# 9. Criptografia basada en pairings

En el capítol anterior s'han presentat les corbes el·líptiques i alguns dels seus usos en criptografia. Tot i això, no ens hem endinsat en una de les construccions més populars en els últims anys en criptografia basada en corbes el·líptiques: els *pairings*.

Algunes corbes el·líptiques tenen una estructura addicional anomenada aparellament (en anglès, es coneix com a *pairing*), que obre la porta a tot un nou conjunt d'eines criptogràfiques. En particular, els *pairings* es fan servir en criptografia per a atacar criptosistemes basats en el logaritme discret i també en la construcció de noves primitives criptogràfiques.

En aquest capítol descriurem les propietats dels *pairings* i la seva definició (tot explicant les eines matemàtiques necessàries per a construir-los), i veurem alguns algorismes criptogràfics basats en aquests.

## 9.1 Propietats dels *pairings*

Existeixen diferents tipus de *pairings*, però no tots són adequats per a usos criptogràfics. Els únics *pairings* coneguts que són útils en criptografia i, a més, eficientment computables, són els *pairings* de Weil i de Tate sobre corbes el·líptiques.

Més enllà de la seva definició explícita, que veurem més endavant, a continuació en descrivim les propietats que els caracteritzen.

**Definició 9.1** Siguin $\mathbb{G}_0, \mathbb{G}_1$ i $\mathbb{G}_T$ grups cíclics d'ordre primer $q$ amb $G_0 \in \mathbb{G}_0$ i $G_1 \in \mathbb{G}_1$ elements generadors dels grups. Diem que $\mathbb{G}_0$ i $\mathbb{G}_1$ són els grups d'origen i $\mathbb{G}_T$ el grup objectiu.

Un aparellament (conegut en anglès com a ***pairing***) és una aplicació $e : \mathbb{G}_0 \times \mathbb{G}_1 \to \mathbb{G}_T$ que satisfà dues propietats:

1. És **bilinial**, és a dir, per a tot $P_0, Q_0 \in \mathbb{G}_0$ i $P_1, Q_1 \in \mathbb{G}_1$:

$$e(P_0, P_1 + Q_1) = e(P_0, P_1) * e(P_0, Q_1)$$

$$e(P_0+Q_0,P_1) = e(P_0,P_1) * e(Q_0,P_1)$$

2. És **no degenerat**, és a dir, $g_T = e(G_0,G_1)$ és un generador de $\mathbb{G}_T$.

És a dir, l'aplicació $e$ és lineal per a les dues entrades. La bilinialitat dels *pairings* implica la següent propietat, en la qual es basen les construccions criptogràfiques que els fan servir:

$$e(\alpha P_0, \beta P_1) = e(P_0,P_1)^{\alpha\beta} = e(\beta P_0, \alpha P_1)$$

La propietat es deriva directament de la definició de bilinearitat:

$$e(\alpha P_0, \beta P_1) = e(P_0, \beta P_1)^{\alpha} = e(\alpha P_0, P_1)^{\beta} = e(P_0,P_1)^{\alpha\beta}$$

**Notació**

Noteu que s'ha fet servir notació additiva a $\mathbb{G}_0$ i $\mathbb{G}_1$ i multiplicativa a $\mathbb{G}_T$. Això és així ja que com veurem a continuació, els grups $\mathbb{G}_0$ i $\mathbb{G}_1$ estan definits per punts de corbes el·líptiques (i en aquest capítol anomenem suma a l'operació de grup) i, en canvi, el grup $\mathbb{G}_T$ serà un grup multiplicatiu.

Diem que un aparellament és **simètric** si $\mathbb{G}_0 = \mathbb{G}_1$ i **asimètric** en cas contrari.

La definició explícita dels *pairings* de Weil i de Tate és complexa i, per això, sovint s'obvia aquesta definició i es descriuen únicament les propietats que els caracteritzen. Entendre les propietats dels *pairings* és suficient per poder comprendre els esquemes criptogràfics que se'n deriven però, d'altra banda, entendre com es calculen ens permet aproximar-nos més detalladament a la seva estructura. Les dues seccions següents estan centrades en la definició explícita dels *pairings*: en primer lloc, es detallen un conjunt d'eines matemàtiques que permeten definir els *pairings*; a continuació, es presenta la definició explícita dels *pairings* de Weil i de Tate, tot exemplificant el càlcul amb una corba petita. El lector interessat pot doncs aprofundir en la construcció dels *pairings* seguint la lectura de la propera secció. D'altra banda, el lector que no desitgi aprofundir en la construcció dels *pairings*, pot passar directament a la Secció 9.4 per focalitzar-se directament en els algorismes criptogràfics que es basen en les propietats que proporcionen els *pairings*.

## 9.2 Eines matemàtiques per a la construcció dels *pairings*

En aquesta secció es presenten les eines matemàtiques que permeten definir explícitament els *pairings* de Weil i de Tate. En primer lloc, s'estén la definició de corbes el·líptiques que hem vist fins ara sobre $\mathbb{Z}_p$ a cossos finits amb un nombre d'elements no primer. A continuació, es defineix la $r$-torsió dels punts d'una corba el·líptica i s'observa l'estructura dels subgrups que conforma. Finalment, es descriu el concepte de divisor d'una funció i es presenta el seu ús en el context de la criptografia de corbes el·líptiques.

### 9.2.1 Corbes el·líptiques sobre cossos estesos

Fins ara hem fet servir corbes el·líptiques definides sobre cossos finits amb un nombre primer d'elements ($\mathbb{Z}_p$). Ara bé, també es poden construir corbes el·líptiques sobre altres cossos finits, com ara cossos amb un nombre d'elements potència d'un primer $\mathbb{F}_{p^d}$. Al capítol de *Fonaments matemàtics* ja hem vist com construir aquests cossos finits fent servir polinomis irreductibles. L'ús d'aquests cossos per a la creació de corbes el·líptiques és anàleg al cas de cossos finits amb nombre primer d'elements.

**Exemple 9.1 Exemple de corba el·líptica sobre cos estès**

Com ja hem vist a l'Exemple 8.6, la corba el·líptica $E/\mathbb{Z}_{11} : y^2 = x^3 - 5x + 5$ té 17 elements:

$$[\mathscr{O},(0,4),(0,7),(1,1),(1,10),(2,5),(2,6),(4,4),(4,7),(6,2),(6,9),(7,4),(7,7),\\(8,2),(8,9),(10,3),(10,8)]$$

La corba està definida sobre $\mathbb{Z}_{11}$, un cos finit amb un nombre primer d'elements (11).

La corba el·líptica $E/(\mathbb{Z}_{11}/z^2+1) : y^2 = x^3 - 5x + 5$ té en canvi 119 elements. La corba està definida sobre el cos finit d'$11^2$ elements fent servir el polinomi $z^2+1$, irreductible a $\mathbb{Z}_{11}$ i de grau 2.

Alguns dels 119 elements d'aquesta corba el·líptica són:

$$[\mathscr{O},(0,4),(0,7),(1,1),(1,10),(2,5),(2,6),(4,4),(4,7),(6,2),(6,9),(7,4),(7,7),\\(8,2),(8,9),(10,3),(10,8),(5,4z),(5,7z),(9,2z),(9,9z),(z+2,z+3),(z+2,10z),\\(z+4,4z+8),(z+4,7z+3),\ldots]$$

Podem comprovar que aquests punts efectivament pertanyen a la corba verificant que compleixen l'equació que la defineix. Així, per exemple, per al punt $(5,4z)$, tenim que:

$$\begin{aligned} y^2 &= x^3 - 5x + 5 \mod 11\\ (4z)^2 &= 5^3 - 5\cdot 5 + 5 \mod 11\\ 16z^2 &= 105 \mod 11\\ 5z^2 &= 6 \mod 11\\ 5z^2 - 6 &= 0 \mod 11 \end{aligned}$$

Efectivament, el residu de dividir $5z^2-6$ entre $z^2+1$ a $\mathbb{Z}_{11}$ és 0 (ja que $5(z^2+1) = 5z^2+5 = 5z^2-6 \mod 11$).

### 9.2.2 Els punts de la $r$-torsió

En gran part d'aquest capítol hem treballat amb corbes el·líptiques definides sobre cossos finits amb un nombre primer d'elements. A la secció anterior hem vist que també podem definir corbes sobre cossos estesos amb ordre la potència d'un primer. A continuació veurem una de les construccions que es poden definir quan treballem amb grups sobre cossos estesos, els grups formats per $r$-torsions.

**Definició 9.2** Sigui $E$ una corba el·líptica definida sobre un cos finit $\mathbb{Z}_p$ i $n$ un primer divisor de $\#E/\mathbb{Z}_p$. El **grau d'immersió** (en anglès, parlem d'*embedding degree*) d'$E$ respecte a $n$ és el menor enter $k$ tal que $n$ divideix $p^k - 1$.

Per al cas $n = \#E/\mathbb{Z}_p$, direm simplement que $k$ és el grau d'immersió d'$E$.

Una vegada definit el grau d'immersió, passem a definir la $r$-torsió:

**Definició 9.3** Sigui $r$ un primer diferent de $p$. Es defineixen els punts de la **r-torsió**, $E[r]$, com el conjunt de punts $P$ que pertanyen a $E/\mathbb{F}_{p^k}$ tals que $rP = \mathscr{O}$. És a dir,

$$E[r] = \{P \in E/\mathbb{F}_{p^k} \text{ tals que } rP = \mathscr{O}\}$$

amb $k$ el grau d'immersió d'$E$ respecte a $r$.

**Exemple 9.2 Exemple de 3-torsió**

La corba $E/\mathbb{Z}_{11} : y^2 = x^3 + 10x + 4$ té 15 elements ($\#E/\mathbb{Z}_{11} = 15$):

$$[\mathscr{O},(0,2),(0,9),(1,2),(1,9),(4,3),(4,8),(5,5),(5,6),(6,4),(6,7),(9,3),(9,8),(10,2),(10,9)]$$

El grau d'immersió d'$E$ respecte a $n = 3$ (amb $3 \mid 15$) és $k = 2$, ja que 2 és el menor enter tal que $n \mid p^k - 1$. Noteu que per a $k = 1$ la condició no es compleix, ja que $3 \nmid 11^1 - 1$. En canvi, per a $k = 2$ tenim que $3 \mid 11^2 - 1$.

Hi ha 9 punts a la 3-torsió:

$$\begin{aligned} E[3] =& \{P \in E/(\mathbb{Z}_{11}/z^2+1) \text{ tals que } 3P = \mathscr{O}\} = \\ =& [\mathscr{O},(1,2),(1,9),(8,3z),(8,8z),(2z+1,z+9),(2z+1,10z+2),(9z+1,z+2), \\ & (9z+1,10z+9)] \end{aligned}$$

Dels 9 punts de la 3-torsió, 3 es troben al cos base ($\mathscr{O},(1,2),(1,9) \in E/\mathbb{Z}_{11}$) i la resta al cos estès $E/(\mathbb{Z}_{11}/z^2+1)$.

