)}S + mD],$$

onde $v(w) = \frac{4}{3} - \frac{(-1)^w}{3 \cdot 2^{w-2}}$.

- Qual método é melhor?

O Algoritmo López-Dahab (CHES 1999)

- Método para curvas elíptica definidas sobre $GF(2^m)$
- Método baseado no método de Montgomery (1987)
- Ideia principal:

$$kP = (101011)P = 43P$$

$$\begin{array}{ccc} 1 & P & 2P \\ 0 & 2P & 3P \\ 1 & 5P & 6P \\ 0 & 10P & 11P \\ 1 & 21P & 22P \\ 1 & 43P & 44P \end{array}$$

- Computações em termos das coordenadas x de lP e $(l+1)P$.

O Algoritmo López-Dahab (CHES 1999)
Algoritmo 3.40

Entrada: $(k = (k_{t-1}, \ldots, k_1, k_0)_2, k_{t-1} = 1, P = (x, y) \in \mathbb{F}_{2^m}$.

Saída: kP.

1. $X_1 \leftarrow x, Z_1 \leftarrow 1, X_2 \leftarrow x^4 + b, Z_2 \leftarrow x^2$.
2. **for** i **from** $t-2$ **downto** 0 **do**

    2.1 **if** k_i **then**

$$T \leftarrow Z_1,$$
$$Z_1 \leftarrow (X_1 Z_2 + X_2 Z_1)^2,$$
$$X_1 \leftarrow x Z_1 + X_1 X_2 T Z_2,$$
$$T \leftarrow X_2,$$
$$X_2 \leftarrow X_2^4 + b Z_2^4,$$
$$Z_2 \leftarrow T^2 Z_2^2.$$

O Algoritmo López-Dahab (CHES 1999)
Algoritmo 3.40/2

2.1 **else**

$T \leftarrow Z_2$,

$Z_2 \leftarrow (X_1 Z_2 + X_2 Z_1)^2$,

$X_2 \leftarrow x Z_2 + X_1 X_2 T Z_1$,

$T \leftarrow X_1$,

$X_1 \leftarrow X_1^4 + b Z_1^4$,

$Z_1 \leftarrow T^2 Z_1^2$.

3. $x_3 \leftarrow X_1/Z_1$.
4. $y_3 \leftarrow (x + X_1/Z_1)[(X_1 + xZ_1)(X_2 + xZ_2) + (x^2 + y)(Z_1 Z_2)]$
   $\cdot (xZ_1 Z_2)^{-1} + y$.
5. **return**((x_3, y_3)).

Análise do Algoritmo 3.40

- O algoritmo utiliza a representação binária de k.
- O algoritmo não utiliza memória adicional.
- Em cada iteração executa-se a mesma operação (uma soma e uma duplicação).
- O tempo de execução é: $6mM + (1I + 10M)$.
- Método preferido nas implementações hardware (exemplo: Sun)
- Certicom tem uma patente.

Geração e Validação de Parâmetros

Ricardo Dahab & Julio López

Laboratório de Criptografia Aplicada
IC-UNICAMP
Campinas, 7 de março de 2005

Roteiro

1. Parâmetros dos CCE
2. Geração de curvas verificáveis aleatóriamente
3. Verificação de que uma curva foi gerada aleatóriamente
4. Geração de parâmetros
5. Validação de parâmetros
6. Os parâmetros NIST (1999)
7. Os parâmetros SEC v. 1.0 (2000) (Revisão 2005, v. 2.0)

Parâmetros dos CCE (Domain Parameters)

$$D = (q, \mathrm{FR}, S, a, b, P, n, h)$$

- q: a ordem do corpo finito ($q = 2^m,\ q = p,\ q = p^m$).
- FR: a representação dos elmentos do corpo finito $\mathbb{F}_q$.
- S: semente para gerar os coeficientes elípticos a, b.
- a, b: coeficientes elípticos no corpo finito $\mathbb{F}_q$
  $y^2 = x^3 + ax + b, \quad y^2 + xy = x^3 + ax^2 + b.$
- $P = (x_p, y_p)$: ponto elíptico em $E(\mathbb{F}_q)$.
- n: a ordem do ponto P (isto é, $nP = \mathcal{O}$).
- h: o cofator, $h = \#E(\mathbb{F}_q)/n$.

