Pairings sobre Curvas Elípticas

Eduardo Soria Vázquez

Memoria del Trabajo de Fin de Grado
dirigido por Juan Gabriel Tena Ayuso

Grado en Matemáticas
Universidad de Valladolid
2013–2014

defendido el 14 de julio de 2014

Universidad de Valladolid

Índice general

Lista de algoritmos

Agradecimientos

A Juan Tena Ayuso, por su inestimable ayuda a la hora de dirigir este trabajo, y a los miembros de la Universidad de Valladolid, en particular a aquellos del Departamento de Álgebra, Geometría y Topología que me han escuchado y orientado durante este periodo.

À l'Université de Versailles Saint-Quentin-en-Yvelines, où j'étais très chaleureusement acueilli et pour tout ce que l'Université a fait pour moi au long des deux dernières années, avec une mention spéciale pour M. Guillermo Moreno-Socías et M. Vincent Cossart.

A un nivel más personal, a mi familia, por su apoyo y comprensión incalculables, pero sobre todo por enseñarme las lecciones más valiosas de la vida. Por último, pero no menos importante, a aquellos que me asistieron de forma sincera y desinteresada en los momentos en los que olvidé dichas lecciones.

A todos y cada uno de ellos, junto a todos los que pudiera haber olvidado, muchas gracias.

À tous et à chacun, et à tous ceux que j'aurais pu oublier, je tiens à vous exprimer mes plus sincères remerciements.

Estructura del trabajo

Los pairings son aplicaciones bilineales definidas sobre pares de puntos de una curva elíptica y con valores en un cierto grupo abeliano multiplicativo. En el año 1993 se descubre su uso criptoanalítico, como herramienta de ataque al Problema del Logaritmo Discreto en las curvas elípticas. Durante años, ésta fue la única aplicación conocida de los pairings en criptología, hasta que en el 2000 se descubre su primer uso constructivo. A raíz de ello, la disciplina de la criptología basada en pairings alcanza un enorme desarrollo, cuya revisión es el objetivo de este Trabajo de Fin de Grado.

Comenzamos en el Capítulo 1 con un breve repaso de los fundamentos e historia de la criptología, centrándonos en el caso de la criptografía de clave pública por ser el que más nos atañe.

En el Capítulo 2 revisamos la teoría básica sobre curvas elípticas. Tras una serie de definiciones y resultados que manejaremos habitualmente, definimos la estructura de grupo de los puntos de una curva elíptica y damos métodos efectivos para el cálculo de la operación de grupo. Desde ese momento centramos nuestra atención en el caso de las curvas elípticas sobre cuerpos finitos, recordando su potencial como elemento básico a la hora de elaborar criptosistemas basados en el Problema del Logaritmo Discreto.

En el Capítulo 3 definimos el concepto de grado de inmersión de una curva elíptica, cuya relación con el grupo de llegada de los pairings se explicitará en el capítulo siguiente. Antes de hacerlo, nos dedicamos a catalogar y construir curvas elípticas cuyo grado de inmersión sea pequeño, dado el gran interés que esto representará en otros capítulos.

En el Capítulo 4 damos la definición explícita del pairing de Weil, cuya construcción requiere de la introducción de conceptos elementales de la Teoría de Divisores de una curva elíptica. Para concluir, definimos el pairing de Weil modificado, de interés criptográfico más práctico, y damos un algoritmo eficiente para su cálculo.

En el Capítulo 5 mostramos algunas de las técnicas criptológicas basadas en pairings más relevantes, algunas de las cuales son imposibles sin el uso de éstos.

Por último, en el Capítulo 6, damos una breve historia de los sistemas de voto electrónico, además de discutir sus requisitos de seguridad y compararlos con los sistemas de votación clásicos en papel. Tras esta introducción exponemos un sistema concreto de voto electrónico basado en pairings, que hace uso de numerosos resultados expuestos a lo largo de todo el Trabajo de Fin de Grado.

Capítulo 1

Nociones básicas de criptología

Los secretos han formado parte de las civilizaciones desde su mismo origen. Siempre ha habido información que trataba de ocultarse fuera de ciertos grupos, ya fuera con la intención de salvaguardar los intereses de un gobierno, los de un negocio, o los de individuos concretos. La problemática toma más relevancia de la que nunca tuvo en el presente, donde los pagos en línea o mediante tarjetas inteligentes, el correo electrónico, las bases de datos de carácter sensible y otra enorme variedad de gestos diarios afectan a cada ciudadano y al conjunto de ellos. Una pérdida del control sobre la información almacenada o en circulación podría tener gravísimas consecuencias para todos los partícipes.

Uno de los métodos de transmitir y almacenar estos datos de forma segura consiste en la utilización de la *criptografía* (del griego krypto, "oculto", y graphos, "escribir"), que podemos definir como el estudio y la concepción de procedimientos para cifrar (ocultar) una determinada información. El *criptoanálisis* se opone a esta ciencia, pues tiene como objetivo encontrar la información oculta, de la cual no se es el destinatario, mediante el análisis de los métodos criptográficos y los textos cifrados. Al conjunto de ambas disciplinas se le denomina *criptología*. La *esteganografía*, que consiste en la disimulación misma de la existencia de información secreta, es ajena a los intereses de este trabajo.

1.1. Caracterización de los sistemas criptográficos

Establezcamos en primer lugar una clasificación global de los diferentes métodos criptográficos. La información a transmitir recibe el nombre de "mensaje o texto *en claro*" y, tras aplicarle las transformaciones oportunas para ocultarlo, el de "mensaje *cifrado* o *encriptado*". Una *función de encriptado*, o de cifrado, es por lo tanto la dada por una trasformación:

$$f : \mathcal{M} \longrightarrow \mathcal{C}$$

donde $\mathcal{M}$ representa el conjunto de los mensajes en claro y $\mathcal{C}$ el conjunto de los mensajes cifrados. La transformación f^{-1} es la trasformación de *desencriptado*.

La experiencia ha demostrado que cuando una función de encriptado f ha sido utilizada en un gran número de ocasiones, se ha vuelto cada vez más vulnerable. Es por lo tanto deseable poder cambiar regularmente de función f. Con tal fin, definimos un *criptosistema* o *sistema criptográfico* o de *cifrado*, como una familia:

$$\mathcal{F} = (f_K)_{K \in \mathcal{K}}$$

de funciones de encriptado, cada una de ellas determinadas mediante un parámetro K, llamado *clave*. Según el principio de Kerchoffs, para un criptosistema lo suficientemente bien concebido se puede hacer $\mathcal{F}$ público sin perder seguridad, ya que el secreto de K resultaría suficiente para asegurar la confidencialidad del mensaje. Los sistemas donde $\mathcal{F}$ permanece oculto son conocidos como *criptosistemas de uso restringido* o de "seguridad por oscuridad" y son considerados como inseguros por la comunidad criptográfica por su falta de exposición a un análisis riguroso.

He aquí el contexto habitual: Dos entidades, expedidor y destinatario, a los que llamaremos Alice y Bob respectivamente, tratan de comunicarse en presencia de un observador o *criptoanalista*, a quien llamaremos Eve. El objetivo de Eve es el de *desencriptar* el mensaje cifrado C transmitido, es decir, deducir el mensaje original M. Idealmente, desearía encontrar también la transformación ${f_K}^{-1}$.

Dentro de los sistemas donde $\mathcal{F}$ es público (*criptosistemas de uso general*), podemos todavía distinguir entre los *criptosistemas simétricos* o de *clave privada* y los *criptosistemas asimétricos*, o *de clave pública*. Obsérvese que en estos casos Eve conoce la familia $\mathcal{F}$, pero el secreto de K aporta la seguridad.

Un criptosistema se dice de clave privada si el conocimiento de la función f_K implica el conocimiento de la función de desencriptado ${f_K}^{-1}$. En otras palabras, la inversa de la función de encriptado es fácil de calcular y ésta ha de mantenerse por lo tanto en secreto.

De forma opuesta, un criptosistema se dice de clave pública cuando el conocimiento de f_K no permite conocer en un tiempo razonable la función de desencriptado ${f_K}^{-1}$, pudiendo por lo tanto volver f_K pública. A partir de este hecho, cualquiera puede enviar mensajes cifrados a Bob, no solamente Alice. Permite además el establecimiento de sistemas de firma digital, en los cuales Bob puede firmar con su clave privada y el resto del mundo puede verificar que se trata de él utilizando su clave pública. En lo que se refiere a terminología, llamaremos *clave pública* a la información necesaria para construir f_K y *clave privada* a la información necesaria para construir ${f_K}^{-1}$, guardada en secreto.

1.2. Historia de la criptología basada en pairings

Lo que sigue es una breve reseña histórica de la investigación que ha conducido a los diversos elementos analizados en este trabajo, eludiendo específicamente la historia de la criptología clásica. Para una revisión global de la historia de la criptología, los libros divulgativos [Sin11] y [Kah96], el primero de los cuales está disponible en castellano, resultan lecturas amenas y clarificadoras.

Uno de los avances más relevantes en criptografía es el de la concepción de los criptosistemas de clave pública, debido al célebre artículo "New Directions in Cryptography" [DH76] de Diffie y Hellman en 1976. Es difícil imaginar cómo sería el tan interconectado mundo de hoy sin su revolucionaria solución para el clásico problema de la distribución de claves.

Si bien las curvas elípticas han sido sujeto de investigación durante mucho tiempo por su aparición natural en otras áreas de las Matemáticas, no es hasta el año 1985 cuando encuentran su propio lugar en la criptografía. Koblitz y Miller descubren de forma independiente que los criptosistemas basados en el logaritmo discreto (véase Sección 1.3.2) podrían proveer mayor seguridad en el caso de ser definidos sobre el grupo de puntos de una curva elíptica en lugar de sobre el grupo multiplicativo de un cuerpo finito. Desde ese momento, numerosos criptosistemas

basados en curvas elípticas fueron propuestos, algunos de los cuales resultaron no ser tan seguros como se suponía inicialmente.

Es en este momento cuando los pairings entran por primera vez en juego. Los pairings son aplicaciones bilineales definidas sobre pares de puntos de una curva elíptica y con valores en un cierto grupo abeliano multiplicativo. En 1993, Menezes, Okamoto y Vanstone descubrieron que el pairing de Weil podía ser usado para realizar ataques al Problema del Logaritmo Discreto en una cierta categoría de curvas elípticas. Un año después, Frey y Rück usan el pairing de Tate para describir un ataque similar a otra categoría de curvas. Estas técnicas criptoanalíticas fueron la única aplicación conocida de los pairings durante varios años, hasta que en el 2000 Joux descubre que pueden ser utilizados con fines constructivos. A raíz de ello, los esfuerzos de investigación en criptografía basada en pairings resultan tremendamente notables.

Sin lugar a dudas, una de las aplicaciones más importantes de la criptografía basada en pairings es la construcción efectiva de un criptosistema basado en la identidad. Esta variante de la criptografía de clave pública, propuesta por Shamir en 1984, permite construir las claves públicas de los usuarios a partir de su identidad. Las claves privadas correspondientes han de ser calculadas por una autoridad de confianza, pero acaba con la necesidad de autenticar las claves públicas de los usuarios mediante los farragosos sistemas de certificados utilizados hasta entonces en los sistemas clásicos de clave pública, que también requieren de dicha autoridad en la mayoría de los casos. Los responsables fueron, en este caso, Boneh y Franklin en el año 2001, aunque de manera simultánea Cocks dio con otra solución que no necesitaba el uso de pairings. Esquemas de firma basados en la identidad compatibles con el criptosistema de Boneh y Franklin surgieron poco después, permitiendo una solución funcional completa para el problema propuesto por Shamir.

Las aplicaciones de los pairings en la criptología son no obstante y como mostraremos mediante ejemplos concretos mucho más numerosas: protocolos de intercambio de claves, esquemas de firma corta, firmas ciegas, etc. En última instancia, veremos cómo incluso permiten la construcción efectiva de un sistema de voto electrónico.

1.3. Fundamentos de criptografía

A continuación listamos una serie de problemas y herramientas básicos que nos serán útiles a la hora de proponer y evaluar los diversos criptosistemas que aparezcan a lo largo del texto.

1.3.1. Funciones resumen

Una de las herramientas fundamentales de la criptografía moderna son las funciones resumen criptográficas, a menudo llamadas funciones resumen de una vía. Su motivación es la de dar una representación corta de cualquier secuencia binaria de entrada. Habitualmente, constituyen la pieza central del conocido como *modelo del oráculo aleatorio*, [BR93] en el cual se basa la seguridad de numerosos protocolos criptográficos. Damos a continuación una definición muy simplificada, pero suficiente para nuestros intereses. Una visión más amplia y profunda puede encontrarse en [MVOV96, Sección 1.9 y Capítulo 9].

1.1 Definición. Una función resumen o *función hash* h es una función computacionalmente eficiente que envía secuencias binarias de una longitud arbitraria a secuencias binarias de una longitud prefijada, que reciben el nombre de valor-resumen.

Para que una función resumen que dé como salida valores-resumen de n bits (típicamente, $n = 128$ o 160) tenga propiedades deseables, la probabilidad de que una secuencia binaria escogida aleatoriamente tenga como imagen un valor-resumen determinado ha de ser 2^{-n}. Para su uso criptográfico, una función resumen h tiene que verificar que sea computacionalmente imposible encontrar dos entradas distintas que den el mismo valor-resumen (que *colisionen*, es decir, encontrar x e y tales que $h(x) = h(y)$) y que, dado un determinado valor-resumen y, sea computacionalmente imposible encontrar una contraimagen x tal que $h(x) = y$.