A continuació comprovem que els punts compleixen la condició per pertànyer a la 3-torsió. Com a exemple, mostrem els càlculs per als punts $(1,2)$ i $(8,3z)$:

$$\begin{aligned} P &= (1,2) \\ 2P &= (1,9) \\ 3P &= \mathscr{O} \end{aligned}$$

$$\begin{aligned} P &= (8,3z) \\ 2P &= (8,8z) \\ 3P &= \mathscr{O} \end{aligned}$$

És interessant notar l'estructura dels subgrups de la 3-torsió: la 3-torsió té 4 subgrups cíclics, tots ells d'ordre 3:

| Ordre | Subgrup |
|---|---|
| 3 | $\{\mathscr{O},\ (1,2),\ (1,9)\}$ |
| 3 | $\{\mathscr{O},\ (8,3z),\ (8,8z)\}$ |
| 3 | $\{\mathscr{O},\ (2z+1,z+9),\ (2z+1,10z+2)\}$ |
| 3 | $\{\mathscr{O},\ (9z+1,z+2),\ (9z+1,10z+9)\}$ |

**Exercici 9.1** Donada la corba de l'exemple anterior (Exemple 9.2), calculeu el grau d'immersió d'E respecte a $n = 5,7$ i 15.

### 9.2.3 El divisor d'una funció

Per acabar la presentació de les eines matemàtiques que ens permetran definir els *pairings*, exposarem el concepte de divisor d'una funció.

Abans, però, definirem els conceptes de funció racional, i els zeros i pols d'aquestes.

**Definició 9.4** Una **funció racional** $f(x)$ és una funció que pot ser expressada com a una divisió de polinomis en la qual el denominador no és 0 (és a dir, $f(x) = q(x)/p(x)$ amb $p(x) \neq 0$). Quan el numerador $q(x)$ i el denominador $p(x)$ no tenen arrels en comú, diem que la funció està en **forma reduïda**.

**Definició 9.5** Els **zeros** d'una funció racional són els punts en què $f(x) = 0$ mentre que els **pols** són els punts en què $f(x) = \pm\infty$.

Donada una funció racional en forma reduïda expressada amb els polinomis factoritzats:

$$f(x) = \frac{a(x-\alpha_1)^{\mu_1}(x-\alpha_2)^{\mu_2}\cdots(x-\alpha_n)^{\mu_n}}{b(x-\beta_1)^{\nu_1}(x-\beta_2)^{\nu_2}\cdots(x-\beta_n)^{\nu_n}} \tag{9.1}$$

els zeros corresponen als valors $\alpha_i$ mentre que els pols són els $\beta_j$. Noteu que $\alpha_i \neq \beta_j$ per a qualsevol $i$ i $j$, ja que la funció es troba en forma reduïda. Direm que els $\mu_i$ i els $\nu_i$ són la multiplicitat de cada zero i de cada pol, respectivament.

A més, si el grau del polinomi del numerador difereix del grau del polinomi del denominador ($deg(q(x)) \neq deg(p(x))$), hi haurà un zero o un pol a l'infinit. En concret, si $deg(q(x)) > deg(p(x))$ hi haurà un zero a l'infinit i si $deg(q(x)) < deg(p(x))$ hi haurà un pol a l'infinit. La multiplicitat del zero o del pol serà la diferència entre els graus dels polinomis ($|deg(q(x)) - deg(p(x))|$), de manera que l'ordre total de zeros i pols és igual.

Els **divisors** són una eina que es fa servir per descriure els zeros i els pols d'una funció. Donada una funció racional:

$$f(x) = c\prod_i (x-\alpha_i)^{\mu_i} \tag{9.2}$$

escriurem:

$$div(f) = \sum_i \mu_i \langle \alpha_i \rangle$$

En primer lloc, noteu com l'equació 9.2 és equivalent a l'equació 9.1. Només cal expressar els factors que es trobaven al denominador amb valors negatius als exponents. En segon lloc, és important interpretar el divisor com a tal, i evitar operar com si estiguéssim treballant amb números (o, més endavant, amb punts de la corba). Per aquest motiu, es fan servir les claus $\langle$ i $\rangle$ per denotar que no estem parlant d'un enter (o un punt) $\alpha_i$ sinó d'un zero o un pol en aquell punt (més endavant tornarem a fer incís en aquest detall, quan definim els divisors sobre corbes el·líptiques). Per últim, cal destacar que efectivament el divisor ens permet anotar els pols i zeros de la funció, ja que ens descriu a on són i quina multiplicitat tenen.

**Exemple 9.3 Zeros i pols d'una funció**

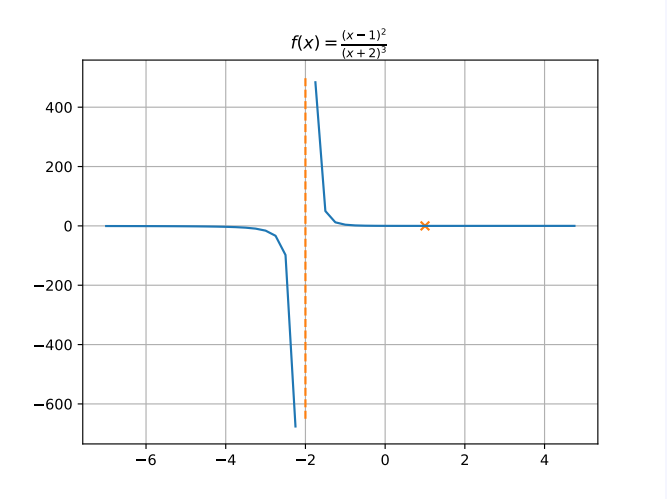
La funció:

$$f(x) = \frac{(x-1)^2}{(x+2)^3} = (x-1)^2(x+2)^{-3}$$

té un zero de multiplicitat 2 a $x = 1$ i un pol de multiplicitat 3 a $x = -2$. Addicionalment, la funció té

un zero de multiplicitat 1 a l'infinit, ja que el grau del denominador és superior al del numerador en una unitat.

Observant la representació gràfica de la funció, podem veure que efectivament és zero en $x = 1$ i tendeix a infinit en $x = -2$:




Per tant, podem descriure els pols i zeros de la funció $f$ amb el divisor:

$$div(f) = 2\langle 1\rangle - 3\langle -2\rangle + \langle \infty\rangle$$

**Exercici 9.2** Indiqueu el divisor de la funció racional:

$$f(x) = \frac{(x-1)^3(x+5)^2}{(x-12)^4}$$

És interessant notar com es reflecteixen les operacions entre funcions en els divisors:

$$div(f \cdot g) = div(f) + div(g) \tag{9.3}$$

$$div(f/g) = div(f) - div(g) \tag{9.4}$$

**Exemple 9.4 Operacions entre funcions**

Donades les funcions:

$$f(x) = (x-1)^2$$
$$g(x) = \frac{1}{(x+2)^3}$$

amb divisors:

$$div(f) = 2\langle 1\rangle - 2\langle \infty\rangle$$

$$div(g) = -3\langle -2\rangle + 3\langle \infty\rangle$$

podem comprovar com el divisor del seu producte és la suma del divisor de cadascuna d'elles:

$$f(x)g(x) = (x-1)^2 \cdot \frac{1}{(x+2)^3} = \frac{(x-1)^2}{(x+2)^3} = (x-1)^2(x+2)^{-3}$$

$$div(f \cdot g) = 2\langle 1\rangle - 3\langle -2\rangle + \langle \infty\rangle = div(f) + div(g)$$

i que el divisor del seu quocient és la resta del divisor de cadascuna d'elles:

$$\frac{f(x)}{g(x)} = \frac{(x-1)^2}{\frac{1}{(x+2)^3}} = (x-1)^2(x+2)^3$$

$$div(f/g) = 2\langle 1\rangle + 3\langle -2\rangle - 5\langle \infty\rangle = div(f) - div(g)$$

A més, dues funcions que tenen el mateix divisor són iguals excepte per una constant i el divisor d'una funció és zero si i només si la funció és constant.

En la criptografia basada en corbes el·líptiques, es fan servir divisors per descriure els punts d'intersecció d'una corba $E$ amb una funció $f(x)$, de manera que s'utilitzen per descriure els zeros i els pols de la funció $E - f(x)$.

**Definició 9.6** Sigui $E$ una corba el·líptica. Un **divisor sobre** $E$ és una suma formal:

$$D = \sum_{P \in E} n_p \langle P\rangle$$

on els $n_p$ són enters i on tots els $n_p$ excepte un nombre finit són zero.

És a dir, ara els zeros i pols seran punts de la corba el·líptica ($P \in E$), i n'expressarem la seva multiplicitat amb els enters $n_p$, de la mateixa manera que ho fèiem anteriorment amb funcions racionals definides sobre els reals.

De nou, és important diferenciar la suma de punts d'una corba (que denotàvem amb el símbol $+$, per exemple, $P_1 + P_2$) i la multiplicació escalar d'un enter per un punt (que denotàvem per $sP$ o bé $s \cdot P$) de la suma formal que conforma un divisor (que denotem fent servir també el símbol $+$ i de manera similar a la multiplicació escalar, però indicant els punts dins de les claus, $s_1\langle P_1\rangle + s_2\langle P_2\rangle$).

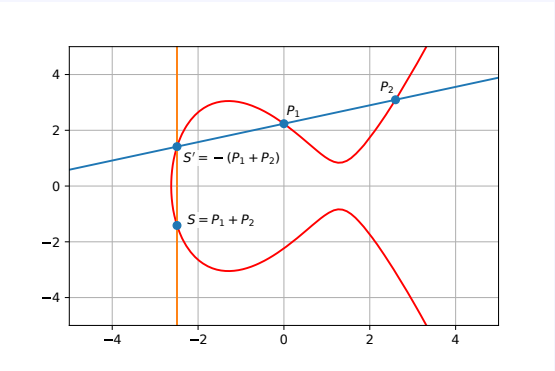
**Exemple 9.5 Divisors de les rectes que defineixen la suma**

Ja hem fet servir funcions sobre corbes el·líptiques a l'inici d'aquest capítol, quan descrivíem com sumar dos punts d'una corba el·líptica. Recapitulant, el procediment per calcular la suma entre dos punts requeria del càlcul de la recta que passava per aquests dos punts (o bé de la recta tangent a la corba en aquell punt, en l'operació de doblat) i de la recta vertical que passava pel tercer (o segon, en el cas del doblat) punt d'intersecció de la corba. A continuació descriurem els divisors d'aquestes funcions.

Anomenem $l_{P_1,P_2}$ a la funció que representa la recta que passa pels punts $P_1$ i $P_2$ (recta que tracem per a calcular la suma $P_1 + P_2$, representada en blau a la figura següent) per a $P_1 \neq P_2$. Aleshores podem escriure els divisors de la funció $l_{P_1,P_2}$:

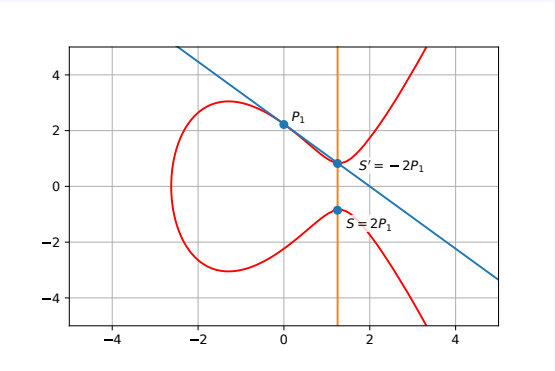
$$div(l_{P_1,P_2}) = \langle P_1\rangle + \langle P_2\rangle + \langle -(P_1+P_2)\rangle - 3\langle \mathcal{O}\rangle$$

ja que efectivament la funció $l_{P_1,P_2}$ interseca la corba $E$ en els punts $P_1$, $P_2$ i $-(P_1+P_2)$ (el punt simètric al resultat de la suma).