Restrições nos Parâmetros n e L

- $n > 2^L$ e $L \geq 160$
- $L \leq \lfloor \log_2 q \rfloor$
- $n > 4\sqrt{q}$

Geração de Curvas Elípticas sobre $\mathbb{F}_{2^m}$

Entrada: Um inteiro m e uma função hash H de l-bits.

Saída: Uma semente S e coeficientes $a, b \in \mathbb{F}_{2^m}$ de uma curva elíptica binária $E_{a,b}: \ y^2 + xy = x^3 + ax^2 + b$.

1. $s = \lfloor (m - 1/l \rfloor, \ v = m - sl$.
2. Escolha uma cadeia binária S de g bits, com $g \leq l$.
3. Calcule $h = H(S)$, e seja b_0 a cadeia binária de comprimento v bits obtida como os v bits mais à direita de h.
4. Seja z o inteiro cuja representação binária é S.

Geração de Curvas Elípticas sobre $\mathbb{F}_{2^m}$ /2

5. **for** i **from** 1 **to** s **do**

   5.1 Seja s_i a cadeia binária de g bits que representa o inteiro $(z + i)$ mod 2^g.

   5.2 Calcule $b_i = H(s_i)$.

6. $b = b_0||b_1||...||b_s$.

7. **if** $b = 0$ **then** volte ao passo 2.

8. Escolha $a \in \mathbb{F}_{2^m}$ $(a \in \{0, 1\})$.

9. **return** (S, a, b)

Verificação de que uma Curva Elíptica sobre $\mathbb{F}_{2^m}$ foi Gerada Aleatóriamente

Entrada: Um inteiro m, uma função hash H de l bits, uma semente S de g bits ($g \geq l$), e os coeficientes $a, b \in \mathbb{F}_{2^m}$ de uma curva elíptica binária $E_{a,b}$.

Saída: Aceitação ou rejeição de que a curva $E_{a,b}$ foi gerada utilizando o algoritmo 4.19.

1. $s = \lfloor (m-1/l \rfloor,\ v = m - sl$.
2. Calcule $h = H(S)$, e seja b_0 a cadeia binária de comprimento v bits obtida como os v bits mais à direita de h.
3. Seja z o inteiro cuja representação binária é S.

Verificação de que uma Curva Elíptica sobre $\mathbb{F}_{2^m}$ foi Gerada Aleatóriamente /2

4. **for** i **from** 1 **to** s **do**

    4.1 Seja s_i a cadeia binária de g bits que representa o inteiro $(z + i) \bmod 2^g$.

    4.2 Calcule $b_i = H(s_i)$.

5. $b' = b_0||b_1||...||b_s$.

6. **if** $b' = b$ **then return** ("aceite"); **else return** ("rejeite").

Geração de Curvas Elípticas sobre $\mathbb{F}_p$

Entrada: Um primo $p > 3$ e uma função hash H de l bits.

Saída: Uma semente S, e coeficientes $a, b \in \mathbb{F}_p$ de uma curva elíptica $E_{a,b}: \ y^2 = x^3 + ax + b$.

1. $t = \lceil (m-1)/l \rceil, \ s = \lfloor (t-1)/l \rfloor, \ v = t - sl.$
2. Escolha uma cadeia binária S de g bits, com $g \leq l$.
3. Calcule $h = H(S)$, e seja r_0 a cadeia binária de comprimento v bits obtida como os v bits mais à direita de h.
4. Seja R_0 a cadeia binária obtida de r_0 onde o bit mais à esquerda de r_0 é zero ($R_0 = (0, [r_0]_{v-1}, [r_0]_{v-2}, ..., [r_0]_0)$.
5. Seja z o inteiro cuja representação binária é S.