1.3.2. El Problema del Logaritmo Discreto

1.2 Definición. Sea $(G, \star)$ un grupo. El *Problema del Logaritmo Discreto* (de forma abreviada, PLD) en G consiste en determinar, para un par dado de elementos g y h en G tales que $h \in \langle g \rangle$ un entero x que satisfaga

$$\underbrace{g \star g \star \cdots \star g}_{x \text{ veces}} = h.$$

La seguridad de los criptosistemas basados en grupos reposa sobre la hipótesis de que el medio más rápido de romperlos es la resolución del Problema del Logaritmo Discreto. Aunque en la práctica son evaluados en función de si existen o no algoritmos capaces de romperlos en un tiempo razonable, existen resultados teóricos que justifican la utilización del PLD como primitiva criptográfica. Victor Shoup publica en 1997 un resultado concerniente a "grupos genéricos" que dice que la resolución del PLD mediante un "algoritmo genérico" requiere $\Omega(p^{1/2})$ operaciones, donde p es el mayor divisor primo del orden del grupo (para más detalles, véase [Sho97]).

El resultado es de gran importancia, pues en esencia afirma que no se puede resolver, a priori, el PLD en tiempo polinomial (ni tan siquiera subexponencial) en el factor p descrito. No obstante, ciertos grupos particulares pueden estar dotados de características o estructuras algebraicas suplementarias capaces de conducir a algoritmos de resolución mucho más eficaces.

Esta es una lección significativa a la hora de comprender el estudio en este área: Para grupos distintos, el Problema del Logaritmo Discreto muestra niveles distintos de dificultad. En $(\mathbb{F}_p, +)$ es resoluble en tiempo lineal, mientras que el mejor algoritmo general conocido para resolver el PLD en $(\mathbb{F}_q^*, *)$, conocido como Index Calculus, es subexponencial [HR83]. En el Capítulo 2 veremos que se puede dotar de una estructura de grupo a las curvas elípticas. El Problema del Logaritmo Discreto para las curvas elípticas se cree de mayor complejidad computacional que el PLD para $\mathbb{F}_q^*$. En particular, si la curva elíptica y su grupo son elegidos cuidadosamente y éste tiene N elementos, el mejor algoritmo conocido para resolver el *Problema del Logaritmo Discreto Elíptico* (que abreviaremos PLDE) toma tiempo exponencial en N.

1.3.3. Los problemas de Diffie-Hellman

A lo largo de toda la sección consideramos un grupo multiplicativo finito $(G, *)$ de orden ℓ primo. Es decir, G cíclico generado por un elemento g. Íntimamente relacionado con el Problema del Logaritmo Discreto se encuentra el Problema Computacional de Diffie-Hellman:

1.3 Problema Computacional de Diffie-Hellman (PCDH). Para $a, b \in \mathbb{Z}_\ell^*$, dados (g, g^a, g^b), el Problema Computacional de Diffie-Hellman consiste en encontrar $h \in G$ tal que $h = g^{ab}$. Le asignamos la notación $\mathrm{CDH}_g(g^a, g^b) = h$.

Obviamente, el PCDH no es más difícil que el PLD, ya que la capacidad para computar logaritmos discretos permite resolver directamente el Problema Computacional de Diffie-Hellman. Por lo tanto, cuando el PCDH es difícil en un grupo, también lo es el PLD. Aunque se desconoce si se verifica la implicación inversa, ambos problemas suelen ser considerados equivalentes cuando se trata de evaluar la seguridad de los criptosistemas.

A parte del Problema Computacional de Diffie-Hellman existe una versión más débil, conocida como el Problema de Decisión de Diffie-Hellman:

1.4 Problema de Decisión de Diffie-Hellman (PDDH). Para $a, b, c \in \mathbb{Z}_\ell^*$, dados (g, g^a, g^b, g^c), el Problema de Decisión de Diffie-Hellman consiste en decidir si $c \equiv ab \pmod{\ell}$. Le asignamos la notación $\mathrm{DDH}_g(g^a, g^b, g^c) = 1$ si $\mathrm{CDH}_g(g^a, g^b) = g^c$, en cuyo caso decimos que (g, g^a, g^b, g^c) es una cuádrupla válida de Diffie-Hellman. En caso contrario, asignamos $\mathrm{DDH}_g(g^a, g^b, g^c) = 0$.

Resulta claro que el PCDH es como mínimo tan difícil como su variante decisional. De hecho, un método de resolver el PDDH sería computar $h = \mathrm{CDH}_g(g^a, g^b)$ y verificar si $g^c = h$. Para la mayoría de grupos, sin embargo, no existen resultados que garanticen que el PDDH es más sencillo que el PCDH. Esta brecha o *grieta* entre ambos problemas da lugar a la siguiente definición:

1.5 Problema Grieta de Diffie-Hellman (PGDH). Para $a, b \in \mathbb{Z}_\ell^*$, dados (g, g^a, g^b), el Problema Grieta de Diffie-Hellman (en inglés, *Gap Diffie-Hellman Problem*) consiste en resolver el Problema Computacional de Diffie-Hellman $\mathrm{CDH}_g(g^a, g^b)$, con la posibilidad de ayudarse de un oráculo de Decisión de Diffie-Hellman.

Para $g_1, g_2, g_3 \in G$ arbitrarios, un oráculo de Decisión de Diffie-Hellman da la respuesta correcta al problema $\mathrm{DDH}_g(g_1, g_2, g_3)$ en tiempo polinómico. El PGDH es a lo sumo tan difícil como el PCDH, ya que hay que resolver siempre el problema computacional para resolver el problema grieta. No obstante, para la mayoría de los grupos no resulta claro si el PGDH es estrictamente más sencillo que el PCDH.

El Problema Grieta de Diffie-Hellman aparece de manera natural en los grupos donde el Problema Computacional es difícil pero el Problema Decisional es fácil. A dichos grupos les denominaremos grupos Grieta Diffie-Hellman. En ellos, el PGDH y el PCDH resultan equivalentes, ya que se puede disponer de un oráculo de Decisión de Diffie-Hellman.

La siguiente imagen muestra la relación entre el PLD, el PCDH y el PDDH en un grupo Grieta Diffie-Hellman. La línea de puntos separa los problemas difíciles de los fáciles, y A $\longrightarrow$ B significa que el problema A es al menos tan difícil como el problema B:

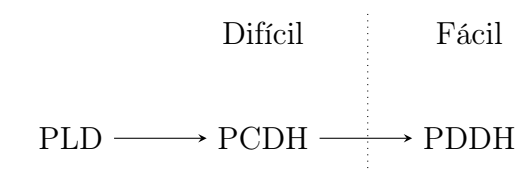

Figura 1.1: Grupo Grieta Diffie-Hellman.

Aunque pueda parecer artificial a priori, la definición del problema grieta resulta de utilidad en numerosas aplicaciones criptográficas, como los esquemas de firma que veremos en la Sección 5.3. La seguridad de un esquema de firma está basada en un problema computacional difícil, mientras que la verificación de la firma es un problema de decisión que debería de resultar sencillo.

Existen muchas otras variantes de los problemas de Diffie-Hellman, cuya definición resulta necesaria en determinadas ocasiones. Tal será el caso del Problema Bilineal de Diffie-Hellman, que definiremos en el Capítulo 5. Posponemos para entonces su formalización, ya que requiere el uso de varios de los conceptos que introduciremos en los próximos tres capítulos.

Capítulo 2

Curvas Elípticas

El objetivo de este capítulo es el de introducir algunas nociones y resultados básicos sobre las curvas elípticas que nos serán de utilidad posteriormente. Al tratarse de un capítulo de preliminares se han omitido muchas de las pruebas de los resultados, para las cuales el lector puede dirigirse a un tratado específico de curvas elípticas como [Sil09]. En lo que sigue, $\Bbbk$ representará siempre a un cuerpo y $\bar{\Bbbk}$ a su clausura algebraica.

2.1. Primeras definiciones

2.1 Definición. Una curva elíptica sobre $\Bbbk$ es un par (E, O) donde E es una curva algebraica proyectiva no singular de género 1 sobre $\Bbbk$ y O un punto con coordenadas en $\Bbbk$.

2.2 Definición. Una ecuación de Weierstraß sobre $\Bbbk$ es una ecuación de la forma:

$$E_w : y^2 + a_1xy + a_3y = x^3 + a_2x^2 + a_4x + a_6 \quad \text{donde} \quad a_i \in \Bbbk \ \forall i. \tag{2.1}$$

Esta ecuación define una curva en $\mathbb{P}_2(\Bbbk)$ con exactamente un punto en el infinito, a saber $(0 : 1 : 0)$, al que denotaremos $\mathcal{O}$ en adelante. Dada una curva E definida de esta manera, definimos asimismo las cantidades:

$$\begin{aligned} d_2 &= a_1^2 + 4a_2 \\ d_4 &= 2a_4 + a_1a_3 \\ d_6 &= a_3^2 + 4a_6 \\ d_8 &= a_1^2a_6 + 4a_2a_6 - a_1a_3a_4 + a_2a_3^2 - a_4^2 \\ c_4 &= d_2^2 - 24d_4 \\ \Delta &= -d_2^2d_8 - 8d_4^3 - 27d_6^2 + 9d_2d_4d_6 \\ j(E) &= c_4/\Delta. \end{aligned}$$

2.3 Definición. La cantidad Δ se llama el *discriminante* de la ecuación de Weierstraß, mientras que $j(E)$ se llama el *j-invariante* de E si $\Delta \neq 0$.

2.4 Teorema. *La curva definida por una ecuación de Weierstraß es una curva no singular si y solamente si el discriminante es no nulo. De ser así, decimos que la ecuación de Weierstraß es no singular.*

2.5 Teorema. *Sea (E, O) una curva elíptica definida sobre $\bar{\Bbbk}$. Entonces existe una ecuación de Weierstraß no singular sobre $\bar{\Bbbk}$ y un isomorfismo de variedades $\phi : E \to E_w$ tal que $\phi(O) = \mathcal{O}$.*

Lo que este último resultado muestra, en esencia, es que toda curva elíptica admite un modelo plano en la forma de Weierstraß y recíprocamente, toda ecuación de Weierstraß no singular define una curva elíptica. Es por ello que en ocasiones abusaremos ligeramente de la notación y denotaremos E a la ecuación (2.1), haciendo referencia a la curva elíptica que define.

2.6 Definición. Escribimos $E/\Bbbk$ si E se puede expresar mediante una ecuación cuyos coeficientes se encuentran todos en $\Bbbk$.

Dos curvas elípticas se dicen *isomorfas* si lo son como variedades proyectivas. Brevemente, dos variedades proyectivas V_1, V_2 definidas sobre $\Bbbk$ son isomorfas sobre $\Bbbk$ si existen morfismos $\phi : V_1 \to V_2$, $\psi : V_2 \to V_1$, tales que $\phi \circ \psi$ y $\psi \circ \phi$ son la aplicación identidad en V_1 y V_2 respectivamente. El siguiente resultado relaciona la noción de isomorfismo de curvas elípticas con los coeficientes de sus ecuaciones de Weierstraß.

2.7 Teorema. *Si dos curvas elípticas $E_1/\Bbbk$ y $E_2/\Bbbk$ son isomorfas sobre $\Bbbk$, entonces $j(E_1) = j(E_2)$. La implicación recíproca es también cierta si $\Bbbk$ es un cuerpo algebraicamente cerrado.*

Si una curva elíptica está definida sobre un cuerpo $\Bbbk$ cuya característica no sea 2 ni 3, su ecuación de Weierstraß correspondiente puede ser simplificada considerablemente. Es por ello, y porque estaremos habitualmente en tal caso, que introducimos el siguiente resultado.

2.8 Proposición. *Sea $(E, \mathcal{O})$ una curva elíptica dada por una ecuación de Weierstraß. Si* $\mathrm{char}(\Bbbk) \neq 2, 3$ *entonces se puede realizar un cambio de variables afín en $\bar{\Bbbk}$ que envía al punto $\mathcal{O}$ él mismo y transforma (2.1) en:*

$$y^2 = x^3 + Ax + B \quad \textit{donde} \quad A, B \in \bar{\Bbbk} \tag{2.2}$$

cuyo discriminante es $\Delta = \Delta(A, B) = -16(4A^3 + 27B^2) \neq 0$ y cuyo invariante es $j(E) = -1728(4A^3)/\Delta$.

2.9 Definición. Denotamos $E(\Bbbk)$ al conjunto de puntos de E que se pueden escribir con todas sus coordenadas en $\Bbbk$. A estos puntos los llamamos $\Bbbk$-racionales o, cuando se sobreentienda el cuerpo al que nos referimos, racionales.

2.10 Ejemplo. Si el punto $(\sqrt{13} : 0 : 2\sqrt{13}) \in E$ tenemos que $(\sqrt{13} : 0 : 2\sqrt{13}) \in E(\mathbb{Q})$, ya que $(\sqrt{13} : 0 : 2\sqrt{13}) = (1 : 0 : 2)$.