Per al cas en què els dos punts a sumar són iguals (línia blava a la figura següent), aleshores tenim que:

$$div(l_{P_1,P_1}) = 2\langle P_1\rangle + \langle -2P_1\rangle - 3\langle \mathcal{O}\rangle$$




Anomenem $v_S$ a la funció que representa la recta vertical que passa pels punts $S$ i $-S$ (línia taronja de les dues figures anteriors). Aleshores, podem dir que:

$$div(v_S) = \langle S\rangle + \langle -S\rangle - 2\langle \mathcal{O}\rangle$$

ja que la funció $v_S$ interseca la corba $E$ en els punts $S$ i $S'$.

Tot i que ja es pot intuir de la descripció que s'ha fet dels divisors de funcions racionals, en els propers paràgrafs acabarem d'aprofundir en la multiplicitat del punt $\mathcal{O}$ de les funcions anteriors.

A continuació es defineixen algunes característiques dels divisors.

El **suport** d'un divisor $D$ és el conjunt de tots els punts $P$ que tenen el coeficient $n_p$ diferent de zero. Dos divisors $D_1$ i $D_2$ tenen un **suport disjunt** si la intersecció entre el suport de tots dos és un conjunt buit (és a dir, no tenen punts en comú al suport).

El **grau** d'un divisor $D$ és la suma dels coeficients $n_p$:

$$deg(D) = \sum_{P \in E} n_p$$

Un divisor D és un **divisor principal** si existeix una funció racional $f$ tal que $D = div(f)$, és a dir, si representa els zeros i els pols d'una funció racional. Equivalentment podem afirmar que un divisor és principal si i només si té grau zero i $\sum_{P \in E} n_P P = \mathscr{O}$. Noteu que en aquesta última expressió no hi ha les claus $\langle$ i $\rangle$, de manera que aquí s'està calculant el sumatori de multiplicacions escalars (a diferència de l'expressió del divisor).

**Exemple 9.6 Divisors de les rectes que defineixen la suma**

A l'Exemple 9.5 hem vist els divisors de les rectes que es fan servir per a sumar punts ($l_{P_1,P_2}$, $l_{P_1,P_1}$ i $v_S$), deduint-los a partir de la definició de les pròpies rectes:

$$div(l_{P_1,P_2}) = \langle P_1 \rangle + \langle P_2 \rangle + \langle -(P_1 + P_2) \rangle - 3\langle \mathscr{O} \rangle$$
$$div(l_{P_1,P_1}) = 2\langle P_1 \rangle + \langle -2P_1 \rangle - 3\langle \mathscr{O} \rangle$$
$$div(v_S) = \langle S \rangle + \langle -S \rangle - 2\langle \mathscr{O} \rangle$$

No hem descrit amb detall, però, perquè aquests divisors inclouen el punt a l'infinit $\mathscr{O}$ ni la seva multiplicitat. Doncs bé, com que les funcions que defineixen aquestes rectes són funcions racionals, els seus divisors són principals i, per tant, han de tenir grau zero. Per això la multiplicitat de $\mathscr{O}$ a $div(l)$ i $div(v)$ és $-3$ i $-2$, respectivament.

**Exemple 9.7 Divisors principals**

Sigui $P \in E$ un punt de la corba el·líptica $E$, tal que l'ordre de $P$ és $n$. Aleshores, el divisor:

$$D = n\langle P \rangle - n\langle \mathscr{O} \rangle$$

és un divisor principal.

En efecte, el divisor compleix les dues propietats necessàries per ser principal:

$$deg(D) = \sum_{P \in E} n_p = n - n = 0$$
$$\sum_{P \in E} n_P P = nP - n\mathscr{O} = \mathscr{O}$$

Noteu que com que l'ordre del punt $P$ és $n$, $nP = \mathscr{O}$.

Cal destacar també que en aquest cas sabem que existeix una funció racional que té com a divisor $D$ (ja que el divisor és principal) però, a diferència de l'exemple anterior (Exemple 9.6), ara no la coneixem. Més endavant, a la propera subsecció, veurem com construir funcions que tinguin un divisor concret.

Dos divisors $D_1$ i $D_2$ són **equivalents** si difereixen en un divisor principal, és a dir, si $D = D_1 - D_2$ és un divisor principal. Com que un divisor principal té grau zero, dos divisors equivalents tenen el mateix grau. Per denotar que dos divisors són equivalents, escrivim $D_1 \sim D_2$.

Ja per acabar, només ens queda descriure l'avaluació d'una funció en un divisor, cosa que ens permetrà il·lustrar el teorema de la reciprocitat de Weil. L'**avaluació d'una funció racional $f$ en un divisor** $D = \sum_{P \in E} n_P \langle P \rangle$ amb el suport de D i de $div(f)$ disjunts es defineix com:

$$f(D) = \prod_{P \in E} f(P)^{n_P}$$

Per tal de poder avaluar la funció $f$ en un divisor $D$, els suports de $D$ i de $div(f)$ han de ser disjunts, ja que si $P \in div(f)$ aleshores $f(P)$ seria zero o infinit i, per tant, $f(D)$ també ho seria.

Això ens permet descriure el teorema de la reciprocitat de Weil, que és la base de moltes de les propietats que es fan servir en criptografia basada en *pairings*:

**Teorema 9.1** Siguin $f$ i $g$ dues funcions diferents de zero en una corba el·líptica tals que $div(f)$ i $div(g)$ tenen suports disjunts. Aleshores, $f(div(g)) = g(div(f))$.

**Exemple 9.8 Reciprocitat de Weil**

A continuació comprovarem la reciprocitat de Weil per a una corba i funcions concretes, tot aprofitant per exemplificar els diferents conceptes presentats en aquesta secció. Aquest exemple està basat en l'exemple 3.3.1 del manual *Pairings for beginners* de *Craig Costello*.

Donada una corba $E/\mathbb{Z}_{503} : y^2 = x^3 + 1$ i les funcions $f(x,y) = \frac{20y+9x+179}{199y+187x+359}$ i $g(x,y) = y + 251x^2 + 129x + 201$ sobre $E$, els divisors d'$f$ i $g$ són:

$$div(f) = 2\langle 433,98\rangle + \langle 232,113\rangle - \langle 432,27\rangle - 2\langle 127,258\rangle$$
$$div(g) = \langle 413,369\rangle + \langle 339,199\rangle + \langle 147,443\rangle + \langle 124,42\rangle - 4\langle \mathcal{O}\rangle$$

Pel que fa als divisors d'$f$, noteu com efectivament els punts $(433,98)$ i $(232,113)$ són punts on el numerador d'$f$ és 0; i els punts $(432,27)$ i $(127,258)$ són punts on el denominador és 0.

$$\begin{aligned}
(433,98) &: 20y + 9x + 179 = 20 \cdot 98 + 9 \cdot 433 + 179 \mod 503 = 0 \\
(232,113) &: 20y + 9x + 179 = 20 \cdot 113 + 9 \cdot 232 + 179 \mod 503 = 0 \\
(432,27) &: 199y + 187x + 359 = 199 \cdot 27 + 187 \cdot 432 + 359 \mod 503 = 0 \\
(127,258) &: 199y + 187x + 359 = 199 \cdot 258 + 187 \cdot 127 + 359 \mod 503 = 0
\end{aligned}$$

A més, tots ells pertanyen a la corba $E$:

$$\begin{aligned}
98^2 &= 433^3 + 1 \mod 503 \\
113^2 &= 232^3 + 1 \mod 503 \\
27^2 &= 432^3 + 1 \mod 503 \\
256^2 &= 127^3 + 1 \mod 503
\end{aligned}$$

De manera similar, pel que fa als divisors de $g$, els punts $(413,369)$, $(339,199)$, $(147,443)$, $(124,42)$ són punts on el numerador és 0 i, a més, pertanyen a la corba $E$ (els càlculs són anàlegs i es deixen com a exercici per al lector).

Pel que fa al grau i el suport dels divisors d'$f$ i $g$, podem dir que:

$$deg(div(f)) = 2 + 1 - 1 - 2 = 0$$
$$deg(div(g)) = 1 + 1 + 1 + 1 - 4 = 0$$

$$sup(div(f)) = \{(433,98),(232,113),(432,27),(127,258)\}$$
$$sup(div(g)) = \{(413,369),(339,199),(147,443),(124,42),\mathscr{O}\}$$
$$sup(div(f)) \cap sup(div(g)) = \emptyset$$

Els dos divisors són principals, ja que el seu grau és 0 i, a més:

$$div(f) = 2(433,98) + (232,113) - (432,27) - 2(127,258) = \mathscr{O}$$

$$div(g) = (413,369) + (339,199) + (147,443) + (124,42) - 4 \cdot \mathscr{O} = \mathscr{O}$$

Com que el suport dels dos divisors és disjunt, podem calcular l'avaluació de la funció $f$ en el divisor $div(g)$:

$$\begin{aligned} f(div(g)) =& f(413,369) \cdot f(339,199) \cdot f(147,443) \cdot f(124,42) \cdot f(\mathscr{O})^{-4} = \\ =& \frac{20 \cdot 369 + 9 \cdot 413 + 179}{199 \cdot 369 + 187 \cdot 413 + 359} \cdot \frac{20 \cdot 199 + 9 \cdot 339 + 179}{199 \cdot 199 + 187 \cdot 339 + 359} \cdot \frac{20 \cdot 443 + 9 \cdot 147 + 179}{199 \cdot 443 + 187 \cdot 147 + 359} \cdot \\ & \cdot \frac{20 \cdot 42 + 9 \cdot 124 + 179}{199 \cdot 42 + 187 \cdot 124 + 359} \cdot \left( \frac{20 \cdot 1 + 9 \cdot 0 + 179}{199 \cdot 1 + 187 \cdot 0 + 359} \right)^{-4} = 321 \end{aligned}$$

Cal remarcar que per a la resta de punts de la corba, els enters $n_p$ són 0 i, per tant, el resultat del seu factor és sempre 1.

**Coordenades projectives**

Noteu que per a calcular l'avaluació d'$f$ en el punt $\mathscr{O}$ hem considerat que $x = 0$ i $y = 1$. Això és així ja que s'ha utilitzat la representació en coordenades projectives, de manera que $\mathscr{O} = (0:1:0)$. El lector interessat en aprendre aquesta representació pot consultar *The arithmetics of Elliptic Curves* de Joseph H. Silverman.

També podem calcular l'avaluació de la funció $g$ en el divisor $div(f)$:

$$\begin{aligned} g(div(f)) =& g(433,98)^2 \cdot g(232,113) \cdot g(432,27)^{-1} \cdot g(127,258)^{-2} = \\ =& (98 + 251 \cdot 433^2 + 129 \cdot 433 + 201)^2 \cdot (113 + 251 \cdot 232^2 + 129 \cdot 232 + 201) \cdot \\ & \cdot (27 + 251 \cdot 432^2 + 129 \cdot 432 + 201)^{-1} \cdot (258 + 251 \cdot 127^2 + 129 \cdot 127 + 201)^{-2} = \\ =& 321 \end{aligned}$$

Així doncs, efectivament, $f(div(g)) = g(div(f)) = 321$.

### 9.2.4 Construcció de funcions a partir del divisor

Com a últim apunt abans de presentar la construcció explícita dels *pairings* de Weil i de Tate, veurem com podem construir funcions amb un divisor donat.