Geração de Curvas Elípticas sobre $\mathbb{F}_p$ /2

6. **for** i **from** 1 **to** s **do**

   6.1 Seja s_i a cadeia binária de g bits que representa o inteiro $(z + i) \bmod 2^g$.

   6.2 Calcule $R_i = H(s_i)$.

7. $R = R_0||R_1||...||R_s$.

8. Seja r o inteiro cuja representação binária é R.

9. **if** $r = 0$ ou $4r + 27 \equiv 0 \pmod{p}$ **then** volte ao passo 2.

10. Escolha $a, b \in \mathbb{F}_p$ ($a \neq 0$ e $b \neq 0$) tal que $r \cdot b^2 \equiv a^3 \pmod{p}$.

11. **return** (S, a, b)

Verificação de que uma Curvas Elíptica sobre $\mathbb{F}_p$ foi Gerada Aleatóriamente

Entrada: Um primo $p > 3$, uma função hash H de l bits, uma semente S de g bits ($g \geq l$), e os coeficientes $a, b \in \mathbb{F}_p$ de uma curva elíptica $E_{a,b}$.

Saída: Aceitação ou rejeição de que a curva $E_{a,b}$ foi gerada utilizando o algoritmo 4.17.

1. $t \leftarrow \lceil (m - 1/l \rceil,\ s \leftarrow \lfloor (t-1)/l \rfloor,\ v \leftarrow t - sl$.
2. Calcule $h = H(S)$, e seja r_0 a cadeia binária de comprimento v bits obtida como os v bits mais à direita de h.
3. Seja R_0 a cadeia binária obtida de r_0 onde o bit mais à esquerda de r_0 é zero ($R_0 = (0, [r_0]_{v-1}, [r_0]_{v-2}, ..., [r_0]_0)$.
4. Seja z o inteiro cuja representação binária é S.

Verificação de uma Curvas Elíptica sobre $\mathbb{F}_p$ Gerada Aleatóriamente /2

5. **for** i **from** 1 **to** s **do**

   5.1 Seja s_i a cadeia binária de g bits que representa o inteiro $(z + i) \bmod 2^g$.

   5.2 Calcule $R_i = H(s_i)$.

6. $R = R_0||R_1||...||R_s$.

7. Seja r o inteiro cuja representação binária é R.

8. **if** $r \cdot b^2 \equiv a^3 \pmod{p}$ **then return** ("aceite"); **else return** ("rejeite").

Geração de Parâmetros

Entrada: A ordem do corpo q, uma reprsentação FR para $\mathbb{F}_q$, o nível de segurança L ($160 \leq L \leq \lfloor \log_2 q \rfloor$ e $2^L \geq 4\sqrt{q}$).

Saída: Os parâmetros $D = (q, \mathrm{FR}, S, a, b, P, n, h)$.

1. Utilize o Algoritmo 4.14 (ou 4.17) para obter S e os coeficientes elípticos $a, b \in \mathbb{F}_q$ verificáveis aleatóriamente.
2. Calcule $N = \#E(\mathbb{F}_q)$.
3. Verifique que N é divisível por um primo grande n. Em caso contrário, volte ao passo 1.
4. Verifique que $n \nmid q^k - 1$ para $1 \leq k \leq 20$. Em caso contrário, volte ao passo 1.

Geração de Parâmetros /2

5. Verifique que $n \neq q$. Em caso contrário, volte ao passo 1.
6. $h = N/n$.
7. Escolha um ponto $P\prime \in E(\mathbb{F}_q)$ e defina $P = hP\prime$.
   Repita até que $P \neq \mathcal{O}$.
8. **return** $(q, \mathrm{FR}, S, a, b, P, n, h)$.

Validação de Parâmetros

Entrada: Os parâmetros $D = (q, \mathrm{FR}, S, a, b, P, n, h)$.

Saída: Aceitação ou Reijeição da validade de D.