2.2. La estructura de grupo de las curvas elípticas

Lo que sigue son una serie de definiciones y resultados orientados a construir la estructura de grupo para los puntos racionales de una curva elíptica. Dicha estructura, junto con sus características concretas, interviene de forma directa en la construcción de toda la teoría y herramientas que veremos a lo largo de los próximos capítulos.

2.11 Definición. $(PQ)_E$ designa, si $P \neq Q$, la recta que pasa por P y Q, o si $P = Q$, la tangente a E en P.

2.12 Lema. *Toda recta corta a una curva elíptica en exactamente tres puntos, contándolos con sus respectivas multiplicidades. Si $P, Q \in E(\Bbbk)$ entonces el tercer punto de $(PQ)_E \cap E$ pertenece a $E(\Bbbk)$.*

Demostración. Consecuencia directa del Teorema de Bézout. □

Nos valdremos a continuación del punto prefijado $O \in E(\Bbbk)$. Sean $P, Q \in E(\Bbbk)$.

2.13 Definición. Definimos $S = P \underset{O}{+} Q$ del siguiente modo. La recta $(PQ)_E$ corta a E en un tercer punto R. La recta $(OR)_E$ corta E en un tercer punto S, que denotaremos $P \underset{O}{+} Q$.

La existencia y unicidad de $P \underset{O}{+} Q$ (y de R) queda garantizada por el lema anterior.

2.14 Proposición. *Si los tres puntos de intersección de una recta con E son P_1, P_2, P_3, entonces $(P_1 \underset{O}{+} P_2) \underset{O}{+} P_3 = T_0$, donde T_0 es el tercer punto de $(OO)_E \cap E$.*

Demostración. Según la construcción que acabamos de enunciar, P_3 es el tercer punto R de intersección entre la recta $(P_1P_2)_E$ y E. Decir que $P_1 \underset{O}{+} P_2$ es el tercer punto de intersección entre $(OP_3)_E$ y E es equivalente a decir que O es el tercer punto de intersección entre $(P_1 \underset{O}{+} P_2, P_3)_E$ y E.

Finalmente, la recta $(OO)_E$ corta a E en el punto T_0 buscado. Resaltemos el caso $O = \mathcal{O}$ y O de abscisa nula, donde $T_0 = \mathcal{O}$. Es decir, que si $O = \mathcal{O}$, la proposición se reescribe como $(P_1 \underset{\mathcal{O}}{+} P_2) \underset{\mathcal{O}}{+} P_3 = \mathcal{O}$. □

2.15 Teorema. *Si $O \in E(\Bbbk)$, entonces $(E(\Bbbk), \underset{O}{+})$ es un grupo abeliano de elemento neutro O.*

Demostración. La demostración resulta sencilla a partir de los últimos resultados, exceptuando la propiedad asociativa. Para ello, puede revisarse [Sil09, Proposición III.3.4e].

□

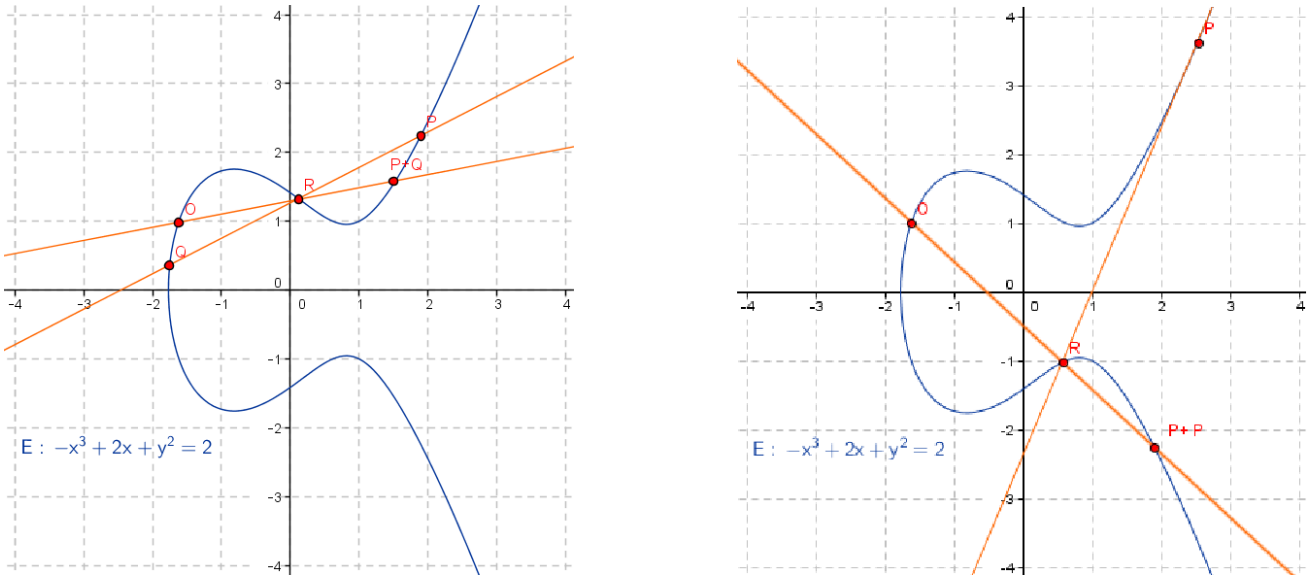

Figura 2.1: Cálculo de $P \underset{O}{+} Q$ (izquierda) y $P \underset{O}{+} P$ (derecha) para la curva elíptica $y^2 = x^3 - 2x + 2$.

2.16 Corolario. *Sea $\tilde{\Bbbk}$ un subcuerpo de $\Bbbk$ tal que $O \in E(\tilde{\Bbbk})$. $(E(\tilde{\Bbbk}), \underset{O}{+})$ es un subgrupo abeliano de $(E(\Bbbk), \underset{O}{+})$.*

2.3. Ecuaciones analíticas de la suma de puntos

Según los resultados vistos en el apartado anterior podemos, siempre que $\mathrm{char}(\Bbbk) \neq 2, 3$, reducirnos al caso de una curva $(E, \mathcal{O})$ sobre $\Bbbk$ dada por una ecuación de Weierstraß de la forma $y^2 = x^3 + Ax + B$ donde $A, B \in \Bbbk$, $\Delta(A, B) \neq 0$ y $\mathcal{O} = (0 : 1 : 0)$. Si además elegimos $O = \mathcal{O}$ la operación $\underset{\mathcal{O}}{+}$ se vuelve mucho más simple.

Siendo el cálculo más frecuente el del múltiplo de un punto, buscamos optimizarlo al menos ligeramente. Para ello, nos valdremos de una técnica genérica, conocida bajo el nombre inglés de *square and multiply* que permite realizar esta operación de manera mucho más eficaz que mediante sumas sucesivas.

En lo que sigue, una curva elíptica $(E, \mathcal{O})$ sobre $\Bbbk$ será siempre de la forma indicada arriba, y en lugar de $\underset{\mathcal{O}}{+}$ escribiremos simplemente $+$.

2.3.1. Suma de puntos de una curva elíptica

Daremos a continuación fórmulas explícitas para el cálculo del opuesto y la suma de puntos de una curva elíptica.

2.17 Proposición. *Sea $E : y^2 = x^3 + Ax + B$.*

1. *Sea $P_0 = (x_0, y_0)$. Su punto opuesto es:*

$$-P_0 = (x_0, -y_0). \tag{2.3}$$

2. *Sean P_1, P_2 con $P_i = (x_i, y_i) \in E(\Bbbk)$ para $i = 1, 2$ y $P_2 \neq -P_1$. Buscamos $P_3 = (x_3, y_3) \in E(\Bbbk)$ tal que $P_3 = P_1 + P_2$. Definimos λ y ν según la siguiente tabla:*

	λ	ν
$P_1 \neq P_2$	$\dfrac{y_2 - y_1}{x_2 - x_1}$	$\dfrac{y_1 x_2 - y_2 x_1}{x_2 - x_1}$
$P_1 = P_2$	$\dfrac{3x_1^2 + A}{2y_1}$	$\dfrac{-x_1^3 + Ax_1 + 2B}{2y_1}$

Se verifica entonces que $y = \lambda x + \nu$ es la ecuación de la recta $(P_1 P_2)_E$, y además:

$$\begin{aligned} x_3 &= \lambda^2 - x_1 - x_2 \\ y_3 &= -\lambda x_3 - \nu = \lambda(x_1 - x_3) - y_1 \end{aligned}$$

Observación. En el caso $P_1 = P_2$ de la última tabla, se verifica siempre que $2y_1 \neq 0$, puesto que por (2.3), un punto P de ordenada nula es opuesto a sí mismo, y por tanto $P + P = \mathcal{O}$.

Demostración.

1. Para calcular $-P_0$, debemos tomar la intersección entre E y $(\mathcal{O}, P_0)_E$. Esta recta viene dada por la ecuación $x = x_0$. Sustituyendo con ella en la ecuación de E, nos encontramos ante un polinomio cuadrático $f(x_0, y) = y^2 - (x_0^3 + Ax_0 + B)$ que tiene por tanto dos raíces $y_0, y_0' \in \bar{\Bbbk}$, puesto que $-P_0 \in E(\Bbbk)$. Por lo tanto, $-P_0 = (x_0, y_0')$. Además, $f(x_0, y) = (y - y_0)(y - y_0')$ ya que es mónico. El estudio de los coeficientes de este polinomio en y nos da la solución:
$$0 = -y_0 - y_0'.$$

2. Sea $Q = (x_3', y_3')$ el tercer punto de intersección entre $(P_1P_2)_E$ y E. Entonces, la ecuación de $(P_1P_2)_E$ es $y = \lambda x + \nu$. Sustituyendo con ella en la ecuación de E, nos encontramos ante un polinomio cúbico $f(x, \lambda x + \nu) = -x^3 + (\lambda x + \nu)^2 - Ax - B$ con raíces x_1, x_2, x_3'. Tenemos también por tanto que $f(x, \lambda x + \nu) = -(x - x_1)(x - x_2)(x - x_3')$ y observando los coeficientes en x^2, concluimos que:
$$x_1 + x_2 + x_3' = \lambda^2.$$
Despejando, obtenemos el valor de x_3' y por lo tanto también el de $y_3' = \lambda x_3' + \nu$. Para finalizar, puesto que $P_1 + P_2 = -Q$, aplicamos la fórmula del opuesto a Q y sustituimos $\nu = y_1 - \lambda x_1$. □

Tenemos ahora fórmulas explícitas tanto para calcular la suma de puntos como su duplicación, es decir, el caso $P_1 = P_2$. Dichas fórmulas nos serán de utilidad en la próxima sección.

2.3.2. Cálculo del múltiplo de un punto

Una vez definida la ley de suma de puntos de una curva elíptica podemos definir, para todo $m \in \mathbb{Z}$, el morfismo de multiplicación por dicho entero.

2.18 Definición. Sea $(E, \mathcal{O})$ una curva elíptica sobre $\Bbbk$. Para todo entero m definimos el morfismo $[m] : E \longrightarrow E$, llamado de multiplicación por m, como sigue a continuación. Si m es positivo, $[m](P) = \underbrace{P + \cdots + P}_{m \text{ veces}}$. Si m es negativo, $[m](P) = [-m](-P)$. Si $m = 0$, entonces $[0](P) = \mathcal{O}$.

A lo largo del texto, abreviaremos habitualmente la notación escribiendo mP en lugar de $[m](P)$. Nos encontramos ahora en las condiciones adecuadas para definir con total rigor el Problema del Logaritmo Discreto Elíptico:

2.19 Definición. Sea $(E, \mathcal{O})$ una curva elíptica sobre $\Bbbk$ y sean $P, Q \in E(\Bbbk)$ puntos tales que Q pertenece al subgrupo generado por P. El Problema del Logaritmo Discreto Elíptico (PLDE) consiste en encontrar un entero m tal que $Q = mP$.

Existen numerosos algoritmos que permiten efectuar la multiplicación de un entero por un punto de manera más eficaz que mediante sumas sucesivas. A continuación mostraremos un método genérico que puede ser utilizado en cualquier grupo y que se basa en la representación

binaria del entero m. Conocido habitualmente como algoritmo de *exponenciación binaria*, en nuestro caso lo llamaremos de *doblado y adición*, ya que estamos en un grupo aditivo:

Algoritmo 1 Doblado y adición

Entrada: $P \in E(\Bbbk)$ y $m = m_0 + m_1 \cdot 2 + m_2 \cdot 2^2 + \cdots + m_{n-1} \cdot 2^{n-1} \in \mathbb{N}$
Salida: $Q = [k](P) \in E(\Bbbk)$
1: $Q \leftarrow P$
2: **para** $i = n - 2$ **hasta** 0 **hacer**
3: $\quad Q \leftarrow [2](Q)$
4: $\quad$ **si** $m_i = 1$ **entonces**
5: $\quad\quad Q \leftarrow Q + P$
6: $\quad$ **fin si**
7: **fin para**
8: **devolver** Q

Con este algoritmo realizamos $n - 1$ multiplicaciones por dos (doblados) y un número de sumas igual al número de cifras no nulas de la representación binaria de m, menos una. Mucho más eficiente que realizar m sumas, aunque existen por supuesto algoritmos aún mejores en múltiples casos.