Un divisor d'especial importància en la definició dels *pairings* és el divisor:

$$div(f_{m,P}) = m\langle P \rangle - \langle mP \rangle - (m-1)\langle \mathscr{O} \rangle$$

per a qualsevol enter $m$ i qualsevol $P \in E$. El divisor és un divisor principal, ja que el grau és zero (en efecte, $m - 1 - (m-1) = 0$) i $\sum_{P \in E} n_P P = mP - mP - (m-1)\mathscr{O} = \mathscr{O}$. Per tant, sempre existeix una funció racional

$f_{m,P}$ per a qualsevol $m$ i $P$.

A més, si el punt $P$ pertany a la $r$-torsió, aleshores:

$$\begin{aligned} div(f_{r,P}) &= r\langle P\rangle - \langle rP\rangle - (r-1)\langle \mathscr{O}\rangle = \\ &= r\langle P\rangle - \langle \mathscr{O}\rangle - (r-1)\langle \mathscr{O}\rangle = \\ &= r\langle P\rangle - r\langle \mathscr{O}\rangle \end{aligned}$$

Sabem que existeix una funció racional $f_{m,P}$ per a qualsevol $m$ i $P$ ja que el divisor és principal, però necessitem saber com trobar aquesta funció. Doncs bé, podem construir $f_{m,P}$ iterativament, a partir d'una funció constant de divisor zero, de la manera següent:

$$f_{m+1,P} = f_{m,P} \cdot \frac{l_{mP,P}}{v_{(m+1)P}} \tag{9.5}$$

on $l_{mP,P}$ és la recta que passa pels punts $mP$ i $P$, i $v_{(m+1)P}$ és la recta vertical que passa pel punt $(m+1)P$ (recordeu que hem descrit aquestes funcions a l'Exemple 9.5).

Noteu com, efectivament, $div(f_{m+1,P}) = div(f_{m,P} \cdot \frac{l_{mP,P}}{v_{(m+1)P}})$:

$$\begin{aligned} div(f_{m+1,P}) &= (m+1)\langle P\rangle - \langle (m+1)P\rangle - m\langle \mathscr{O}\rangle \\ div(f_{m,P}) &= m\langle P\rangle - \langle mP\rangle - (m-1)\langle \mathscr{O}\rangle \\ div(l_{mP,P}) &= \langle mP\rangle + \langle P\rangle + \langle -(m+1)P\rangle - 3\langle \mathscr{O}\rangle \\ div(v_{(m+1)P}) &= \langle (m+1)P\rangle + \langle -(m+1)P\rangle - 2\langle \mathscr{O}\rangle \end{aligned}$$

i, fent servir les Equacions 9.3 i 9.4, veiem que:

$$\begin{aligned} &div(f_{m,P} \cdot \frac{l_{mP,P}}{v_{(m+1)P}}) = \\ &= div(f_{m,P}) + div(l_{mP,P}) - div(v_{(m+1)P}) = \\ &= \Big(m\langle P\rangle - \langle mP\rangle - (m-1)\langle \mathscr{O}\rangle\Big) + \Big(\langle mP\rangle + \langle P\rangle + \langle -(m+1)P\rangle - 3\langle \mathscr{O}\rangle\Big) - \\ &\quad - \Big(\langle (m+1)P\rangle + \langle -(m+1)P\rangle - 2\langle \mathscr{O}\rangle\Big) = \\ &= m\langle P\rangle + \langle P\rangle + \langle mP\rangle - \langle mP\rangle + \langle -(m+1)P\rangle - \langle -(m+1)P\rangle - \langle (m+1)P\rangle - \\ &\quad - (m-1)\langle \mathscr{O}\rangle - 3\langle \mathscr{O}\rangle + 2\langle \mathscr{O}\rangle = \\ &= (m+1)\langle P\rangle - \langle (m+1)P\rangle - m\langle \mathscr{O}\rangle = \\ &= div(f_{m+1,P}) \end{aligned}$$

El divisor d'una funció la determina excepte múltiples escalars diferents de zero, de manera que podrem construir funcions amb uns divisors concrets fent servir l'Equació 9.5.

**Exemple 9.9 Construcció de la funció $f_{r,P}$**

Procedim a construir la funció $f_{3,P}$ per a $P = (2,11) \in E/\mathbb{Z}_{23} : y^2 = x^3 - x$.

La funció $f_{3,P}$ es construeix iterativament a partir d'$f_{1,P}$, la funció constant amb divisor zero:

$$f_{1,P} = 1$$

Per a construir $f_{2,P}$ cal calcular la funció $l_{P,P}$ i la funció $v_{2P}$.

La funció $l_{P,P}$ és la recta tangent a la corba que passa pel punt $P$:

$$m = \frac{3x_1^2 + a}{2y_1} \mod p = \frac{3 \cdot 2^2 - 1}{2 \cdot 11} \mod 23 = 12$$

$$y = mx + b;\ 11 = 12 \cdot 2 + b;\ b = 10$$

$$l_{P,P} : y = 12x + 10$$

La funció $v_{2P}$ és la recta vertical que passa pel punt $2P$:

$$2P = 2(2,11) = (2,12)$$

$$v_{2P} : x = 2$$

Aleshores:

$$f_{2,P} = f_{1,P} \cdot \left(\frac{l_{P,P}}{v_{2P}}\right) = \frac{y - 12x - 10}{x - 2} = \frac{y + 11x + 13}{x + 21}$$

Per últim, es calcula $f_{3,P}$ a partir d'$f_{2,P}$:

$$f_{3,P} = f_{2,P} \cdot v_P = \frac{y + 11x + 13}{x + 21}(x - 2) = y + 11x + 13$$

## 9.3 Construcció explícita dels *pairings* de Weil i Tate

Arribats a aquest punt, ja estem en disposició de poder descriure els *pairings* de Weil i de Tate.

### 9.3.1 El *pairing* de Weil

Siguin $P, Q \in E/\mathbb{F}_{p^k}[r]$ dos punts de la $r$-torsió d'una corba el·líptica i $D_P, D_Q$ dos divisors de grau zero amb suports disjunts tals que:

$$D_P \sim \langle P \rangle - \langle \mathcal{O} \rangle$$

$$D_Q \sim \langle Q \rangle - \langle \mathcal{O} \rangle$$

Existeixen funcions $f$ i $g$ tals que:

$$div(f) = rD_P$$

$$div(g) = rD_Q$$

El ***pairing* de Weil** és una aplicació que rep un parell de punts de la $r$-torsió i retorna una arrel $r$-èsima de la unitat, definida com:

$$w_r(P,Q) = \frac{f(D_Q)}{g(D_P)}$$

| Arrels de la unitat | Una arrel de la unitat és un número que elevat a un enter positiu dona 1. Quan l'enter positiu és 2, parlem d'arrels quadrades; i quan és 3, d'arrels cúbiques. |
|---|---|

**Exemple 9.10 Càlcul del *pairing* de Weil**

A continuació calcularem el *pairing* de Weil per a dos punts concrets d'una corba el·líptica. Aquest exemple està basat en l'exemple 5.1.1 del manual *Pairings for beginners* de *Craig Costello*.

La corba $E/\mathbb{Z}_{23} : y^2 = x^3 - x$ té $\#E/\mathbb{Z}_{23} = 24$ elements.

El punt $P = (2,11)$ té ordre $r = 3$.

El grau d'immersió de la corba respecte a $r = 3$ és $k = 2$, ja que $3 \nmid 23 - 1$ però en canvi $3 \mid 23^2 - 1$.

Hi ha 9 punts a la 3-torsió:

$$E/(\mathbb{Z}_{23}/(z^2+1))[3] = [\mathscr{O}, (2,11), (2,12), (21,11z), (21,12z), (5z,2z+2), (5z,21z+21), (18z,2z+21), (18z,21z+2)]$$

Calcularem el *pairing* de Weil $w_3(P,Q)$ per a $P = (2,11)$ i $Q = (21,12z)$, dos punts que pertanyen a la 3-torsió.

En primer lloc, hem de trobar els divisors de grau zero $D_P$ i $D_Q$ amb suports disjunts, i equivalents a $\langle P\rangle - \langle\mathscr{O}\rangle$ i $\langle Q\rangle - \langle\mathscr{O}\rangle$, respectivament. Per fer-ho, seleccionem dos punts addicionals aleatoris de la corba sobre el cos estès, $R = (17z, 2z+21)$ i $S = (10z+18, 13z+13)$, i fixem els divisors $D_P$ i $D_Q$ com:

$$D_P = \langle P+R\rangle - \langle R\rangle$$
$$D_Q = \langle Q+S\rangle - \langle S\rangle$$

Noteu com efectivament els divisors tenen grau zero i suports disjunts:

$$deg(D_P) = deg(D_Q) = 1-1 = 0$$
$$sup(D_P) = \{P+R, R\} = \{(z+16, 18z+20), (17z, 2z+21)\}$$
$$sup(D_Q) = \{Q+S, S\} = \{(19z+22, 12z+10), (10z+18, 13z+13)\}$$
$$sup(D_P) \cap sup(D_Q) = \emptyset$$

A més, els divisors són efectivament equivalents a $\langle P\rangle - \langle\mathscr{O}\rangle$ i $\langle Q\rangle - \langle\mathscr{O}\rangle$. En efecte, el divisor $D'_P$ resultant de restar $\langle P\rangle - \langle\mathscr{O}\rangle$ a $D_P$ és un divisor principal:

$$D'_P = D_P - (\langle P\rangle - \langle\mathscr{O}\rangle) = \langle P+R\rangle - \langle R\rangle - \langle P\rangle + \langle\mathscr{O}\rangle$$
$$deg(D'_P) = 1-1-1+1 = 0$$
$$(P+R) - R - P + \mathscr{O} = \mathscr{O}$$

Els càlculs per a $D_Q$ són anàlegs.

En segon lloc, necessitem trobar les funcions $f$ i $g$ que tenen com a divisors $3D_P$ i $3D_Q$, respectivament. Podem trobar $f$ i $g$ com:

$$f = f_{3,P}\left(\frac{v_{P+R}}{l_{P,R}}\right)^3$$

$$g = f_{3,Q}\left(\frac{v_{Q+S}}{l_{Q,S}}\right)^3$$

on $l$ i $v$ són les funcions que descriuen la recta que passa per dos punts i la recta vertical que passa per un punt, respectivament.