1. Verifique que q é uma potência prima ($q = p^m$, p primo e $m \geq 1$)
2. **if** $p = 2$ **then** verifique que m é primo.
3. Verifique que FR é uma representação válida.
4. Verifique que $a, b, x_p = x(P), y_p = y(P)$ são elementos de $\mathbb{F}_q$ (isto é, os elementos estão no formato da representação FR).
5. Verifique que os elementos a, b definem uma curva elíptica sobre $\mathbb{F}_q$ (isto é, $4a^3 + 27b^2 \neq 0$ para $\mathbb{F}_p$, $p > 3$, e $b \neq 0$ para $\mathbb{F}_{2^m}$).
6. **if** a curva elíptica foi gerada aleatóriamente **then**

   6.1 Verifique que S é uma cadeia binária de g bits, onde $g \geq l$ e H é uma função de l bits.

Validação de Parâmetros /2

6.2 Utilize o Algoritmo 4.18 (corpos primos) ou Algoritmo 4.21 (corpos binários) para verificar que a e b foram derivados de forma apropriada de S.

7. Verifique que $P \neq \mathcal{O}$.
8. Verifique que P satisfaz a equação da curva elíptica definida por os coeficientes a e b (isto é, $P \in E_{a,b}(\mathbb{F}_q)$).
9. Verifique que n é primo e que $n > 4\sqrt{q}$.
10. Verifique que $nP = \mathcal{O}$.
11. Calcule $h\prime = \lfloor (\sqrt{q}+1)^2/n \rfloor$ e verifique que $h = h\prime$.
12. Verifique que $n \nmid q^k - 1$ para $1 \leq k \leq 20$.
13. Verifique que $n \neq q$.
14. **if** qualquer verificação falha, **then return** (¨inválido¨); **else return** (¨válido¨).

Parâmetros NIST

- No padrão FIPS 186-2, o NIST recomenda 15 curvas elípticas de diferentes níveis de segurança para o governo federal americano.
- As curvas são de três tipos:
  - Curvas aleatórias sobre corpos primos ($\mathbb{F}_p$)
  - Curvas aleatórias sobre corpos binários ($\mathbb{F}_{2^m}$)
  - Curvas de Koblitz sobre corpos binários ($\mathbb{F}_{2^m}$)

Curvas Aleatórias sobre Corpos Primos ($\mathbb{F}_p$)

- O cofator $h = 1$.
- O coeficiente $a = -3$.
- O coeficiente b aleatório.
- Os primos p recomendados são:

$$p_{192} = 2^{192} - 2^{64} - 1,$$
$$p_{224} = 2^{224} - 2^{96} + 1,$$
$$p_{256} = 2^{256} - 2^{224} + 2^{192} + 2^{96} - 1,$$
$$p_{384} = 2^{384} - 2^{128} - 2^{96} + 2^{32} - 1,$$
$$p_{521} = 2^{521} - 1.$$

Curvas Aleatórias sobre Corpos Primos ($\mathbb{F}_p$) /2

- p: o primo p que define o corpo finito $\mathbb{F}_p$.
- S: a semete para gerar de forma aleatória o coeficiente b.
- r: o número inteiro definido no Algoritmo 4.17 (usa SHA-1).
- a, b: $a = -3$, $rb^2 \equiv a^3 \pmod{p}$.
- n: a ordem do ponto base P (n primo).
- h: o cofator $h = 1$.
- x, y: as coordenadas do ponto base P ($nP = \mathcal{O}$).