2.4. Curvas elípticas sobre cuerpos finitos

En esta sección, y en los capítulos posteriores, trabajaremos salvo que se explicite lo contrario con curvas elípticas definidas sobre cuerpos finitos $\Bbbk = \mathbb{F}_q$, donde q es la potencia de un número primo p. Esta elección está motivada por la obtención de grupos finitos de gran interés criptográfico, donde es cómodo operar y la dificultad de resolución del Problema del Logaritmo Discreto es grande. El siguiente resultado (véase [Men93]) nos aporta más información sobre la estructura de dichos grupos:

2.20 Teorema. *$(E(\mathbb{F}_q), \underset{\mathcal{O}}{+})$ es un grupo abeliano finito de rango 1 ó 2. El tipo de grupo es (n_1, n_2), es decir $E(\mathbb{F}_q) \cong \mathbb{Z}_{n_1} \oplus \mathbb{Z}_{n_2}$, donde $n_2 | n_1$ y además $n_2 | q - 1$.*

Los siguientes resultados nos hablan sobre el endomorfismo de Frobenius y la traza de Frobenius para una determinada curva elíptica E. Serán de gran relevancia en lo posterior.

2.21 Definición. Sea $E/\mathbb{F}_q$ una curva elíptica sobre un cuerpo finito $\mathbb{F}_q$. Llamamos (q-ésimo) endomorfismo de Frobenius $\pi : E \to E$ a la aplicación:

$$\pi(x, y) = (x^q, y^q), \qquad \pi(\mathcal{O}) = \mathcal{O}$$

Tenemos que $\pi \in End(E)$ para cualquier curva elíptica $E/\mathbb{F}_q$.

2.22 Teorema. *Sea $E/\mathbb{F}_q$ una curva elíptica y sea π el q-ésimo endomorfismo de Frobenius. Entonces:*

1. *Existe un entero $t = t_q$, al que llamamos traza de Frobenius, tal que*

$$\pi^2 - t\pi + q = 0. \tag{2.4}$$

es decir, que para todo $P \in E$, tenemos la ecuación

$$\pi^2(P) - t\pi(P) + qP = \mathcal{O}.$$

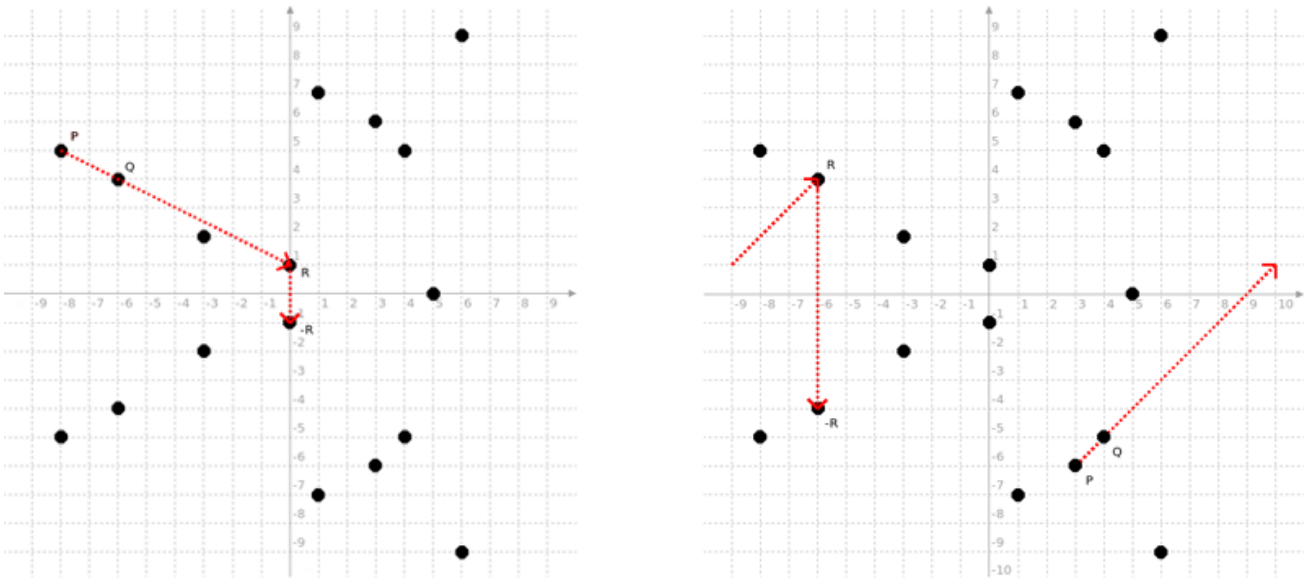

Figura 2.2: Cálculo de $P \underset{\mathcal{O}}{+} Q = -R$ para la curva elíptica $y^2 = x^3 - 10x + 1$ en $\mathbb{Z}_{19} \times \mathbb{Z}_{19}$. La naturaleza cíclica del cuerpo de definición hace que en la imagen de la derecha, la recta que ha de unir P y Q con R desaparezca por un lateral y aparezca por el opuesto.

2. *La traza t del q-ésimo endomorfismo de Frobenius está vinculada con el número de puntos racionales de la curva mediante la relación:*

$$\#E(\mathbb{F}_q) = q + 1 - t$$

Demostración. Un esquema sencillo de la prueba pueden encontrarse en [SZ03, Teorema 3.2]. Se apoya entre otros en el resultado [Sil09, Corolario III.5.5]. □

Hasse halló una cota para la traza de Frobenius y, por tanto, para el cardinal de una curva elíptica definida sobre un cuerpo finito.

2.23 Teorema de Hasse. *Sea E una curva elíptica sobre $\mathbb{F}_q$. Entonces:*

$$\#E(\mathbb{F}_q) = q + 1 - t \text{ con } t \text{ verificando } |t| \leqslant 2\sqrt{q}$$

Demostración. Véase [Sil09, Teorema V.1.1]. □

2.24 Definición. Sea $E/\mathbb{F}_q$ una curva elíptica sobre un cuerpo finito $\mathbb{F}_q$. El discriminante de la ecuación 2.4 es negativo o nulo, y lo denotamos $D_\pi = t^2 - 4q$.

En función del valor de su traza de Frobenius, las curvas elípticas pueden ser catalogadas según la siguiente serie de definiciones. Las distintas denominaciones permiten, entre otras cosas, diferenciar si son vulnerables a distintos tipos de ataques al logaritmo discreto elíptico mediante pairings, conocidos como algoritmos de reducción.

2.25 Definición. Una curva elíptica E definida sobre un cuerpo finito $\mathbb{F}_q$ de característica p se denomina supersingular si $t \equiv 0 \pmod{p}$, es decir, si $p|t$. En caso contrario, recibe el nombre de ordinaria.

Las curvas supersingulares, que revisaremos en el próximo capítulo, son interesantes gracias a que Menezes, Okamoto y Vanstone mostraron en 1993 [MOV93] como trasladar mediante el pairing de Weil el PLDE de una curva definida sobre el cuerpo finito $\mathbb{F}_q$ al PLD sobre una cierta extensión de grado pequeño de $\mathbb{F}_q$.

2.26 Definición. Una curva elíptica E definida sobre un cuerpo finito primo $\mathbb{F}_p$ recibe el nombre de anómala si $t = 1$.

Tres artículos, debidos a Satoh y Araki [SA98], Semaev [Sem98] y Smart [Sma99], casi simultáneos, dan la manera de reducir el PLDE sobre estas curvas al PLD sobre un grupo aditivo. Su revisión se aleja de los objetivos de este trabajo.

2.27 Definición. Una curva elíptica E definida sobre un cuerpo finito $\mathbb{F}_q$ de característica p recibe el nombre de curva FR si $t = 2$.

El sobrenombre recibido por estas curvas proviene de las siglas de Frey y Rück [FR94] que en 1994 mostraron su vulnerabilidad mediante un ataque de reducción que utiliza el pairing de Tate y funciona de manera similar al de Menezes, Okamoto y Vanstone.

2.5. Puntos de torsión de una curva elíptica

Describimos a continuación brevemente los puntos de orden finito de una curva elíptica.

2.28 Definición. Sea E una curva elíptica definida sobre un cuerpo $\Bbbk$ y sea $m \geqslant 1$ un entero. A un punto $P \in E(\bar{\Bbbk})$ que cumpla $mP = \mathcal{O}$ le llamamos punto de orden m o punto de m-torsión de E. Denotamos al conjunto de puntos de m-torsión del siguiente modo:

$$E[m] = \left\{ P \in E(\bar{\Bbbk}) : mP = \mathcal{O} \right\}.$$

Es sencillo comprobar que $E[m]$ es un subgrupo de $E(\bar{\Bbbk})$. Si queremos además denotar que las coordenadas de un determinado punto de m-torsión están en un determinado cuerpo $\mathbb{K}$, $\Bbbk \subseteq \mathbb{K} \subseteq \bar{\Bbbk}$, escribimos $E(\mathbb{K})[m]$. Podemos definir de forma más precisa la estructura de grupo de $E[m] = E(\bar{\Bbbk})[m]$:

2.29 Proposición. *Sea E una curva elíptica sobre $\Bbbk$ y sea $m \in \mathbb{Z}$ con $m \neq 0$ en $\Bbbk$, es decir, o bien $\mathrm{char}(\Bbbk) = 0$ o $\mathrm{char}(\Bbbk) = p > 0$ y $p \not| m$. Entonces:*

$$E[m] \cong \frac{\mathbb{Z}}{m\mathbb{Z}} \times \frac{\mathbb{Z}}{m\mathbb{Z}}$$

Demostración. Puede consultarse en [Sil09, Corolario III.6.4]. □

En criptografía se trabaja habitualmente con el subgrupo cíclico engendrado por un punto $P \in E(\mathbb{F}_q)$ de orden ℓ primo. Otra consideración usual es que, si $N = \#E(\mathbb{F}_q)$, se verifique que $mcd(N, q) = 1$ y por lo tanto $p \not| \ell$. De no ser así, es decir si $p|\ell$, nos hallaríamos ante una curva anómala, para la cual el PLDE puede resolverse en tiempo lineal y no nos resulta por lo tanto interesante.

En este caso común, y que será el que consideremos en adelante, podemos por lo tanto afirmar que $E[\ell] \cong \mathbb{Z}/\ell\mathbb{Z} \times \mathbb{Z}/\ell\mathbb{Z}$. Sin embargo, estos ℓ^2 puntos de $E[\ell]$ no están todos necesariamente definidos sobre el cuerpo base $\mathbb{F}_q$ y en general $E(\mathbb{F}_q)[\ell]$ será sólo un subgrupo de $E[\ell]$. Veamos dos ejemplos:

2.30 Ejemplo. La curva elíptica $E : y^2 = x^3 + x + 8$, sobre el cuerpo $\mathbb{F}_{11}$, posee sólo dos puntos de 2-torsión con coeficientes en el cuerpo base. Concretamente, $E(\mathbb{F}_{11})[2] = \{\mathcal{O}, (8, 0)\} \cong \mathbb{Z}/2\mathbb{Z}$.

2.31 Ejemplo. Consideremos, para la curva elíptica $E : y^2 = x^3 + 7x$, sobre el cuerpo $\mathbb{F}_{13}$, $\ell = 3$. En este caso, sus 9 puntos de 3-torsión son $\mathbb{F}_{13}$-racionales. Explícitamente:

$$E(\mathbb{F}_{13})[3] = E[3] = \{\mathcal{O}, (3, 3), (3, 10), (4, 1), (4, 12), (9, 5), (9, 8), (10, 2), (10, 11)\}.$$

Estos puntos constituyen un subgrupo isomorfo al grupo $\mathbb{Z}/3\mathbb{Z} \times \mathbb{Z}/3\mathbb{Z}$, el cual contiene cuatro subgrupos de orden 3. Cada uno de ellos está formado por un punto, su opuesto y elemento neutro:

- $G_1 = \{\mathcal{O}, (3, 3), (3, 10)\}$.
- $G_2 = \{\mathcal{O}, (4, 1), (4, 12)\}$.
- $G_3 = \{\mathcal{O}, (9, 5), (9, 8)\}$.
- $G_4 = \{\mathcal{O}, (10, 2), (10, 11)\}$.

2.6. Seguridad de los criptosistemas basados en curvas elípticas

La motivación de los resultados expuestos a lo largo del tema es la de proporcionar las herramientas necesarias para el estudio de criptosistemas basados en curvas elípticas. Si bien la mayoría de los protocolos basados en el Problema del Logaritmo Discreto fueron originalmente definidos sobre el grupo multiplicativo de un cuerpo finito $\mathbb{F}_q^*$, éste, junto con los problemas de Diffie-Hellman, pueden ser definidos sobre un grupo cualquiera.

Cuando en el año 1985 se propuso el uso del grupo de puntos de una curva elíptica, la razón principal fue que para él no existía un ataque análogo al Index Calculus (recordemos la Sección 1.3.2). Desprovistos de esta técnica criptoanalítica, capaz de resolver el PLD en tiempo subexponencial en el grupo multiplicativo de un cuerpo finito, la complejidad del problema pasaba a ser a priori exponencial en el grupo de puntos de una curva elíptica. Gracias a ello los criptosistemas basados en curvas elípticas pueden ofrecer el mismo nivel de seguridad que sus análogos en cuerpos finitos utilizando claves de longitud mucho más corta. El hecho de que existan numerosas curvas diferentes con cardinal similar puede ser también considerado como una ventaja adicional.