Efectivament, el divisor de $f$ és $3D_P$ (els càlculs per a $g$ són equivalents i s'ometen per brevetat):

$$\begin{aligned} div(f) &= div(f_{3,P}) + 3(div(v_{P+R}) - div(l_{P,R})) = \\ &= \Big(3\langle P\rangle - 3\langle\mathscr{O}\rangle\Big) + 3\Big(\langle -(P+R)\rangle + \langle P+R\rangle - 2\langle\mathscr{O}\rangle - \langle P\rangle - \langle R\rangle - \langle -(P+R)\rangle + 3\langle\mathscr{O}\rangle\Big) = \\ &= 3\langle P+R\rangle - 3\langle R\rangle \end{aligned}$$

Procedim doncs a construir les funcions $f$ i $g$. Per a construir $f$, cal trobar les funcions $v_{P+R}$, $l_{P,R}$ i $f_{3,P}$:

La funció $v_{P+R}$ és la recta vertical que passa pel punt $P+R$:

$$P+R = (2,11) + (17z, 2z+21) = (z+16, 18z+20)$$

$$v_{P+R} : x = z+16$$

La funció $l_{P,R}$ és la recta que passa pels punts $P$ i $R$:

$$m = \frac{y_2 - y_1}{x_2 - x_1} \mod p = \frac{2z+21-11}{17z-2} = 6z+13$$

$$y = mx + c; c = y - mx = 11 - 2(6z+13) = 11z+8$$

$$l_{P,R} : y = (6z+13)x + (11z+8)$$

La funció $f_{3,P}$ es construeix iterativament a partir d'$f_{1,P}$, tal com hem vist a l'Exemple 9.9.

$$f_{3,P} = y + 11x + 13$$

Així doncs, la funció $f$ és:

$$f(x,y) = f_{3,P}\left(\frac{v_{P+R}}{l_{P,R}}\right)^3 = (y+11x+13)\cdot\left(\frac{x-z-16}{y-(6z+13)x-(11z+8)}\right)^3$$

De la mateixa manera, per a construir $g$ cal trobar les funcions $v_{Q+S}$, $l_{Q,S}$ i $f_{3,Q}$. A continuació es proporcionen aquestes funcions, i es deixa com a exercici per al lector els càlculs individuals per a trobar-les:

**Exercici 9.3** Trobeu les funcions $v_{Q+S}$, $l_{Q,S}$ i $f_{3,Q}$.

$$v_{Q+S} : x = 19z+22$$

$$l_{Q,S} : y = (3z+1)x + (18z+2)$$

$$f_{3,Q} : y = -11zx - 10z$$

De manera que la funció $g$ és:

$$g(x,y) = f_{3,Q}\left(\frac{v_{Q+S}}{l_{Q,S}}\right)^3 = (y+11zx+10z)\cdot\left(\frac{x-19z-22}{y-(3z+1)x-(18z+2)}\right)^3$$

En tercer lloc i ja per acabar, procedim a calcular el *pairing* de Weil:

$$w_3(P,Q) = \frac{f(D_Q)}{g(D_P)} = \frac{f(Q+S)g(R)}{f(S)g(P+R)} = \frac{(7z+22)(21z+22)}{(5z+4)(15z+1)} = 15z+11$$

Noteu com, efectivament, $15z+11$ és una arrel 3-èsima de la unitat, ja que $(15z+11)^3 = 1$.

A partir de l'exemple anterior, podem veure ara alguns dels efectes de la bilinialitat del *pairing* de Weil:

**Exemple 9.11 Bilinialitat en el *pairing* de Weil**

A l'exemple anterior hem calculat el *pairing* de Weil per a $P = (2,11)$ i $Q = (21,12z)$. A continuació veurem exemples de bilinialitat en el *pairing* de Weil per a aquests punts concrets.

Els punts resultants de doblar $P$ i $Q$ són $2P = (2,12)$ i $2Q = (21,11z)$.

Si calculem el *pairing* de Weil de $2P$ i $Q$ o bé el de $P$ i $2Q$, veurem que són iguals, i també que coincideixen amb el quadrat del de $P$ i $Q$:

$$w_3(2P,Q) = w_3(P,2Q) = 8z+11 = (15z+11)^2 = w_3(P,Q)^2$$

### 9.3.2 El *pairing* de Tate

En la definició bàsica del *pairing* de Tate, el resultat del *pairing* per un parell de punts concret no és únic. Com que en aplicacions criptogràfiques aquesta característica és problemàtica, habitualment s'utilitza el *pairing* de Tate reduït, que no és res més que el *pairing* de Tate elevat a $(q^k-1)/r$. Així, s'aconsegueix que el resultat sigui una arrel $r$-èsima de la unitat, i que per cada parell de punts el valor del *pairing* sigui únic. En aquest text descrivim doncs directament el *pairing* de Tate reduït.

Siguin $P,Q \in E/\mathbb{F}_{q^k}[r]$ dos punts de la $r$-torsió d'una corba el·líptica. Existeix una funció $f$ tal que:

$$div(f) = r\langle P\rangle - r\langle O\rangle$$

Sigui $D_Q$ un divisor de grau zero amb suport disjunt de $div(f)$ i tal que:

$$D_Q \sim \langle Q\rangle - \langle \mathcal{O}\rangle$$

El ***pairing* de Tate reduït** és una aplicació que rep un parell de punts de la $r$-torsió i retorna una arrel $r$-èsima de la unitat, definida com:

$$t_r(P,Q) = f(D_Q)^{(q^k-1)/r}$$

A diferència del *pairing* de Weil, en el *pairing* de Tate només cal que un dels punts pertanyi a la $r$-torsió (el punt $P$ que descriu el divisor d'$f$), i el segon punt pot no ser-hi (però ha de complir unes propietats concretes que no són trivials de definir). Com que els punts de la $r$-torsió les compleixen, per simplicitat en aquest document seleccionem sempre punts que hi pertanyin.

**Exemple 9.12 Càlcul del *pairing* de Tate**

A continuació calcularem el *pairing* de Tate reduït per als mateixos dos punts que en l'exemple anterior (Exemple 9.10).

Sigui $E/\mathbb{Z}_{23} : y^2 = x^3 - x$ la corba el·líptica, amb $P = (2,11), Q = (21,12z) \in E/(\mathbb{Z}_{23}/(z^2+1))[3]$. El grau d'immersió de la corba respecte a $r = 3$ és $k = 2$.

En primer lloc, trobem una funció $f$ amb divisor $3\langle P\rangle - 3\langle O\rangle$. Tal com hem calculat a l'exemple anterior, la funció $f_{3,P} : y + 11x + 13$ té aquest divisor.

En segon lloc, trobem un divisor $D_Q$ de grau zero amb suport disjunt al divisor d'$f$ i equivalent a $\langle Q\rangle - \langle \mathscr{O}\rangle$. Per fer-ho, podem fer servir la mateixa estratègia que a l'exemple del *pairing* de Weil: seleccionem un punt $S$ aleatori i fixem:

$$D_Q = \langle Q + S\rangle - \langle S\rangle$$

Per al punt $S = (10z+18, 13z+13)$, tenim doncs que:

$$D_Q = \langle (19z+22, 12z+10)\rangle - \langle (10z+18, 13z+13)\rangle$$

Noteu que en aquest cas no ens cal calcular la funció que té per divisor $D_Q$.

Finalment, calculem el *pairing* de Tate reduït:

$$t_r(P,Q) = f(D_Q)^{(p^k-1)/r} = \left(\frac{f(Q+S)}{f(S)}\right)^{(q^k-1)/r} = \left(\frac{f(19z+22,12z+10)}{f(10z+18,13z+13)}\right)^{(23^2-1)/3} =$$
$$= \left(\frac{14z+12}{8z+17}\right)^{176} = 15z+11$$

De nou, podem comprovar com el resultat del *pairing* és una arrel 3-èsima de la unitat: $(15z+11)^3 = 1$.

**Exemple 9.13 Bilinialitat en el *pairing* de Tate**

De la mateixa manera que amb el *pairing* de Weil, podem veure també un exemple concret de les propietats de bilinealitat del *pairing* de Tate a partir dels valors de l'exemple anterior.

Recordem que $P = (2,11)$, $Q = (21,12z)$, $2P = (2,12)$ i $2Q = (21,11z)$.

Si calculem el *pairing* de Tate de $2P$ i $Q$ o bé el de $P$ i $2Q$, veurem que són iguals, i també que coincideixen amb el quadrat del de $P$ i $Q$:

$$t_r(2P,Q) = t_r(P,2Q) = 8z+11 = (15z+11)^2 = t_r(P,Q)^2$$

## 9.4 Algorismes criptogràfics basats en *pairings*

En aquesta secció es descriuen algorismes criptogràfics que fan servir *pairings*. En primer lloc, veurem l'esquema de signatura BLS, que permet fer signatures curtes i, a més, permet agregar-les. A continuació,

descriurem la criptografia basada en la identitat i explicarem un dels algorismes d'aquesta família, l'algorisme de xifratge de Boneh-Franklin.

### 9.4.1 L'esquema de signatura BLS

L'esquema de signatura digital BLS (anomenat així per les inicials dels cognoms dels seus creadors, Dan Boneh, Ben Lynn i Hovav Shacham) va ser proposat l'any 2001. L'esquema fa servir un *pairing* bilineal per a la verificació de signatures.

La característica principal d'aquest esquema de signatura digital és que produeix signatures *curtes*: la mida de la signatura digital és la meitat de la tindria una signatura DSA amb el mateix nivell de seguretat. Això fa que l'esquema sigui idoni per a situacions amb poc ample de banda o quan és necessari que un humà transcrigui la signatura digital manualment.

L'esquema de signatura BLS fa servir un *pairing* i una funció hash. Sigui $e : \mathbb{G}_0 \times \mathbb{G}_1 \to \mathbb{G}_T$ un *pairing* on $\mathbb{G}_0, \mathbb{G}_1$ i $\mathbb{G}_T$ són grups cíclics d'ordre primer $q$ amb $G_0 \in \mathbb{G}_0$ i $G_1 \in \mathbb{G}_1$ elements generadors dels grups. Sigui $H$ una funció hash que relaciona els missatges de l'espai de missatges a elements de $\mathbb{G}_0$.

L'algorisme de **generació de claus BLS** consta dels passos següents:

1. Es tria un enter aleatori $\alpha \in_R \mathbb{Z}_q$.
2. Es calcula $u = \alpha \cdot G_1 \in \mathbb{G}_1$.
3. La clau pública és $k_{pub} = u$, mentre que la clau privada és $k_{priv} = \alpha$.

L'algorisme de generació de claus és doncs anàleg al de l'ECDSA: es tria un enter aleatori que serà la clau privada i es multiplica aquest enter pel generador d'un grup cíclic, que és un paràmetre públic del sistema. La clau pública és un element de $\mathbb{G}_1$, és a dir, un punt d'una corba el·líptica.

Les claus generades per l'algorisme anterior es poden fer servir per a generar i validar signatures digitals fent servir els algorismes següents:

A partir d'un missatge en clar $m$, la clau privada de l'emissor $k_{priv} = \alpha$, i els paràmetres de domini, es calcula la **signatura digital BLS** del missatge:

1. Es calcula $\sigma = \alpha \cdot H(m) \in \mathbb{G}_0$
2. La signatura és el valor $\sigma$.

L'algorisme de signatura requereix de l'ús d'una funció hash que s'aplica al missatge abans de signar-lo, i que el converteix en un element del grup $\mathbb{G}_0$, és a dir, en un punt de la corba el·líptica. Una vegada es té la representació del missatge $m$ en el grup $\mathbb{G}_0$, només cal calcular una multiplicació escalar entre el punt de la corba que representa el missatge i la clau privada. Convé destacar que la signatura és doncs un punt d'una corba el·líptica (un element de $\mathbb{G}_0$), de manera que si la representació dels punts és curta, la signatura també ho serà.

Fins aquí només s'ha fet ús de corbes el·líptiques però encara no s'ha introduït l'ús del *pairing*. L'algorisme de verificació de la signatura és precisament el punt de l'esquema que requereix de l'ús de *pairings*:

A partir d'un missatge en clar $m$, la clau pública $k_{pub}$, els paràmetres de domini i una signatura del missatge $\sigma$, els passos següents permeten **verificar una signatura BLS**:

1. Es verifica que $e(H(m),u) = e(\sigma, G_1)$.
2. Si la igualtat es compleix, aleshores la signatura és vàlida i la verificació finalitza correctament. En cas contrari, la signatura es considera invàlida i la verificació fracassa.