Curvas Aleatórias sobre Corpos Binários ($\mathbb{F}_{2^m}$)

- O cofator $h = 2$.
- O coeficiente $a = 1$.
- O coeficiente $b \neq 0$ aleatório.
- Os primos m recomendados são: $163, 233, 283, 409$ e 571.
- Os polinômios irredutíveis são:

$$f_{163}(x) = x^{163} + x^7 + x^6 + x^3 + 1,$$
$$f_{233}(x) = x^{233} + x^{74} + 1,$$
$$f_{283}(x) = x^{283} + x^{12} + x^7 + x^5 + 1,$$
$$f_{409}(x) = x^{409} + x^{87} + 1,$$
$$f_{571}(x) = x^{571} + x^{10} + x^5 + x^2 + 1.$$

Curvas Aleatórias sobre Corpos Primos ($\mathbb{F}_{2^m}$) /2

- m: o primo m que define o corpo binário $\mathbb{F}_{2^m}$.
- $f(x)$: o polinômio irredutível de grau m.
- S: a semete para gerar de forma aleatória o coeficiente b.
- a, b: $a = 1,\ \ b \neq 0$ aleatório em termos de S e SHA-1.
- n: a ordem do ponto base P (n primo).
- h: o cofator $h = 2$.
- x, y: as coordenadas do ponto base P $(nP = \mathcal{O})$.

Curvas de Koblitz sobre Corpos Binários ($\mathbb{F}_{2^m}$)

- O cofator $h = 2$ $(m = 163)$ ou $h = 4$.
- O coeficiente $a = 0$ para $m \in \{233, 283, 409, 571\}$;
  $a = 1$ para $m = 163$.
- O coeficiente $b = 1$.
- Os primos m recomendados são: $163, 233, 283, 409$ e 571.
- Os polinômios irredutíveis são:

$$f_{163}(x) = x^{163} + x^7 + x^6 + x^3 + 1,$$
$$f_{233}(x) = x^{233} + x^{74} + 1,$$
$$f_{283}(x) = x^{283} + x^{12} + x^7 + x^5 + 1,$$
$$f_{409}(x) = x^{409} + x^{87} + 1,$$
$$f_{571}(x) = x^{571} + x^{10} + x^5 + x^2 + 1.$$

Curvas de Koblitz sobre Corpos Binários ($\mathbb{F}_{2^m}$) /2

- m: o primo m que define o corpo binário $\mathbb{F}_{2^m}$.
- $f(x)$: o polinômio irredutível de grau m.
- $a \in \{0, 1\}$, $b = 1$.
- n: a ordem do ponto base P (n primo).
- h: o cofator $h = 2$ ou $h = 4$.
- x, y: as coordenadas do ponto base P ($nP = \mathcal{O}$).

Parâmetros SEC (Standard for Efficient Cryptography)

- O padrão SEC recomenda 33 curvas elípticas de diferentes níveis de segurança.
- As curvas são de quatro tipos:
  - Curvas aleatórias sobre corpos primos ($\mathbb{F}_p$)
  - Curvas de ¨Koblitz¨ sobre corpos primos ($\mathbb{F}_p$)
  - Curvas aleatórias sobre corpos binários ($\mathbb{F}_{2^m}$)
  - Curvas de Koblitz sobre corpos binários ($\mathbb{F}_{2^m}$)

Curvas Aleatórias sobre Corpos Primos ($\mathbb{F}_p$)

- O cofator $h = 1$ ou $h = 4$.
- O coeficiente $a = -3$.
- O coeficiente b aleatório.
- $\lceil \log_2 p \rceil \in \{112, 128, 160, 192, 224, 256, 384, 521\}$.

  $p_{112} = (2^{128} - 3)/76439,$

  $p_{128} = 2^{128} - 2^{97}11,$

  $p_{160} = 2^{160} - 2^{32} - 2^{14} - 2^{12} - 2^9 - 2^8 - 2^3 - 2^2 - 1,$ K.

  $p_{160} = 2^{160} - 2^{31} - 1$

  $p_{192} = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1,$ K.

  $p_{224} = 2^{224} - 2^{32} - 2^{12} - 2^{11} - 2^9 - 2^7 - 2^4 - 2 - 1,$ K.

  $p_{256} = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1,$ K.

  $p_{192},\ p_{224},\ p_{256},\ p_{384},\ p_{521}$ NIST.