Muchos investigadores han tratado sin éxito extender el método del Index Calculus a las curvas elípticas. No obstante, el hecho de que dicho ataque no pueda ser utilizado no garantiza que no exista ningún otro algoritmo capaz de resolver el Problema del Logaritmo Discreto Elíptico en un tiempo inferior al exponencial. De hecho, las curvas elípticas están dotadas de una cierta estructura que convierte, a algunas más que a otras, en blancos potenciales para el diseño de algoritmos eficientes para la resolución del PLDE.

A pesar de la cota dada por el Teorema de Hasse, para una curva elíptica arbitraria sobre un cuerpo finito, computar su número de puntos racionales (es decir, el orden del grupo que definen) resulta ser un arduo problema sobre el que se ha publicado ampliamente. Puede consultarse al

respecto [BSS05, Capítulo VI]. Por este motivo, durante los primeros años de la criptografía basada en curvas elípticas, los diseñadores de estos criptosistemas tendían a elegir curvas con una estructura particular que les permitiera computar su número de puntos de manera rápida. Es el caso de las curvas supersingulares (propuestas en [Mil86] y [Kob87]) y las curvas anómalas (propuestas en [Miy93]) que, como comentamos en la sección anterior, se probaron unos pocos años más tarde vulnerables a determinados ataques de reducción.

Resulta por lo tanto desaconsejable seleccionar curvas elípticas basándose en una estructura suplementaria que facilite el conteo de puntos y/o simplifique la aritmética en ellas. Una solución natural sería pues la de evitar tales curvas, eligiendo para el protocolo una curva de forma totalmente aleatoria y verificando a posteriori que no sea susceptible a ataques de reducción, en cuyo caso se desecharía para repetir el proceso. En [BK98] se muestra que, con una probabilidad abrumadora, una curva seleccionada de tal modo es robusta ante ataques de reducción del tipo MOV y FR. La otra posibilidad pasa por tomar un cuerpo finito lo suficientemente grande como para que el criptosistema sea seguro no sólo en el grupo de puntos de la curva elíptica, sino también en el correspondiente grupo de llegada de los ataques de reducción. Si bien esta solución puede parecer absurda a primera vista por eliminar la ventaja del uso de claves más cortas, en algunos casos permite el diseño de grupos Grieta Diffie-Hellman (véase la Sección 5.3.1).

Capítulo 3

Grado de Inmersión

En las aplicaciones criptográficas basadas en el Problema del Logaritmo Discreto Elíptico, como ya hemos comentado anteriormente, suele utilizarse un subgrupo cíclico $\langle P \rangle \subseteq E(\mathbb{F}_q)$ del mayor orden primo posible. El orden ℓ de este subgrupo, que será pues un divisor del cardinal de $E(\mathbb{F}_q)$, determinará el grupo de llegada μ_ℓ de los pairings que estudiaremos en el Capítulo 4. El valor del grado de inmersión de una curva elíptica, relacionado con este μ_ℓ, será crucial para las técnicas criptográficas y criptoanalíticas expuestas en el Capítulo 5 y en última instancia para el protocolo de voto electrónico expuesto en el Capítulo 6. Por dichos motivos resulta ser el tema central de este Trabajo de Fin de Grado y le dedicamos el capítulo al completo. El siguiente resultado nos permitirá definir el concepto con el rigor pertinente:

3.1 Lema. *Sea ℓ un divisor primo del cardinal $N = \#E(\mathbb{F}_q)$ y tal que $mcd(q, \ell) = 1$. Existe un entero positivo k que verifica las condiciones equivalentes:*

1. $\ell | (q^k - 1)$.
2. $\mathbb{F}_{q^k}^*$ *contiene un subgrupo cíclico de orden ℓ .*

Demostración. El grupo $(\mathbb{F}_{q^k})^*$ es cíclico, con cardinal $q^k - 1$. Tal grupo contiene a un subgrupo de cardinal ℓ si, y solamente si, $\ell | (q^k - 1)$, es decir, $q^k \equiv 1 \pmod{\ell}$. Ahora bien, por hipótesis, $mcd(q, \ell) = 1$, y por tanto $q \in (\mathbb{Z}/\ell\mathbb{Z})^*$. El orden k de q en tal grupo es una solución al problema. □

3.2 Definición. El mínimo k verificando el Lema 3.1 se denomina grado de inmersión de $E/\mathbb{F}_q$ respecto a ℓ. Si ℓ es el mayor divisor primo de N, entonces decimos simplemente que k es el grado de inmersión de $E/\mathbb{F}_q$.

3.3 Ejemplo. Sea $E : y^2 = x^3 + 1$ una curva elíptica definida sobre $\mathbb{F}_{101}$, de la cual sabemos además que $\#E(\mathbb{F}_{101}) = 102$. El punto $P = (87, 61)$ pertenece a la curva y tiene orden 17. Para calcular el grado de inmersión de $E/\mathbb{F}_{101}$ respecto a 17, hay que encontrar el mínimo k tal que $17 | 101^k - 1$. Es evidente que no se verifica para $k = 1$, pero para $k = 2$, $101^2 - 1 = 17 \cdot 600$. Luego $E/\mathbb{F}_{101}$ tiene grado de inmersión 2 respecto a 17

Aunque veremos que para las curvas supersingulares el grado de inmersión es siempre pequeño y fácil de determinar, para las curvas elípticas ordinarias éste es en general muy grande (exponencial en $log(q)$, según demostraron Koblitz y Balasubramanian en [BK98]). La búsqueda de curvas ordinarias con grado de inmersión pequeño resulta, por los motivos previamente expuestos, un interesante problema que en nuestro caso vamos a abordar para las familias de curvas de j-invariante 0 y 1728.

3.1. Curvas supersingulares

Gracias al trabajo de Menezes, Okamoto y Vanstone sabemos que cualquier curva supersingular E puede clasificarse en uno de las seis tipos que resumimos en la tabla siguiente. Para la prueba de los resultados, el lector puede referirse a [Men93, Lema 2.9, Corolario 2.11, Lema 2.13]. Recordamos que t hace referencia a la traza de Frobenius, y destacamos una vez más el hecho que vuelve tan especiales a las curvas supersingulares: Todas tienen un grado de inmersión $k \leqslant 6$.

Tipo	t	Estructura del Grupo	n_1	k
I	0	Cíclico	$(q+1)$	2
II	0	$\mathbb{Z}_{(q+1)/2} \oplus \mathbb{Z}_2$	$(q+1)/2$	2
III	$\pm\sqrt{q}$	Cíclico	$q+1 \mp \sqrt{q}$	3
IV	$\pm\sqrt{2q}$	Cíclico	$q+1 \mp \sqrt{2q}$	4
V	$\pm\sqrt{3q}$	Cíclico	$q+1 \mp \sqrt{3q}$	6
VI	$\pm 2\sqrt{q}$	$\mathbb{Z}_{\sqrt{q}\mp 1} \oplus \mathbb{Z}_{\sqrt{q}\mp 1}$	$\sqrt{q} \mp 1$	1

3.2. Curvas sobre $\mathbb{F}_q$ con j-invariante 0 ó 1728

A lo largo de tanto de esta sección como de la próxima, ℓ representará un número primo y E una curva elíptica definida sobre un cuerpo finito $\mathbb{F}_q$, con $q = p^m$, $p \geqslant 5$ primo. Buscamos, dentro de las familias propuestas, distinguir las curvas supersingulares de las ordinarias y caracterizarlas. Con tal fin citamos los siguientes resultados (véase [MT93]), que nos dan las posibles clases de isomorfismos de curvas elípticas con j-invariante 1728 ó 0.

3.4 Proposición. *El número de clases de isomorfismos de curvas elípticas con j-invariante 1728 sobre $\mathbb{F}_q$, donde $q = p^m$ con $p \geqslant 5$ primo, viene dado por:*

1. *Si $q \equiv 3 \pmod 4$ (luego $p \equiv 3 \pmod 4$ y m es impar), existen dos clases de isomorfismos, con representantes:*

   $$y^2 = x^3 + x, \qquad y^2 = x^3 - x \tag{3.1}$$

   Ambas curvas son supersingulares.

2. *Si $q \equiv 1 \pmod 4$, existen cuatro clases de isomorfismos, con representantes:*

   $$E_r : y^2 = x^3 + \omega^r x, \qquad 0 \leqslant r \leqslant 3, \tag{3.2}$$

   Donde ω es un generador de $\mathbb{F}_q^*$. Para $p \equiv 3 \pmod 4$, luego m par, estas curvas son supersingulares. En caso contrario, son ordinarias.

Observación. Las curvas E_0 y E_2 son independientes del generador particular ω, pero E_1 y E_3 pueden ser intercambiadas cuando se modifica éste.

3.5 Proposición. *El número de clases de isomorfismos de curvas elípticas con j-invariante 0 sobre $\mathbb{F}_q$, donde $q = p^m$ con $p \geqslant 5$ primo, viene dado por:*

1. *Si $q \equiv 2 \pmod 3$, existen dos clases de isomorfismos, con representantes:*

$$y^2 = x^3 + 1, \qquad y^2 = x^3 + b, \quad b \in \mathbb{F}_q^* \setminus (\mathbb{F}_q^*)^2 \tag{3.3}$$

   Ambas curvas son supersingulares.

2. *Si $q \equiv 1 \pmod 3$, existen seis clases de isomorfismos, con representantes:*

$$E'_r : y^2 = x^3 + \omega^r, \quad 0 \leqslant r \leqslant 5, \tag{3.4}$$

   Donde ω es un generador de $\mathbb{F}_q^*$. Para $p \equiv 2 \pmod 3$, luego m par, estas seis curvas son supersingulares. En caso contrario, son ordinarias.

Observación. Sólo las curva E'_0 y E'_3 son independientes del generador particular ω.

Descartamos el estudio de las curvas supersingulares de ambas familias por poder englobarse en el estudio general de Menezes ya expuesto y pasamos a centrarnos en el caso ordinario.

3.3. Curvas ordinarias con grado de inmersión pequeño

Existen varios métodos para la búsqueda de curvas elípticas ordinarias con grado de inmersión pequeño. Algunos de ellos pueden encontrarse en [BSS05, Sección IX.15.] o, para una revisión más profunda y exhaustiva, en [FST10].

Habitualmente la idea, basada en las propiedades del endomorfismo de Frobenius que enunciamos en la Sección 2.4, consiste en encontrar para un k dado una ecuación $t^2 - 4q = D_\pi h^2$ adecuada, con D_π libre de cuadrados y pequeño. A continuación, se busca una curva elíptica con discriminante D_π y cardinal $q + 1 - t$. No obstante, el método que nosotros seguiremos, expuesto en [MST], utiliza una aproximación distinta: Fijamos $D_\pi = -1, -3$ (o lo que es lo mismo, los recién expuestos $j(E) = 1728$ y $j(E) = 0$ con ecuaciones de Weierstraß $y^2 = x^3 + Ax$, $y^2 = x^3 + B$ respectivamente) y buscamos ℓ primo y q adecuados que permitan obtener el k deseado. El siguiente resultado, fruto de Cocks y Pinch [BW05] nos será de ayuda:

3.6 Lema. *Una curva elíptica E tiene grado de inmersión k respecto a ℓ si y sólo si $t \equiv 1 + \zeta_k \pmod \ell$, donde ζ_k es una raíz primitiva k-ésima de la unidad módulo ℓ.*

Según la sección anterior, si $p \geqslant 5$ primo, una curva elíptica E sobre $\mathbb{F}_q$, con $j(E) = 1728$ es ordinaria cuando $p \equiv 1 \pmod 4$ y curvas elípticas con $j(E) = 0$, cuando $p \equiv 1 \pmod 3$. Completamos la caracterización de las dos familias de curvas mediante el siguiente resultado, debido a Munuera y Tena [MT93]:

3.7 Lema. *La traza de Frobenius t_r de las curvas E_r dadas en (3.2) verifica:*

1. $t_0 \equiv 2 \pmod 4$, $t_0/2 \equiv 1 \pmod 4$;
2. $t_1 \equiv 0 \pmod 4$, $t_3 = -t_1 \equiv 0 \pmod 4$;

3. $t_2 \equiv 2 \pmod 4,\ t_2/2 \equiv 3 \pmod 4$;

La traza de Frobenius t'_r *de las curvas* E'_r *dadas en (3.4) verifica:*

1. $t'_0 \equiv 2 \pmod 6$;

2. $t'_1 \equiv 1 \pmod 6,\ t'_4 = -t'_1 \equiv 5 \pmod 6$;

3. $t'_2 \equiv 5 \pmod 6,\ t'_5 = -t'_2 \equiv 1 \pmod 6$;

4. $t'_3 \equiv 4 \pmod 6$;

En lo subsiguiente consideraremos de forma separada los casos de grado de inmersión 1 y 2, además de dar un método general para k superiores. Para completar el capítulo, daremos ejemplos de curvas calculadas mediante nuestro método.

3.3.1. Curvas ordinarias con grado de inmersión 1

De acuerdo con el Lema 3.6, la traza de Frobenius de una curva elíptica con grado de inmersión $k = 1$ respecto a ℓ debe ser $t \equiv 2 \pmod \ell$.