Convé notar com una signatura digital correcta serà donada per vàlida per l'algorisme de verificació, ja que per les propietats de bilinealitat del *pairing*:

$$e(H(m),u) = e(H(m),\alpha G_1) = e(H(m),G_1)^{\alpha} = e(\alpha H(m),G_1) = e(\sigma,G_1)$$

L'esquema de signatura BLS és determinista, ja que donats uns paràmetres de domini, una clau privada i un missatge en clar, la signatura que es produeix és única. Això el diferencia de la variant clàssica de l'algorisme de signatura ECDSA, que és probabilístic.

**Exemple 9.14 Exemple de signatura i validació amb BLS**

Sigui $E/\mathbb{Z}_{43} : y^2 = x^3 + 7x$ una corba el·líptica d'ordre 44. Farem servir com a grups cíclics $\mathbb{G}_0$ i $\mathbb{G}_1$ dos subgrups cíclics de la 11-torsió. El grau d'immersió de la corba respecte a $r = 11$ és $k = 2$. L'exemple utilitzarà com a *pairing* $e$ el *pairing* de Weil ($w_{11}$).

Sigui $G_0 = (4,7)$ el generador del subgrup cíclic $\mathbb{G}_0$ i $G_1 = (2,8z)$ el generador del subgrup cíclic $\mathbb{G}_1$ (tots dos d'ordre 11). $\mathbb{G}_0$ es troba en el cos base ($\mathbb{G}_0 < E/\mathbb{Z}_{43}$), mentre que $\mathbb{G}_1$ es troba al cos estès $E/(\mathbb{F}_{43^2}/z^2+1)$.

Abans de signar, l'usuari ha de disposar d'un parell de claus. Per aconseguir-les, l'usuari executa l'algorisme de generació de claus:

1. Tria un enter aleatori $\alpha = 5 \in_R \mathbb{Z}_{43}$.
2. Calcula $u = \alpha \cdot G_1 = 5(2,8z) = (39,7z) \in \mathbb{G}_1$.
3. La clau pública és $k_{pub} = (39,7z)$, mentre que la clau privada és $k_{priv} = 5$.

Ara, l'usuari pot signar executant l'algorisme de generació de la signatura. Suposem que la representació del missatge $m$ al subgrup $\mathbb{G}_0$ és $H(m) = (41,35)$. Efectivament, $H(m) \in \mathbb{G}_0$ ja que $H(m) = 2G_0$. Recordeu que ja hem vist a la secció 8.4.3 com crear funcions hash que retornin punts d'una corba concreta.

1. Es calcula $\sigma = \alpha \cdot H(m) = 5(41,35) = (4,36) \in \mathbb{G}_0$
2. La signatura és el valor $\sigma = (4,36)$.

Un receptor pot verificar la signatura a partir del missatge $m$, la signatura $\sigma$, i la clau pública $k_{pub}$ (i coneixent els paràmetres de domini que són públics):

1. El receptor verifica que $e(H(m),u) = e(\sigma,G_1)$.
   (a) $e(H(m),u) = w_{11}((41,35),(39,7z)) = 34z+7$.
   (b) $e(\sigma,G_1) = w_{11}((4,36),(2,8z)) = 34z+7$.
2. Com que la igualtat es compleix, la signatura és vàlida.

**Exercici 9.4** Suposeu que es fa servir l'esquema de signatura BLS amb la construcció ingènua de la funció hash $H$ següent:

$$H(m) = H'(m) \cdot G_1$$

on $H'(m) = SHA\text{-}1(m) \mod q$. Expliqueu perquè aquesta construcció no és segura.

Més enllà de produir signatures curtes, l'esquema de signatura BLS té algunes propietats addicionals que el fan especialment interessant. El BLS permet agregació de signatures i esquemes de signatures llindar. A continuació descriurem com construir un esquema de signatures agregables en base a l'algorisme de signatura BLS.

### Agregació de signatures

Els esquemes de signatura digital que permeten **agregació de signatures** es caracteritzen per permetre comprimir diverses signatures (sobre diferents missatges i amb diferents claus) en una sola signatura agregada, que es pot fer servir per verificar totes les signatures de cop. Aquesta signatura agregada té una longitud similar a la d'una signatura individual, independentment del nombre de signatures que comprimeixi.

De la mateixa manera que un esquema de signatura queda definit per tres algorismes (generació de claus, signatura i verificació), els esquemes que permeten agregació de signatures incorporen dos algorismes addicionals: l'agregació de signatures i la verificació d'una signatura agregada. L'agregació de signatures rep un conjunt de signatures (i, en alguns esquemes, les claus públiques associades) i genera una única signatura agregada. Com que l'agregació de signatures no requereix de claus privades ni de la interacció dels signants, qualsevol persona (amb coneixement de les signatures i les claus públiques) pot executar l'algorisme i generar una signatura agregada. Això implica que l'agregació de signatures es pot fer amb posterioritat a la creació de les signatures. La verificació de signatures rep una signatura agregada i el conjunt de missatges que s'han signat (i, en alguns esquemes, les claus públiques associades), i valida que totes les signatures que resumeix la signatura agregada siguin correctes.

Els esquemes de signatura digital que permeten agregació de signatures són útils en diversos escenaris. Per exemple, en la verificació d'una cadena de certificats digitals, és habitual haver de validar diverses signatures, des del certificat a comprovar fins al certificat arrel de la CA en el qual es confia. Un altre escenari on l'agregació de signatures és especialment interessant és en criptomonedes basades en cadena de blocs, on té diverses aplicacions. D'una banda, en les transaccions amb múltiples entrades, permetria agregar les signatures de cada entrada en una sola signatura, cosa que redueix la mida d'aquestes transaccions. La mida de les transaccions és crítica en sistemes *blockchain*, ja que és un dels grans limitadors de l'escalabilitat del sistema. D'altra banda, l'agregació de signatures també permetria implementar sortides multisignatura de manera eficient. Les sortides multisignatura són sortides de transaccions que requereixen més d'una signatura per a ser gastades. Aquestes sortides especifiquen un conjunt de claus públiques i un llindar de signatures mínim necessàries per a autoritzar el pagament. Així, per tal de gastar aquestes sortides es requereix habitualment d'un conjunt de signatures. Si es disposa d'un esquema amb agregació de signatures, aquest conjunt de signatures a proporcionar es pot resumir en una única signatura, oferint de nou millores en la longitud de les transaccions. Addicionalment, alguns esquemes d'agregació de signatures també permeten agregar les claus públiques, cosa que redueix encara més la mida de les transaccions i ofereix privadesa afegida.

**BLS i Ethereum**

Les primeres versions de les criptomonedes Bitcoin i Ethereum feien servir ECDSA. La nova versió d'Ethereum (Eth2) que es troba actualment en desplegament (2021) incorpora signatures BLS, amb l'objectiu d'accelerar la verificació de signatures. Bitcoin incorpora des de novembre de 2021 l'ús de signatures Schnorr, que també permeten agregació.

A continuació es descriu l'algorisme d'agregació de signatures BLS ingenu, que permet agregar signatures però no és segur. Més endavant es descriu el problema de seguretat d'aquesta versió de l'algorisme i una

modificació que permet solucionar-lo.

A partir d'un conjunt d'$n$ claus públiques $K^{pub} = \{k_1^{pub}, \cdots, k_n^{pub}\}$ i d'un conjunt d'$n$ signatures $\Sigma = \{\sigma_1, \cdots, \sigma_n\}$ es **calcula la signatura agregada** $\sigma_{ag}$:

1. Es calcula la signatura agregada $\sigma_{ag} = \sigma_1 + \cdots + \sigma_n$
2. La signatura agregada és el valor $\sigma_{ag} \in \mathbb{G}_0$.

L'agregació de signatures consisteix doncs en la suma de punts de la corba $\mathbb{G}_0$, que conformen les signatures individuals, i la signatura agregada és també un punt de la mateixa corba. Noteu com el procés d'agregació de signatures no requereix ni dels missatges signats ni de les claus privades que han generat les signatures.

A partir d'un conjunt d'$n$ claus públiques $K^{pub} = \{k_1^{pub}, \cdots, k_n^{pub}\}$, d'un conjunt d'$n$ missatges $M = \{m_1, \cdots, m_n\}$ i d'una signatura agregada $\sigma_{ag}$, els passos següents permeten **verificar la signatura agregada**:

1. Es comprova si $e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(m_1), k_1^{pub}) \cdot \cdots \cdot e(H(m_n), k_n^{pub})$
2. La signatura agregada és vàlida (i, per tant, totes les signatures individuals que resumeix es consideren vàlides) si la comprovació del pas anterior es fa amb èxit. En cas contrari, es rebutja la signatura agregada.

L'algorisme de verificació dona per vàlida una signatura agregada que comprimeix un conjunt de signatures individuals vàlides per les propietats del *pairing*:

$$\begin{aligned} e(\sigma_{ag}, G_1) &= e(\sigma_1 + \cdots + \sigma_n, G_1) = \\ &= e(\sigma_1, G_1) \cdot \cdots \cdot e(\sigma_n, G_1) = \\ &= e(\alpha_1 H(m_1), G_1) \cdot \cdots \cdot e(\alpha_n H(m_n), G_1) = \\ &= e(H(m_1), \alpha_1 G_1) \cdot \cdots \cdot e(H(m_n), \alpha_n G_1) = \\ &= e(H(m_1), k_1^{pub}) \cdot \cdots \cdot e(H(m_n), k_n^{pub}) \end{aligned}$$

La verificació de signatures agregades es pot calcular de manera especialment eficient quan el missatge signat és el mateix per a totes les signatures, és a dir, quan $m = m_1 = m_2 = \cdots = m_n$. En aquests casos, l'equació de verificació es pot simplificar:

$$e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(m_1), k_1^{pub}) \cdot \cdots \cdot e(H(m_n), k_n^{pub})$$

$$e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(m), k_1^{pub}) \cdot \cdots \cdot e(H(m), k_n^{pub})$$

$$e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(m), k_1^{pub} + \cdots + k_n^{pub})$$

de manera que es passa de necessitar calcular $n+1$ *pairings* a calcular-ne només 2.

**Exercici 9.5** Aquest exercici fa servir els mateixos paràmetres de domini que l'Exemple 9.14. Sigui $E/\mathbb{Z}_{43} : y^2 = x^3 + 7x$ una corba el·líptica d'ordre 44; $\mathbb{G}_0$ i $\mathbb{G}_1$ dos subgrups cíclics de la 11-torsió generats per $G_0 = (4,7)$ i $G_1 = (2,8z)$, respectivament; i $e$ el *pairing* de Weil $(w_r)$.

També aprofitem el parell de claus i la signatura generades a l'Exemple 9.14:

$$k_1^{priv} = 5,\ k_1^{pub} = (39,7z)$$

$$m = (41,35),\ \sigma_1 = (4,36)$$

1. Sigui $k_2^{priv} = 7$ la clau privada d'un segon usuari. Genereu la seva clau pública.
2. Genereu una signatura $\sigma_2$ de l'usuari 2 per al missatge $m = (41,35)$.
3. Agregueu les dues signatures ($\sigma_1$ i $\sigma_2$) en una sola signatura agregada $\sigma_{ag}$.
4. Verifiqueu la signatura agregada $\sigma_{ag}$ fent servir l'algorisme de verificació de signatura agregada.
5. Verifiqueu de nou la signatura agregada $\sigma_{ag}$, aprofitant que els dos missatges signats són idèntics.