Curvas Aleatórias sobre Corpos Primos ($\mathbb{F}_p$) /2

- p: o primo p que define o corpo finito $\mathbb{F}_p$.
- S: a semete para gerar de forma aleatória o coeficiente b.
- r: o número inteiro definido no Algoritmo 4.17 (usa SHA-1).
- a, b: $a = -3$, $rb^2 \equiv a^3 \pmod{p}$.
- n: a ordem do ponto base P (n primo).
- h: o cofator $h = 1$ ou $h = 4$.
- x, y: as coordenadas do ponto base P ($nP = \mathcal{O}$).

Curvas de Koblitz sobre Corpos Primos ($\mathbb{F}_p$)

- O cofator $h = 1$.
- O coeficientes a, b: (0,7), (0,3), (0,5), (0,7).
- $\lceil \log_2 p \rceil \in \{160, 192, 224, 256\}$.

$$p_{160} = 2^{160} - 2^{32} - 2^{14} - 2^{12} - 2^9 - 2^8 - 2^3 - 2^2 - 1$$

$$p_{192} = 2^{192} - 2^{32} - 2^{12} - 2^8 - 2^7 - 2^6 - 2^3 - 1$$

$$p_{224} = 2^{224} - 2^{32} - 2^{12} - 2^{11} - 2^9 - 2^7 - 2^4 - 2 - 1$$

$$p_{256} = 2^{256} - 2^{32} - 2^9 - 2^8 - 2^7 - 2^6 - 2^4 - 1$$

Curvas Aleatórias sobre Corpos Binários ($\mathbb{F}_{2^m}$)

- O cofator $h = 2$.
- O coeficiente $Tr(a) = 1$.
- O coeficiente $b \neq 0$ aleatório.
- Os primos m são: $113, 131, 163, 193, 233, 283, 409$ e 571.
- Os polinômios irredutíveis são:

  $f_{113}(x) = x^{113} + x^9 + 1$,

  $f_{131}(x) = x^{131} + x^8 + x^3 + x^2 + 1$,

  $f_{193}(x) = x^{193} + x^{15} + 1$,

  $f_{239}(x) = x^{239} + x^{36} + 1$ ou $x^{239} + x^{158} + 1$.

  $f_{163}(x), f_{233}(x), f_{283}(x), f_{409}(x), f_{571}(x)$, NIST.

Curvas de Koblitz sobre Corpos Binários ($\mathbb{F}_{2^m}$)

- O cofator $h = 2$ ou $h = 4$.
- O coeficientes a, b: (1,1), (0,1), (0,1), (0,1), (0,1), (0,1).
- $m \in \{163, 233, 239, 283, 409, 571\}$.

Artigos: parâmetros

- FIPS 186-2. Digital Signature Standard (DSS). Federal Information Processing Standard Publication 186-2, National Institute of Standards and Technology, 2000.
- *SEC 2: Recommended Elliptic Curve Domain Parameters*, 2000. `http://www.secg.org`
- *The security of DSA and ECDSA*
  Bypassing the Standard Elliptic Curve Certification Scheme,
  Serge Vaudenay, PKC 2003, LNCS 2567, 2003.
- *Certicom Proposal to Revise*
  SEC1: Elliptic Curve Cryptography, Version 1.0
  Daniel R. L. Brown, 14 de Janeiro de 2005.

Artigos: implementação em software

- *Software implementation of elliptic curve cryptography over binary fields*, D. Hankerson, J. López e A. Menezes, CHES 2000, LNCS 1965, 2000.
- *Software implementation of the NIST elliptic curves over prime fields*, M. Brown, D. Hankerson, J. López e A. Menezes, CT-RSA, LNCS 2020, 2001.
- *Performance comparisons of elliptic curve systems in software*, K. Fong, D. Hankerson, J. López, A. Menezes, M. Tucker, The 5th Workshop on Elliptic Curves Cryptography (ECC-2001).
- *Field Inversion and Point Halving Revisited*, K. Fong, D. Hankerson, J. López e A. Menezes, IEEE Transsaction on Computers, VOL 53, 2004.