Curvas de j-invariante 1728

Consideremos en primer lugar las curvas elípticas de j-invariante 1728. Entonces, tenemos discriminante $D_\pi = -1$ y se puede por tanto probar, aunque no lo haremos, que el endomorfismo de Frobenius $\pi \in \mathbb{Z}[i]$. Calculamos según ésto su traza:

$$t = Tra(\pi) = \pi + \bar{\pi} = (a + bi) + (a - bi) = 2a$$

De esto último, y puesto que $t \equiv 2 \pmod \ell$, se deduce que $a = 1 + c\ell$, luego $t = 2a = 2 + 2c\ell$ para un cierto $c \in \mathbb{Z}$. Por otro lado, ℓ divide al cardinal de la curva, luego:

$$\ell | \#E(\mathbb{F}_q) = q + 1 - t = q + 1 - 2 + 2c\ell \Rightarrow q \equiv 1 \pmod \ell$$

Lo cual nos permite calcular la norma de π y avanzar otro paso:

$$q = N(\pi) = \pi\bar{\pi} = (a + bi)(a - bi) = a^2 + b^2 \equiv 1 \pmod \ell$$

Y como vimos que $a = 1 + c\ell$, entonces $\ell | b$ y tenemos un entero d de tal forma que $b = d\ell$. Sustituyendo, lo que acabamos de probar es que:

3.8 Teorema. *Una de las cuatro curvas* $E_r/\mathbb{F}_q$ *tiene grado de inmersión 1 respecto a* ℓ *si y sólo si:*

$$q = (c^2 + d^2)\ell^2 + 2c\ell + 1, \quad c, d \in \mathbb{Z}, \tag{3.5}$$

Dicha curva tiene cardinal $(c^2 + d^2)\ell^2$*. Además, si* $\ell > 2$*, entonces* $c \equiv d \pmod 2$.

La conclusión sobre la paridad compartida por c y d se deduce a partir de la expresión dada para q, que es impar. El teorema anterior no especifica cual de las cuatro curvas tiene grado de inmersión 1, pero a partir del Lema 3.7, podemos ser más precisos:

3.9 Proposición. *Sea $E_r/\mathbb{F}_q$ con grado de inmersión 1 respecto a un primo impar ℓ expresada mediante las ecuaciones del Teorema 3.8. Entonces:*

1. *Si $c \equiv 1 \pmod 2$, se trata de E_1 ó E_3.*
2. *Si $c \equiv 0 \pmod 4$, se trata de E_0.*
3. *Si $c \equiv 2 \pmod 4$, se trata de E_2.*

Demostración. Recordemos del Teorema 3.8 que $t = 2 + 2c\ell$. Todas las conclusiones son aplicando el Lema 3.7. Si c es impar, entonces $t = 2 + 2(2c' + 1)\ell \equiv 2(1 + \ell) \pmod 4$. Como ℓ es impar, la curva obtenida es E_1 ó E_3, pero no podemos distinguir más al intercambiarse la traza de ambas según el generador de $\mathbb{F}_q^*$ elegido.

Si c es par, entonces $t \equiv 2 \pmod 4$ y se trata de la traza de Frobenius de E_0 ó E_2. Como $t/2 = 1 + c\ell$, el resultado final se obtiene por simple aplicación de las hipótesis. □

Curvas de j-invariante 0

En el caso de las curvas de j-invariante 0, su anillo de endomorfismos es el anillo de enteros de $\mathbb{Q}(\sqrt{-3})$. Esto conlleva, aunque no lo vamos a demostrar, que el endomorfismo de Frobenius $\pi \in \mathbb{Z}[\frac{1+\sqrt{-3}}{2}]$.

3.10 Teorema. *Una de las seis curvas $E'_r/\mathbb{F}_q$ tiene grado de inmersión 1 respecto a $\ell > 3$ si y sólo si:*

$$q = (c\ell + 2)^2 + 3(d\ell + 1)(\ell(d - c) - 1), \quad c, d \in \mathbb{Z} \tag{3.6}$$

Dicha curva tiene cardinal $\ell^2(c^2 + 3d(d - c))$.

Demostración. Sea π el endomorfismo de Frobenius de $E'_r/\mathbb{F}_q$. Tenemos en esta ocasión que $\pi = a + b\frac{1+\sqrt{-3}}{2}$ para un par de enteros a, b. De esto obtenemos que $t = Tra(\pi) = 2a + b$. Por otro lado, por el Lema 3.6, existe $c \in \mathbb{Z}$ tal que $t = 2 + c\ell$. A raíz de estas dos ecuaciones, deducimos que $b = 2(1 - a) + c\ell$.

De la definición de π también tenemos que $q = N(\pi) = a^2 + b^2 + ab$. Sustituyendo con la expresión que obtuvimos para b, llegamos a que $q = (c\ell + 2)^2 + 3a(a - c\ell - 2)$. De esto se sigue:

$$\ell | \#E(\mathbb{F}_q) = q + 1 - t = q - 1 - c\ell \Rightarrow \ell | (q - 1) = c^2\ell^2 + 3 + 4c\ell + 3a^2 - 3ac\ell - 6a \Rightarrow \ell | 3(a - 1)^2$$

Por lo tanto, si asumimos $\ell > 3$, podemos tomar $a = d\ell + 1$, para un $d \in \mathbb{Z}$. Sustituyendo en la última igualdad dada para q, y junto con la última expresión que dimos para t, obtenemos los resultados enunciados. □

El teorema anterior no especifica cual de las seis curvas tiene grado de inmersión 1, pero a partir del Lema 3.7, podemos precisar más:

3.11 Proposición. *Sea $E'_r/\mathbb{F}_q$ con grado de inmersión 1 respecto a un primo $\ell > 3$ expresada mediante las ecuaciones del Teorema 3.10. Entonces:*

1. *Si $c \equiv 0 \pmod 6$, se trata de E'_0.*
2. *Si $c \equiv 3 \pmod 6$, se trata de E'_2 ó E'_4.*
3. *Si $c\ell \equiv 2 \pmod 6$, se trata de E'_3.*
4. *Si $c\ell \equiv 5 \pmod 6$, se trata de E'_1 ó E'_5*

Demostración. Recordemos del Teorema 3.10 que $t = 2 + c\ell$. Todas las conclusiones son aplicando el Lema 3.7, a raíz del cual los dos últimos enunciados son triviales. Veamos los otros dos: Si $c \equiv 0 \pmod 6$, entonces $t \equiv 2 \pmod 6$ y estamos ante E'_0. Si $c \equiv 3 \pmod 6$, entonces $t \equiv 2 + 3\ell \equiv 3(\ell + 1) - 1 \pmod 6$. Como $\ell + 1$ es par, $t \equiv -1 \pmod 6$ y la curva es E'_2 ó E'_4. □

3.3.2. Curvas ordinarias con grado de inmersión 2

De acuerdo con el Lema 3.6, la traza de Frobenius de una curva elíptica con grado de inmersión $k = 2$ respecto a ℓ debe ser $t \equiv 0 \pmod \ell$. Damos a continuación cuatro resultados que siguen la misma línea de los cuatro recién dados para $k = 1$.

Curvas de j-invariante 1728

3.12 Teorema. *Una de las cuatro curvas $E_r/\mathbb{F}_q$ tiene grado de inmersión 2 respecto a un primo impar ℓ si y sólo si:*

$$q = c^2\ell^2 + d\ell - 1, \quad c, d \in \mathbb{Z} \text{ y } d\ell - 1 \text{ cuadrado perfecto.} \tag{3.7}$$

Dicha curva tiene cardinal $\ell(c^2\ell + d - 2c)$.

Demostración. Como el j-invariante es 1728, tenemos que $\pi \in \mathbb{Z}[i]$, luego $\pi = a + bi$ con $a, b \in \mathbb{Z}$. Tenemos que:

$$q = N(\pi) = a^2 + b^2$$

$$t = Tra(\pi) = 2a \Rightarrow a = c\ell, c \in \mathbb{Z}$$

La última implicación se debe al Lema 3.6 y la hipótesis $\ell > 2$. Como ℓ divide el orden del grupo $E(\mathbb{F}_q)$:

$$\ell | \#E(\mathbb{F}_q) = q + 1 - t = a^2 + b^2 + 1 - 2a \Rightarrow \ell | (b^2 + 1) \Rightarrow b^2 = d\ell - 1, d \in \mathbb{Z}$$

El resultado final se obtiene por simple sustitución mediante las expresiones obtenidas para a y b. □

3.13 Proposición. *Sea $E_r/\mathbb{F}_q$ con grado de inmersión 2 respecto a un primo impar ℓ expresada mediante las ecuaciones del Teorema 3.12. Entonces:*

1. *Si $c \equiv 0 \pmod 2$, se trata de E_1 ó E_3.*
2. *Si $c \equiv 1 \pmod 2$ y $c\ell \equiv 1 \pmod 4$, se trata de E_0.*
3. *Si $c \equiv 1 \pmod 2$ y $c\ell \equiv 3 \pmod 4$, se trata de E_2.*

Demostración. Recordemos del Teorema 3.12 que $t = 2c\ell$. Todas las conclusiones son aplicando el Lema 3.7. Si c es par, entonces $t \equiv 0 \pmod 4$. La curva obtenida es pues E_1 ó E_3, pero no podemos distinguir más al intercambiarse la traza de ambas según el generador de $\mathbb{F}_q^*$ elegido.

Si c es impar, entonces $t \equiv 2(2c'+1)\ell \equiv 2\ell \equiv 2 \pmod 4$. Distinguimos E_0 de E_2 según la congruencia de $t/2 = c\ell$ módulo 4. □

Curvas de j-invariante 0

3.14 Teorema. *Una de las seis curvas $E'_r/\mathbb{F}_q$ tiene grado de inmersión 2 respecto a ℓ si y sólo si:*

$$q = (c\ell)^2 + 3a(a - c\ell), \text{ con } 3a^2 + 1 = d\ell \text{ y } c \not\equiv 0 \pmod 3. \tag{3.8}$$

Dicha curva tiene cardinal $(c\ell)^2 + 3a(a - c\ell) + 1 - c\ell$.

Demostración. Como el j-invariante es 0, tenemos que $\pi \in \mathbb{Z}[\frac{1+\sqrt{-3}}{2}]$, luego $\pi = a + b\frac{1+\sqrt{-3}}{2}$ con $a, b \in \mathbb{Z}$. Tenemos que:

$$q = N(\pi) = a^2 + b^2 + ab = (a+b)^2 - ab$$

$$t = Tra(\pi) = 2a + b \Rightarrow b = c\ell - 2a, c \in \mathbb{Z}$$

La última implicación se debe a que, por el Lema 3.6, $t = c\ell$. Como ℓ divide el orden del grupo $E(\mathbb{F}_q)$ encadenemos las siguientes deducciones, sustituyendo la expresión de t y luego la de b:

$$\ell | \#E(\mathbb{F}_q) = q + 1 - t = (a+b)^2 - ab + 1 - 2a - b \Rightarrow \ell | ((-a)^2 - a(-2a) + 1 - 2a + 2a) = 3a^2 + 1$$

Luego $3a^2 + 1 = d\ell$ para un cierto $d \in \mathbb{Z}$. Como en su expresión tenemos términos de grado uno en a, esta vez no seremos capaces de parametrizar q directamente según c y d. Sustituyendo en $N(\pi)$ el valor de b obtenemos la expresión de q. Para el cardinal sustituimos además con $t = c\ell$.

La congruencia de c con 0 módulo 3 implicaría que pudiésemos sacar factor común 3 de la expresión de q, lo cual es absurdo pues supusimos que era potencia de un primo $p \geqslant 5$. □

3.15 Proposición. *Sea $E'_r/\mathbb{F}_q$ con grado de inmersión 2 respecto a un primo ℓ expresada mediante las ecuaciones del Teorema 3.14. Entonces:*

1. *Si* $c \equiv 0 \pmod 2$*, se trata de* E'_0 *ó* E'_3*. Adicionalmente, si* $c\ell \equiv 2 \pmod 6$*, es* E'_0 *y si* $c\ell \equiv 4 \pmod 6$*, es* E'_3*.*

2. *Si* $c\ell \equiv 1 \pmod 6$*, se trata de* E'_1 *ó* E'_5*.*

3. *Si* $c\ell \equiv 5 \pmod 6$*, se trata de* E'_2 *ó* E'_4*.*

Demostración. Recordemos del Teorema 3.14 que $t = c\ell$. Si c es par, entonces $t \equiv 2$ ó $4 \pmod 6$, salvo si $\ell = 3$. La excepción no está contemplada en el enunciado ya que obtendríamos $t \equiv 0 \pmod 6$, lo cual es imposible para cualquiera de las curvas $E'_r/\mathbb{F}_q$.

A partir del Lema 3.7, todo lo demás es inmediato. □

3.3.3. Curvas ordinarias con grado de inmersión superior

Explicamos a continuación cómo generalizar el método anteriormente expuesto para buscar curvas elípticas ordinarias de j-invariante 0 ó 1728 y con grado de inmersión $k \geqslant 3$ respecto a un cierto ℓ primo.