Per a facilitar la resolució de l'exercici, a continuació es proporcionen alguns valors precalculats:

$$
\begin{array}{ll}
G_0 = (4,7) & G_1 = (2,8z) \\
2G_0 = (41,35) & 2G_1 = (26,42z) \\
3G_0 = (15,30) & 3G_1 = (33,9z) \\
4G_0 = (17,1) & 4G_1 = (28,30z) \\
5G_0 = (10,9) & 5G_1 = (39,7z) \\
6G_0 = (10,34) & 6G_1 = (39,36z) \\
7G_0 = (17,42) & 7G_1 = (28,13z) \\
8G_0 = (15,13) & 8G_1 = (33,34z) \\
9G_0 = (41,8) & 9G_1 = (26,a) \\
10G_0 = (4,36) & 10G_1 = (2,35z) \\
11G_0 = \mathscr{O} & 11G_1 = \mathscr{O}
\end{array}
\qquad
\begin{array}{l}
w_r((41,35),(2,8z)) = 40z+11 \\
w_r((41,35),(39,7z)) = 34z+7 \\
w_r((41,35),(28,13z)) = 35z+18
\end{array}
$$

L'algorisme que acabem de presentar és però vulnerable a atacs de clau pública múrria (en anglès, es coneixen amb el nom de *rogue public key attacks*). En aquests atacs un atacant és capaç de generar una signatura agregada vàlida que inclou una signatura d'un missatge $m$ per part d'una víctima $V$ sense que la víctima hagi proporcionat tal signatura.

L'atac de clau pública múrria fa servir la clau pública d'una víctima $k_{pubV} = u_V$ i genera una signatura agregada $\sigma_{ag}$ vàlida que inclou una signatura de la víctima del missatge $m$. L'atac consta dels passos següents:

1. L'atacant selecciona un enter aleatori $\alpha \in_R \mathbb{Z}_q$.
2. L'atacant calcula la clau pública auxiliar $k_{pubA} = u_A = \alpha \cdot G_1 \in \mathbb{G}_1$.
3. L'atacant calcula la clau pública múrria $k_{pubM} = u_A - u_V \in \mathbb{G}_1$.
4. L'atacant calcula la signatura agregada $\sigma_{ag} = \alpha \cdot H(m)$.
5. L'atacant presenta la signatura agregada $\sigma_{ag}$ per al conjunt de missatges $M = \{m,m\}$ amb claus públiques $K^{pub} = \{k_{pubV}, k_{pubM}\}$

Noteu que l'adversari fa servir la clau pública $k_{pubM}$ per a l'atac, però que en desconeix la clau privada corresponent (l'adversari ha creat aquesta clau pública combinant la clau pública de la víctima i la clau pública auxiliar que ha generat i per a la qual sí que en coneix la clau privada).

**Exercici 9.6** Demostreu que l'atac de clau pública múrria aconsegueix generar una signatura agregada vàlida en la versió bàsica del BLS fent servir les propietats de bilinealitat del *pairing*.

Aquest atac trenca la seguretat de l'esquema de signatura agregada ingènua que hem descrit i motiva la creació de variants que en siguin resistents. Es coneixen diferents variants de l'esquema segures i, a continuació, en descriurem una d'elles.

L'algorisme d'agregació de signatures BLS segur parteix d'una variant modificada de l'algorisme de signatures BLS:

A partir d'un missatge en clar $m$, el parell de claus de l'emissor ($k_{priv} = \alpha$ i $k_{pub} = u = \alpha \cdot G_1 \in \mathbb{G}_1$), i els paràmetres de domini, es calcula la **signatura digital BLS modificada** del missatge:

1. Es calcula $\sigma = \alpha \cdot H(k_{pub}, m)$
2. La signatura és el valor $\sigma$.

És a dir, la signatura es fa no només sobre el missatge $m$ sinó també sobre la clau pública del signant $k_{pub}$, impedint així l'atac de clau pública múrria que hem vist anteriorment. Això requereix de la utilització d'una funció hash $H$ que rebi dos valors (el primer de $\mathbb{G}_1$ i el segon de l'espai de missatges) i retorni un valor de $\mathbb{G}_0$.

L'algorisme d'agregació de signatures es manté tal com l'hem definit anteriorment. L'algorisme de verificació de signatures agregades es modifica lleugerament de manera anàloga al procés de signatura:

A partir d'un conjunt d'$n$ claus públiques $K^{pub} = \{k_1^{pub}, \cdots, k_n^{pub}\}$, d'un conjunt d'$n$ missatges $M = \{m_1, \cdots, m_n\}$ i d'una signatura agregada $\sigma_{ag}$, els passos següents permeten **verificar la signatura agregada**:

1. Es comprova si $e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(k_1^{pub}, m_1), k_1^{pub}) \cdots e(H(k_n^{pub}, m_n), k_n^{pub})$
2. La signatura agregada és vàlida si la comprovació del pas anterior es fa amb èxit. En cas contrari, es rebutja la signatura agregada.

**Demostració** La demostració de perquè aquesta variant és segura queda fora de l'abast d'aquest text. El lector interessat pot consultar l'article original *Compact Multi-Signatures for Smaller Blockchains* de Dan Boneh, Manu Drijvers i Gregory Neven per a llegir-ne els detalls.

### 9.4.2 Criptografia basada en la identitat

Els *pairings* permeten també construir esquemes de xifratge amb propietats addicionals als sistemes de xifratge tradicionals.

En els esquemes de xifratge de clau pública tradicionals, per tal d'enviar un missatge xifrat a una persona caldrà que en coneguem la seva clau pública. Ja hem vist com aquesta associació entre una clau pública i la identitat d'un usuari es fa habitualment amb un certificat digital. Per tant, per aconseguir la clau pública del receptor del missatge, l'emissor pot demanar-li el seu certificat digital o bé pot descarregar-lo d'un repositori públic de certificats. En qualsevol dels dos casos, l'emissor necessita aconseguir i validar la clau pública del receptor abans de poder iniciar el procés de xifratge del missatge. El xifratge basat en la identitat, proposat per Adi Shamir el 1984, permet evitar aquest procés previ i fer servir la identitat del receptor directament com a la seva clau pública.

Tot i que la idea del xifratge basat en la identitat va ser proposada el 1984, en aquell moment només es coneixia un esquema de signatura basat en la identitat (proposat pel mateix Shamir). No va ser fins al 2001 quan Dan Boneh i Matthew K. Franklin van proposar el primer esquema de xifratge basat en la identitat, que utilitzava el pairing de Weil sobre corbes el·líptiques.

El **xifratge basat en la identitat** (IBE, de l'anglès, *Identity Based Encryption*) és un tipus de xifratge de clau pública en què la clau pública d'un usuari es deriva directament d'una informació única sobre la identitat d'aquest usuari.

En els esquemes de xifratge basat en la identitat, qualsevol cadena de caràcters que identifiqui a l'usuari pot ser utilitzada per a calcular la clau pública. Cal, però, que aquest identificador sigui únic entre tots els usuaris de l'esquema. Alguns exemples d'identificadors habituals són el correu electrònic, el número de telèfon o el nom de domini.

Els esquemes de xifratge basats en la identitat requereixen d'una entitat de confiança, que és l'encarregada de generar les claus dels usuaris. L'entitat de confiança disposa d'un parell de claus mestra. La clau pública mestra és coneguda per totes les entitats del sistema i es fa servir en el procés de xifratge, mentre que la clau privada mestra és secreta (només coneguda per l'entitat de confiança). Aquesta clau privada mestra es fa servir per a derivar les claus privades dels usuaris.

Per tant, els usuaris necessiten interactuar amb l'entitat de confiança per tal d'obtenir les claus privades associades als seus identificadors, de manera que cal que l'entitat de confiança pugui autenticar els usuaris (per assegurar que obtenen la clau privada d'un identificador propi) i disposi d'un canal confidencial amb els usuaris (per transmetre'ls la clau privada).

Així, els usuaris d'un esquema IBE es poden intercanviar missatges xifrats sense necessitat d'haver tingut cap contacte previ entre ells per tal d'intercanviar-se les claus públiques però, en canvi, sí que caldrà que els usuaris puguin comunicar-se amb l'entitat de confiança.

El flux d'informació a l'hora de xifrar no és l'única diferència entre els esquemes IBE i els esquemes de clau pública tradicionals. Una altra diferència és la necessitat d'existència de les claus prèvia al procés de xifratge. En un esquema tradicional, l'usuari ha de generar un parell de claus abans de poder rebre un missatge xifrat. En canvi, amb IBE, l'usuari pot rebre un missatge xifrat per al qual encara no se n'ha generat la clau privada.

En una infraestructura de clau pública tradicional existeixen mecanismes de revocació dels certificats digitals, que permeten gestionar situacions com ara el compromís de les claus privades dels usuaris. En els esquemes basats en IBE, les claus públiques no es poden revocar, ja que no hi ha cap manera de comunicar a l'emissor que una clau ha estat revocada: l'emissor fa servir la identitat de l'usuari i la clau pública mestra per a xifrar un missatge, sense comunicar-se amb cap entitat que pugui informar-lo de la possible revocació d'una clau. Per a solucionar aquesta limitació, els esquemes IBE acostumen a combinar l'identificador de l'usuari amb una cadena que representi el període de temps en què es considera vàlida la clau. D'aquesta manera, una mateixa clau només és vàlida durant un període de temps concret, cosa que limita les possibles conseqüències d'una pèrdua o compromís.

**Exemple 9.15 Identificadors per a IBE**

Us sistema IBE per a correu electrònic pot fer servir com a identificador d'usuari el correu de l'usuari concatenat amb la data, de manera que les claus tindrien una vigència d'un dia.

Així, per exemple, l'identificador `2021-09-30:info@uoc.edu` seria la clau pública associada a l'adreça `info@uoc.edu` vàlida durant el dia 30 de setembre de 2021.

Fent servir aquest mateix format, es poden enviar correus que només es poden desxifrar en el futur, per exemple, xifrant un correu per a l'identificador `2221-09-30:info@uoc.edu`.

Una altra diferència notable dels esquemes IBE és que incorporen implícitament un sistema de recuperació de claus. Com que l'entitat de confiança pot calcular totes les claus privades de tots els usuaris, l'entitat de confiança pot recuperar qualsevol clau del sistema. A més, pot fer-ho sense necessitat de guardar claus individuals, ja que les claus privades dels usuaris es deriven directament de la clau privada mestra.

Els esquemes IBE consten dels tres algorismes habituals en els esquemes de xifratge (generació de claus, xifratge i desxifratge) més un algorisme addicional d'inicialització, que consisteix en la generació del parell de claus de l'entitat de confiança. Els dos algorismes de generació de claus (el de l'entitat de confiança i el dels usuaris del sistema) són executats per l'entitat de confiança. Els algorismes de xifratge i desxifratge són executats pels usuaris de l'esquema (emissors i receptors de missatges), com en els esquemes de xifratge

tradicionals.

### L'esquema bàsic de Boneh-Franklin

A continuació es presenta una de les construccions de xifratge basat en la identitat proposades per Dan Boneh i Matthew K. Franklin. La construcció fa servir *pairings* sobre corbes el·líptiques.