Por el Lema 3.6, la raíz primitiva k-ésima de la unidad ζ_k depende del valor de ℓ y la traza de Frobenius t puede expresarse como $t = 1 + \zeta_k + c\ell$ para algún $c \in \mathbb{Z}$. Como además $\ell | \#E(\mathbb{F}_q) = q + 1 - t$, tenemos que $q \equiv \zeta_k \pmod \ell$. Nos valdremos de estas expresiones de t y q en los dos casos a estudiar:

Curvas de j-invariante 1728

Sabemos que $\pi \in \mathbb{Z}[i]$, luego $t = Tra(\pi) = 2a$ y por lo tanto $a = (1 + \zeta_k)/2 + c\ell/2$. Adicionalmente, $q = N(\pi) = a^2 + b^2$. Si sustituimos en esta última igualdad sabiendo que $q = d\ell$ para un $d \in \mathbb{Z}$ y valiéndonos de la expresión de a, obtenemos finalmente:

$$b^2 = d\ell - a^2 = -\frac{(1+\zeta_k)^2}{4} + (d - \frac{c^2\ell}{4})\ell$$

Con los ya prefijados ℓ y ζ_k y dando valores a c, d, obtenemos valores para a, b con los que calcular q, que será válido de ser potencia de un primo. En función del valor obtenido para la traza de Frobenius, en ocasiones podremos distinguir de cuál de las cuatro curvas E_r posibles se trata gracias al Lema 3.7.

Curvas de j-invariante 0

Sabemos que $\pi \in \mathbb{Z}[\frac{1+\sqrt{-3}}{2}]$, luego $t = Tra(\pi) = 2a + b$ y por lo tanto $b = t - 2a = 1 + \zeta_k + c\ell - 2a$. Adicionalmente, $q = N(\pi) = (a+b)^2 - ab$. Si sustituimos en la otra expresión de q requerimos que $(a+b)^2 - ab - \zeta_k \equiv 0 \pmod \ell$, donde sustituyendo con b y desarrollando, obtenemos la ecuación:

$$3a(a - \zeta_k - 1) + \zeta_k^2 + \zeta_k + 1 \equiv 0 \pmod \ell$$

Con los ya prefijados ℓ y ζ_k, con a satisfaciendo la última ecuación y dando valores a c, obtenemos valores para b con los que calcular q, que será válido de ser potencia de un primo. En función del valor obtenido para la traza de Frobenius, en ocasiones podremos distinguir de cuál de las seis curvas E'_r posibles se trata gracias al Lema 3.7.

3.3.4. Ejemplos numéricos

Presentamos a continuación ejemplos concretos de curvas calculadas mediante el método expuesto a lo largo de la sección. Es necesario destacar que son ejemplos sencillos, con valores de ℓ pequeños. Para el uso criptográfico de curvas de estas dos familias habría que seleccionar un ℓ primo mucho mayor.

La manera de proceder ha sido la siguiente: En primer lugar fijamos k, a continuación, el j-invariante y luego dimos valores a c y d. Sin fijar aún el primo ℓ, parametrizamos los valores de q y $\#E(\mathbb{F}_q)$ y tratamos de distinguir de qué curva de la familia seleccionada podría tratarse. Finalmente, para un ℓ concreto, verificamos que el q obtenido fuese potencia de un primo $p \geqslant 5$ e intentamos una vez más distinguir el tipo de curva de ser posible.

En esta primera tabla, damos ejemplos de curvas ordinarias con grado de inmersión $k = 1$ obtenidas mediante los Teoremas 3.8 y 3.10. El tipo de curva ha sido distinguido según las Proposiciones 3.9 y 3.11.

$j(E)$	c, d	ℓ	q	$\#E(\mathbb{F}_q)$	$Curva$
1728	0,2	ℓ	$4\ell^2+1$	$4\ell^2$	E_0
		73	21317	21316	E_0
	2,2	ℓ	$(2\ell+1)^2+4\ell^2$	$8\ell^2$	E_2
		41	13613	13448	E_2
	1,1	ℓ	$(\ell+1)^2+\ell^2$	$2\ell^2$	E_1 ó E_3
		79	12641	12482	E_1 ó E_3
0	2,0	ℓ	$4\ell^2+2\ell+1$	$4\ell^2$	?
		31	3907	3844	E'_3
	1,0	ℓ	$\ell^2+\ell+1$	ℓ^2	?
		59	3541	3481	E'_1 ó E'_5
	-3,1	ℓ	$21\ell^2-3\ell+1$	$21\ell^2$	E'_2 ó E'_4
		53	58831	58989	E'_2 ó E'_4

En la segunda tabla, damos ejemplos para curvas con grado de inmersión $k = 2$ obtenidas mediante los Teoremas 3.12 y 3.14. El tipo de curva ha sido distinguido según las Proposiciones 3.13 y 3.15.

$j(E)$	c, d	ℓ	q	$\#E(\mathbb{F}_q)$	$Curva$
1728	1,1	ℓ	$\ell^2 + \ell - 1$	$\ell(\ell - 1)$	E_0
		101	10301	10100	E_0
	3,1	ℓ	$9\ell^2 + \ell - 1$	$\ell(9\ell - 5)$	E_2
		101	91909	91304	E_2
	2,2	ℓ	$4\ell^2 + 2\ell - 1$	$\ell(4\ell - 2)$	E_1 ó E_3
		1013	4106701	4102650	E_1 ó E_3
0	1,1	$3a^2 + 1$	$\ell^2 - 3a\ell + 3a^2$	$\ell^2 - (3a + 1)\ell + 3a^2 + 1$	?
		1201	1371541	1370341	E_1' ó E_5'
	-1,1	$3a^2 + 1$	$\ell^2 + 3a\ell + 3a^2$	$\ell^2 + (3a + 1)\ell + 3a^2 + 1$	?
		193	42073	42267	E_2' ó E_4'
	4,4	$(3a^2 + 1)/4$	$16\ell^2 - 12a\ell + 3a^2$	$16\ell^2 - 4(3a + 1)\ell + 3a^2 + 1$	E_0' ó E_3'
		127	238759	238252	E_3'

Capítulo 4

El pairing de Weil

Construiremos finalmente en este capítulo un pairing concreto, el pairing de Weil, por ser el primero en aplicarse en criptología y por su mayor sencillez en comparación con el resto de los existentes. Además de recordar bien los conceptos vistos en los Capítulos 2 y 3, necesitaremos dar una pequeña introducción a la Teoría de Divisores de una curva elíptica para lograr la definición formal del pairing. Veremos, entonces, que si E es una curva elíptica definida sobre $\mathbb{F}_q$, el pairing de Weil es una aplicación bilineal que toma pares de elementos de $E[m]$ y da como salida una raíz m-ésima de la unidad en $\mathbb{F}_{q^k}$, donde k es el grado de inmersión de $E/\mathbb{F}_q$.

Continuando el capítulo definiremos el pairing de Weil modificado, que volverá a poner de manifiesto la relevancia de las curvas supersingulares en la criptología basada en pairings. Finalmente, daremos un método efectivo, conocido como el algoritmo de Miller, para calcular el valor de estas abstractas aplicaciones.

4.1. Divisores y funciones racionales

Para una construcción explícita de los pairings necesitaremos introducir en primer lugar la siguiente serie de conceptos y resultados. Aunque la Teoría de Divisores tiene un alcance mucho mayor, limitaremos nuestra exposición al caso concreto de las curvas elípticas.

4.1 Definición. Sea E una curva elíptica. Un divisor D es una suma formal finita de puntos de E, $D = \sum_{P \in E} n_P(P)$, donde los coeficientes n_P son números enteros todos nulos salvo una cantidad finita de ellos.

Obsérvese que en la anterior definición la notación de suma es sólo un símbolo formal, y no una operación. No ha de confundirse pues con la operación de suma de puntos de una curva elíptica. Tampoco han de confundirse los puntos $P \in E$ con los divisores $(P) = 1(P)$, ni pensar que estamos realizando multiplicaciones de puntos por enteros.

La próxima proposición, de fácil comprobación, nos permite construir una estructura algebraica apropiada para el conjunto de los divisores.

4.2 Proposición. *Dados dos divisores $D = \sum_{P \in E} n_P(P)$, $D' = \sum_{P \in E} n'_P(P)$, la fórmula de adición $D + D' = \sum_{P \in E}(n_P + n'_P)(P)$ confiere al conjunto de divisores una estructura de grupo abeliano.*

4.3 Definición. Sean E una curva elíptica y D un divisor de E:

1. Se define el soporte de de D como los puntos de coeficiente no nulo: $sop(D) = \{P \in E : n_P \neq 0\}$.
2. Se define el grado de D como la suma de sus coeficientes: $gr(D) = \sum_{P \in E} n_P \in \mathbb{Z}$.
3. Se define la suma de D como el resultado obtenido al realizar efectivamente las sumas y productos listados formalmente: $sum(D) = sum(\sum_{P \in E} n_P(P)) = \sum_{P \in E} n_P P \in E$.

El siguiente bloque de definiciones presenta las funciones racionales y su relación con los divisores de una curva elíptica.

4.4 Definición. Sea E una curva elíptica definida sobre $\Bbbk$.

1. Una función racional sobre E es una función del tipo
$$f(x, y) = \frac{F_1(x, y)}{F_2(x, y)},$$
donde $F_1, F_2 \in \bar{\Bbbk}[x, y]$ son polinomios definidos módulo la ecuación de la curva E.
2. Un punto $P \in E$ se denomina cero de una función racional f si f se anula en él (concretamente, F_1 se anula en él). Lo denominamos polo de f si f no está definida en P (porque F_2 se anula en él).
3. Dada una función racional f, llamamos su divisor asociado a
$$div(f) = \sum_{P \in E} n_P(P)$$
donde $n_P = n$ si P es un cero de multiplicidad n de f, $n_P = -n$ si P es un polo de multiplicidad n de f y $n_P = 0$ si P no es ni cero ni polo.
4. Llamamos divisores principales a aquellos para los cuales existe una función racional a la cual están asociados.
5. Dos divisores D, D' son equivalentes (y escribimos $D \sim D'$) si se diferencian en un divisor principal, es decir, si para una cierta función racional f se verifica $D = D' + div(f)$.

Es fácil comprobar que la última relación es efectivamente de equivalencia.

4.5 Teorema. *1. Si dos divisores principales $div(f)$, $div(f')$ son iguales, entonces sus funciones racionales asociadas son proporcionales ($\exists c \neq 0 : f = cf'$).*

2. Un divisor $D = \sum_{P \in E} n_P(P)$ es principal si y solamente si
$$gr(D) = 0 \text{ y } sum(D) = \mathcal{O}.$$

Demostración. Por motivos de extensión, referimos una vez más al lector a un tratado concreto sobre curvas elípticas [Sil09, Proposiciones II.3.1 y III.3.4]. □

4.6 Corolario. *1. Dos divisores equivalentes tienen el mismo grado.*

2. Si una función racional no tiene ni ceros ni polos, es constante.

4.7 Ejemplo. Sea la curva elíptica $E : y^2 = x^3 + x + 4$ definida sobre $\mathbb{F}_7$.

- Sea la recta $r : y = 2x+2$, cuyo divisor (principal) nos disponemos a calcular. Recordemos que el divisor de r es la suma formal de los ceros y los polos de la recta en los puntos de la curva elíptica, contados con sus multiplicidades.

  Para determinar los ceros, calculamos la intersección de E y r sustituyendo la ecuación de la recta en E, de modo que obtenemos $x^3 - 4x^2 = 0$. Esta ecuación tiene la raíz doble $x = 0$ y la raíz simple $x = 4$. Sustituyendo estos valores en $E \cap r$, vemos que los ceros son el punto $P_1 = (0, 2)$ con multiplicidad 2 (es decir, r es *tangente* a E en dicho punto) y $P_2 = (4, 3)$ con multiplicidad 1.

  Según el Teorema 4.5, $div(r)$ ha de tener grado 0. Puesto que hay tres ceros, deben existir también tres polos. Visto que r no tiene polos en los puntos afines de E, estos deben estar en el punto del infinito $\mathcal{O} = (0 : 1 : 0)$ de E, que será por lo tanto polo de multiplicidad 3. Por lo tanto:

$$div(r) = 2(P_1) + 1(P_2) - 3(\mathcal{O}).$$

- Sea el divisor $D = 1(P_2) + 1(P_3) - 2(P_4)$, donde $P_3 = (4, 4)$ y $P_4 = (6, 3)$. Este divisor tiene grado $1 + 1 - 2 = 0$. No obstante, podemos comprobar por el Teorema 4.5 que no es principal, puesto que $sum(D) \neq \mathcal{O}$. Veámoslo mediante las herramientas analíticas de la Sección 2.3:

$$sum(D) = P_2 + P_3 - 2P_4 = \mathcal{O} - 2P_4 = 2(6, 4) = (4, 4).$$

4.8 Proposición. *Un divisor D de grado cero admite una expresión en forma canónica:*

$$D = (P) - (\mathcal{O}) + div(f),$$

donde $P \in E$ es único y la función racional f única salvo producto por una constante no nula.

4.9 Definición. Sea $D = \sum_{P \in E} n_P(P)$ un divisor y sea f una función racional tal que D y $div(f)$ tienen soportes disjuntos. Definimos:

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

4.2. El pairing de Weil

4.10 Definición. Sea E una curva elíptica definida sobre un cuerpo finito $\Bbbk = \mathbb{F}_q$ de característica p. Sea m un entero positivo coprimo con p, y sea $\mu_m \subset \overline{\Bbbk}^*$ el grupo de las raíces m-ésimas de la unidad.