Sigui $e : \mathbb{G}_0 \times \mathbb{G}_1 \to \mathbb{G}_T$ un *pairing* on $\mathbb{G}_0, \mathbb{G}_1$ i $\mathbb{G}_T$ són grups cíclics d'ordre primer $q$ amb $G_0 \in \mathbb{G}_0$ i $G_1 \in \mathbb{G}_1$ elements generadors dels grups.

L'esquema d'IBE fa ús d'un criptosistema de clau simètrica (tal que $m = D_k(E_k(m))$) i de dues funcions hash. Sigui $H_0$ una funció hash que relaciona els missatges de l'espai d'identificadors a elements de $\mathbb{G}_0$ i $H_1$ una funció hash que relaciona parells de $\mathbb{G}_1 \times \mathbb{G}_T$ amb les claus del criptosistema simètric.

L'algorisme d'inicialització és executat per l'entitat de confiança per tal de generar el parell de claus mestre:

L'algorisme d'inicialització consisteix en la **generació del parell de claus mestre** de l'entitat de confiança $\mathfrak{C}$ i consta dels passos següents:

1. Es tria un enter aleatori $\alpha \in_R \mathbb{Z}_q$.
2. Es calcula $u_1 = \alpha \cdot G_1 \in \mathbb{G}_1$.
3. La clau pública mestra és $k^{\mathfrak{C}}_{pub} = u_1$, mentre que la clau privada mestra és $k^{\mathfrak{C}}_{priv} = \alpha$.

L'algorisme de generació de claus d'un usuari és executat també per l'entitat de confiança, en el moment en què l'usuari li sol·licita la clau privada associada al seu identificador:

L'algorisme de **generació de claus** d'un usuari rep la identitat *id* de l'usuari i la clau privada mestra $k^{\mathfrak{C}}_{priv} = \alpha$ i executa els passos següents:

1. L'entitat de confiança $\mathfrak{C}$ calcula $sk_{id} = \alpha \cdot H_0(id) \in \mathbb{G}_0$.
2. La clau pública de l'usuari és $k_{pub} = id$, mentre que la clau privada és $k_{priv} = sk_{id}$.

L'algorisme de xifratge consisteix en la derivació d'una clau simètrica $k$ a partir de l'identificador del receptor i la clau pública mestra. Aquesta clau simètrica es fa servir per a xifrar el missatge amb un algorisme de xifratge simètric.

A partir d'un missatge en clar $m$, la identitat del receptor *id*, i la clau pública mestra $k^{\mathfrak{C}}_{pub} = u_1$, es calcula el **missatge xifrat**:

1. Es tria un enter aleatori $\beta \in_R \mathbb{Z}_q$.
2. Es calcula $w_1 = \beta \cdot G_1 \in \mathbb{G}_1$.
3. Es calcula el *pairing* $z = e(H_0(id), \beta u_1) \in \mathbb{G}_T$.
4. Es calcula la clau simètrica $k = H_1(w_1, z)$.
5. Es calcula el missatge xifrat $c = E_k(m)$.
6. La sortida és la tupla $(w_1, c)$.

Noteu com l'usuari pot xifrar sense obtenir la clau pública de l'usuari, ja que aquesta és directament l'identificador.

L'algorisme de desxifratge procedeix a derivar la clau simètrica $k$ amb què s'ha xifrat el missatge, a partir de la informació que rep de l'emissor i la clau privada de l'usuari (que ha obtingut de l'entitat de confiança).

A partir d'un missatge xifrat $(w_1, c)$ i la clau secreta d'un usuari $k_{priv} = sk_{id}$, es calcula el **missatge desxifrat**:

1. Es calcula el *pairing* $z = e(sk_{id}, w_1) \in \mathbb{G}_T$.
2. Es calcula la clau simètrica $k = H_1(w_1, z)$.
3. Es calcula el missatge en clar $m = D_k(c)$.
4. La sortida és el missatge en clar $m$.

L'algorisme serà correcte si la clau simètrica que es fa servir al desxifrar és exactament la mateixa que s'utilitza al xifrar. La clau simètrica $k$ es deriva dels valors $w_1$ i $z$, i el valor $w_1$ es transmet com a part del text xifrat. Per tant, cal comprovar que les $z$ que es calculen en els algorismes de xifratge i desxifratge són les mateixes. En efecte, per les propietats del *pairing*, podem veure com els valors $z$ fets servir pels dos algorismes coincideixen:

$$\begin{aligned} e(sk_{id}, w_1) &= e(\alpha \cdot H_0(id), \beta \cdot G_1) = \\ &= e(H_0(id), G_1)^{\alpha\beta} = \\ &= e(H_0(id), \alpha \cdot G_1)^{\beta} = \\ &= e(H_0(id), u_1)^{\beta} = \\ &= e(H_0(id), \beta \cdot u_1) \end{aligned}$$

## 9.5 Resum

En aquest capítol s'han presentat els *pairings* sobre corbes el·líptiques, tot descrivint-ne les seves propietats. Després, d'una banda, s'han descrit les eines matemàtiques necessàries per entendre la seva formulació explícita i s'ha explicat com construir-los. D'altra banda, s'han presentat els algorismes criptogràfics més populars que els fan servir: un esquema de signatures que permet agregació (l'esquema BLS) i un esquema de criptografia basada en la identitat (l'esquema de Boneh-Franklin).

## 9.6 Solucions dels exercicis

**Exercici 9.1:**

El grau d'immersió d'$E$ respecte a $n = 5$ és $k = 1$ ja que $5 \mid 11^2 - 1$.

El grau d'immersió d'$E$ respecte a $n = 7$ no està definit, ja que $7 \nmid 15$.

El grau d'immersió d'$E$ respecte a $n = 15$ no està definit, ja que 15 no és primer.

**Exercici 9.2:**

La funció pot expressar-se com:

$$f(x) = \frac{(x-1)^3(x+5)^2}{(x-12)^4} = (x-1)^3(x+5)^2(x-12)^{-4}$$

i, per tant:

$$div(f) = 3\langle 1\rangle + 2\langle -5\rangle - 4\langle 12\rangle - \langle \infty\rangle$$

**Exercici 9.3:**

La funció $v_{Q+S}$ és la recta vertical que passa pel punt $Q+S$:

$$Q+S = (21,12z) + (10z+18,13z+13) = (19z+22,12z+10)$$

$$v_{Q+S} : x = 19z+22$$

La funció $l_{Q,S}$ és la recta que passa pels punts $Q$ i $S$:

$$m = \frac{y_2 - y_1}{x_2 - x_1} = \frac{13z+13-12z}{10z+18-21} = 3z+1$$

$$y = mx + c; c = y - mx = 12z - 21(3z+1) = 18z+2$$

$$l_{Q,S} : y = (3z+1)x + (18z+2)$$

La funció $f_{r,Q}$ es construeix iterativament a partir d'$f_{1,Q}$:

$$f_{1,Q} = 1$$

$$f_{2,Q} = f_{1,Q}\left(\frac{l_{Q,Q}}{v_{2Q}}\right) = \frac{y+11zx+10z}{x-21}$$

$$f_{3,Q} = f_{2,Q}\, v_Q = \frac{y+11zx+10z}{x-21}(x-21) = y+11zx+10z$$

**Exercici 9.4:**

La signatura d'un missatge $m$ seria:

$$\begin{aligned}\sigma &= \alpha \cdot H(m) = \\ &= \alpha \cdot H'(m) \cdot G_1 = \\ &= H'(m) \cdot \alpha \cdot G_1 = \\ &= H'(m) \cdot u\end{aligned}$$

i, per tant, la signatura dependria únicament del missatge $m$ i la clau pública $u$. Així doncs, un atacant podria crear signatures vàlides només amb informació pública.

**Exercici 9.5:**

1. Calculem la clau pública:

$$k_2^{pub} = \alpha \cdot G_1 = 7(2,8z) = (28,13z) \in \mathbb{G}_1$$

2. Calculem la signatura:

$$\sigma_2 = \alpha_2 \cdot H(m) = 7(41,35) = (15,30) \in \mathbb{G}_0$$

3. Calculem la signatura agregada:

$$\sigma_{ag} = \sigma_1 + \cdots + \sigma_n = \sigma_1 + \sigma_2 = (4,36) + (15,30) = (41,35)$$

4. Per validar la signatura agregada es comprova si $e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(m_1), k_1^{pub}) \cdot e(H(m_2), k_2^{pub})$:

$$e(\sigma_{ag}, G_1) = e((41,35),(2,8z)) = 40z+11$$
$$e(H(m_1), k_1^{pub}) \cdot e(H(m_2), k_2^{pub}) = e((41,35),(39,7z)) \cdot e((41,35),(28,13z)) = 40z+11$$

La igualtat es compleix, de manera que la signatura agregada és vàlida.

5. Per validar la signatura agregada aprofitant que les dues signatures corresponen al mateix missatge comprovem si $e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(m), k_1^{pub} + \cdots + k_n^{pub})$:

$$e(\sigma_{ag}, G_1) = e((41,35),(2,8z)) = 40z+11$$
$$e(H(m), k_1^{pub} + \cdots + k_n^{pub}) = e((41,35),(39,7z)+(28,13z)) = 40z+11$$

De nou, la igualtat es compleix de manera que la signatura agregada és vàlida.

**Exercici 9.6:**

Per validar la signatura agregada, el verificador comprovarà si

$$e(\sigma_{ag}, G_1) \stackrel{?}{=} e(H(m), k_{pubV}) \cdot e(H(m), k_{pubM})$$

Com que:

$$\begin{aligned} e(\sigma_{ag}, G_1) &= e(\alpha H(m), G_1) = \\ &= e(H(m), \alpha G_1) = \\ &= e(H(m), k_{pubA}) = \\ &= e(H(m), k_{pubV} + k_{pubM}) = \\ &= e(H(m), k_{pubV}) \cdot e(H(m), k_{pubV}) \end{aligned}$$

la verificació serà correcta i la signatura agregada serà donada per vàlida.

## 9.7 Bibliografia

**Boneh, Dan; Lynn, Ben; i Shacham, Hovav** (2001). *Short signatures from the Weil pairing*. International conference on the theory and application of cryptology and information security. Springer, Berlin, Heidelberg.

**Boneh, Dan; Drijvers, Manu; i Neven, Gregory** (2018). *Compact Multi-Signatures for Smaller Blockchains*. International Conference on the Theory and Application of Cryptology and Information Security.

**Boneh, Dan; i Victor Shoup** (2020). *A graduate course in applied cryptography*.

**Costello, Craig** (2012). *Pairings for beginners*.

**Martin, Luther** (2008). *Introduction to identity-based encryption*. Artech house.

**Kerry, Cameron F.; i Charles Romine** (2013). *NIST FIPS PUB 186-4: Digital Signature Standard (DSS)*.

**Open University, The** (2016). *Further pure mathematics: Group theory*.

**Paar, Christof, and Jan Pelzl** (2009). *Understanding cryptography: a textbook for students and practitioners*. Springer Science & Business Media.

**Perlroth, Nicole** (2013). *Government announces steps to restore confidence on encryption standards*. The New York Times.

**Schwennesen, Ben; i Hubert Bray** (2016). *Elliptic curve cryptography and government backdoors*.

**Warburton, David** (2019). *The 2019 TLS Telemetry Report*. F5.