Sean $P, Q \in E[m]$. Sean A y B divisores de grado cero con soportes disjuntos tales que:

$$A \sim (P) - (\mathcal{O}), \qquad B \sim (Q) - (\mathcal{O})$$

Sean f_A, f_B funciones racionales sobre E tales que:

$$div(f_A) = mA, \qquad div(f_B) = mB$$

El *pairing de Weil*, e_m, es la función:

$$e_m : E[m] \times E[m] \longrightarrow \mu_m$$
$$e_m(P, Q) = f_A(B)/f_B(A)$$

El valor de $e_m(P, Q)$ es independiente de la elección de A, B, f_A y f_B.

Observación. Notemos que f_A y f_B existen por el Teorema 4.5, ya que tanto P como Q son puntos de m-torsión. Observemos también que $div(f_A)$ y B tienen soportes disjuntos, así como $div(f_B)$ y A, gracias a lo cual $f_A(B)$ y $f_B(A)$ están bien definidos. Por lo tanto e_m está bien definido.

A continuación listamos una serie de propiedades útiles del pairing de Weil, cuya demostración puede encontrarse en [Sil09, Proposición III.8.1]:

1. *Bilineal:* $\forall P, Q, R \in E[m], e_m(P+Q, R) = e_m(P, R)e_m(Q, R)$ y $e_m(P, Q+R) = e_m(P, Q)e_m(P, R)$.
2. *Alternado:* $\forall P \in E[m], e_m(P, P) = 1$.
3. *Antisimétrico:* $\forall P, Q \in E[m], e_m(P, Q) = e_m(Q, P)^{-1}$.
4. *No-degenerado:* Si $e_m(P, Q) = 1$ para todo $Q \in E[m]$, entonces $P = \mathcal{O}$.
5. Si $E[m] \subseteq E(\Bbbk)$, entonces $e_m(P, Q) \in \Bbbk$ para todo $P, Q \in E[m]$ (es decir, $\mu_m \subseteq \Bbbk^*$).
6. *Compatible:* Si $P \in E[m]$ y $Q \in E[mm']$, entonces $e_{mm'}(P, Q) = e_m(P, m'Q)$.

Observación. Nótese que de la bilinealidad de e_m se deduce en particular que $e_m(aP, bQ) = e_m(aP, Q)^b = e_m(baP, Q) = e_m(P, Q)^{ab}$.

Damos a continuación una manera alternativa de evaluar el pairing de Weil, que en su momento veremos nos resultará más práctica.

4.11 Lema. *Sea S un punto cualquiera de la curva elíptica E. Sean $A = (P + S) - (S)$, $B = (Q - S) - (-S)$. Entonces $A \sim (P) - (\mathcal{O})$ y $B \sim (Q) - (\mathcal{O})$.*

Demostración. Efectivamente, $A' = A - (P) - (\mathcal{O}) = (P + S) - (S) - (P) + (\mathcal{O})$ es un divisor principal, debido al Teorema 4.5 puesto que $gr(A') = 1 - 1 - 1 + 1 = 0$ y $sum(A') = P + S - S - P + \mathcal{O} = \mathcal{O}$. Análogamente, $B' = B - (Q) - (\mathcal{O}) = (Q - S) - (-S) - (Q) + (\mathcal{O})$ es un divisor principal, ya que $gr(B') = 1 - 1 - 1 + 1 = 0$ y $sum(B') = Q - S + S - Q + \mathcal{O} = \mathcal{O}$. □

4.12 Teorema. *Sean $P, Q \in E[m]$ y sea $S \in E$ un punto cualquiera satisfaciendo $S \notin \{\mathcal{O}, P, -Q, P - Q\}$. Sean f_P, f_Q funciones racionales con divisores asociados $div(f_P) = m(P + S) - m(S)$, $div(f_Q) = m(Q - S) - m(-S)$. Entonces:*

$$e_m(P, Q) = \frac{f_P(Q + S)/f_P(S)}{f_Q(P - S)/f_Q(-S)}.$$

Demostración. Gracias al Teorema 4.5 podemos garantizar que existen funciones racionales sobre E asociadas a los divisores enunciados, es decir, tales que $div(f_P) = mA, div(f_Q) = mB$ con $A \sim (P) - (\mathcal{O})$ y $B \sim (Q) - (\mathcal{O})$. La equivalencia de los divisores se desprende directamente del lema anterior. Podemos calcular por lo tanto el pairing de Weil como sigue:

$$e_m(P,Q) = \frac{f_P(B)}{f_Q(A)} = \frac{f_P((Q+S)-(S))}{f_Q((P-S)-(-S))} = \frac{f_P(Q+S)/f_P(S)}{f_Q(P-S)/f_Q(-S)}.$$

La última igualdad es consecuencia de la Definición 4.9 y en ella cada función racional es aplicada sobre un punto de la curva elíptica. La no pertenencia de S al conjunto $\{\mathcal{O}, P, -Q, P-Q\}$ garantiza que las funciones racionales están bien definidas para los divisores dados y que no dividimos entre cero. □

Obsérvese que, en el método de evaluación dado en el Teorema 4.12, el valor de $e_m(P,Q)$ es independiente, además de la elección de f_P y f_Q, del punto $S \in E$ escogido. Resaltemos también la diferencia de notación entre la Definición 4.10, donde el subíndice y el elemento a evaluar en las funciones racionales son divisores, respecto al Teorema 4.12, donde tanto el subíndice como el elemento a evaluar de las funciones racionales son puntos de la curva elíptica. Esto es perfectamente coherente dada la definición por extensión que dimos en 4.9. Utilizaremos en adelante, según nos convenga, cualquiera de las dos maneras de calcular el valor del pairing.

4.2.1. El pairing de Weil modificado

La segunda propiedad listada del pairing de Weil (alternada) lo vuelve muy inconveniente para aplicaciones criptográficas, ya que en ellas trabajamos siempre en el grupo engendrado por un punto $P \in E$. Si tomásemos cualquier par de puntos $R, S \in \langle P \rangle$ tendríamos por la propiedad bilineal que $e_m(R,S) = 1$, es decir, que estaríamos ante la aplicación trivial. La solución a este problema es utilizar un pairing modificado $\hat{e}_m$ utilizando lo que llamamos una aplicación distorsión.

En las aplicaciones criptográficas, generalmente, tomamos m primo, así que restringimos nuestra atención a este caso y lo denotaremos ℓ en adelante.

4.13 Definición. Sean $\ell \geqslant 3$ un número primo, E una curva elíptica y $P \in E[\ell]$. Sea $\phi : E \to E$ un endomorfismo de E. Decimos que ϕ es una *aplicación de ℓ-distorsión para P* si cumple las dos siguientes propiedades:

1. $\phi(nP) = n\phi(P)$ para todo $n \geqslant 1$.
2. El número $e_\ell(P, \phi(P))$ es una raíz primitiva ℓ-ésima de la unidad, es decir:

$$e_\ell(P,\phi(P))^r = 1 \iff \ell | r.$$

4.14 Proposición. *Sea $\ell \geqslant 3$ un número primo, sea E una curva elíptica y veamos $E[\ell] = \mathbb{Z}/\ell\mathbb{Z} \times \mathbb{Z}/\ell\mathbb{Z}$ como un espacio vectorial de dimensión dos sobre el cuerpo $\mathbb{Z}/\ell\mathbb{Z}$. Sean $P, Q \in E[\ell]$. Los siguientes enunciados son equivalentes:*

1. *P y Q forman una base del espacio vectorial $E[\ell]$.*

2. *$P \neq \mathcal{O}$ y Q no es múltiplo de P.*

3. *$e_\ell(P, Q)$ es una raíz primitiva ℓ-ésima de la unidad.*

4. *$e_\ell(P, Q) \neq 1$.*

Demostración. Véase [HPS08, Proposición 5.49]. □

4.15 Definición. Sea E una curva elíptica, sea $P \in E[\ell]$ y sea ϕ una aplicación de ℓ-distorsión para P. El *pairing de Weil modificado* $\hat{e}_\ell$ sobre $E[\ell]$ y relativo a ϕ se define como:

$$\hat{e}_\ell(Q, Q') = e_\ell(Q, \phi(Q')).$$

El pairing de Weil modificado $\hat{e}_\ell$ es no-degenerado y sirve por tanto para nuestros propósitos, como prueba la siguiente proposición.

4.16 Proposición. *Sea E una curva elíptica, sea $P \in E[\ell]$, sea ϕ una aplicación de ℓ-distorsión para P y sea $\hat{e}_\ell$ el pairing de Weil modificado relativo a ϕ. Sean Q y Q' múltiplos de P. Entonces:*

$$\hat{e}_\ell(Q, Q') = 1 \iff Q = \mathcal{O} \text{ ó } Q' = \mathcal{O}.$$

Demostración. Como Q y Q' son múltiplos de P, podemos escribirlos de la forma $Q = sP$ y $Q' = tP$. A partir de la definición de $\hat{e}_\ell$, que ϕ es un morfismo y que el pairing de Weil es lineal, obtenemos:

$$\hat{e}_\ell(Q, Q') = \hat{e}_\ell(sP, tP) = e_\ell(sP, \phi(tP)) = e_\ell(sP, t\phi(P)) = e_\ell(P, \phi(P))^{st}.$$

Como $e_\ell(P, \phi(P))$ es una raíz primitiva ℓ-ésima de la unidad, deducimos:

$$\hat{e}_\ell(Q, Q') = 1 \iff \ell | st \iff \ell | s \text{ ó } \ell | t \iff Q = \mathcal{O} \text{ ó } Q' = \mathcal{O}.$$

□

Sin embargo, no siempre es posible construir aplicaciones de distorsión. En [BSS05] se prueba que no existen aplicaciones de distorsión para curvas ordinarias con $k \neq 1$. En tales casos existen otras herramientas alternativas, como las aplicaciones traza, que pueden revisarse en el mismo libro.

4.3. Algoritmo de Miller

En esta sección describiremos un método del tipo *square and multiply* que nos permitirá calcular el pairing de Weil de manera efectiva. La idea central, debida a Victor Miller, es un algoritmo capaz de evaluar rápidamente ciertas funciones con divisores específicos, como las mostradas en el siguiente teorema:

4.17 Teorema. *Sea E una curva elíptica definida sobre el cuerpo finito $\mathbb{F}_q$ de característica $p \geqslant 5$. Sean $P = (x_P, y_P)$ y $Q = (x_Q, y_Q)$ dos puntos distintos de $\mathcal{O}$ pertenecientes a E. Sea λ la pendiente de la recta $(PQ)_E$ (si es una recta vertical, consideramos $\lambda = \infty$). Definimos la función $g_{P,Q}$ sobre E como sigue:*

$$g_{P,Q} = \begin{cases} \frac{y - y_P - \lambda(x - x_P)}{x + x_P + x_Q - \lambda^2} & \text{si } \lambda \neq \infty \\ x - x_P & \text{si } \lambda = \infty \end{cases} \tag{4.1}$$

Entonces:

$$div(g_{P,Q}) = (P) + (Q) - (P + Q) - (\mathcal{O}).$$

Demostración. CASO 1: Supongamos $\lambda \neq \infty$. Sea $y = y_P + \lambda(x - x_P)$ la recta $(PQ)_E$. Dicha recta tiene tres puntos de corte con E, que son P, Q y $-(P + Q)$, luego:

$$div(y - y_P - \lambda(x - x_P)) = (P) + (Q) + (-P - Q) - 3(\mathcal{O})$$

Las rectas verticales cortan a E en pares de puntos opuestos entre sí, así que:

$$div(x - x_{P+Q}) = (P + Q) + (-P - Q) - 2(\mathcal{O})$$

De donde deducimos, puesto que $x_{P+Q} = \lambda^2 - x_P - x_Q$ (véanse las fórmulas de adición de puntos dadas en la Sección 2.3.1), que la función racional

$$g_{P,Q} = \frac{y - y_P - \lambda(x - x_P)}{x + x_P + x_Q - \lambda^2}$$

Tiene el divisor enunciado aplicando que $div(f/g) = div(f) - div(g)$.

CASO 2: Supongamos $\lambda = \infty$. Entonces $P + Q = \mathcal{O}$, luego hay que comprobar que $g_{P,Q}$ tenga divisor $(P) + (-P) - 2(\mathcal{O})$. La función racional $x - x_P$ tiene dicho divisor.

□

4.18 Teorema. *Sea $m \geqslant 1$, que escribimos con su expresión binaria correspondiente:*

$$m = m_0 + m_1 \cdot 2 + m_2 \cdot 2^2 + \cdots + m_{n-1} \cdot 2^{n-1}$$

con $m_i \in \{0, 1\}$ y $m_{n-1} \neq 0$. El siguiente algoritmo devuelve una función f_P cuyo divisor satisface:

$$div(f_P) = m(P) - (mP) - (m - 1)(\mathcal{O})$$

y donde las funciones $g_{T,T}$ y $g_{T,P}$ utilizadas son las definidas arriba:

Algoritmo 2 Algoritmo de Miller

Entrada: $P \in E$.
Salida: $f = f_P$.

1: $T \leftarrow P$
2: $f \leftarrow 1$
3: **para** $i = n - 2$ hasta 0 **hacer**
4: $\quad f \leftarrow f^2 \cdot g_{T,T}$
5: $\quad T \leftarrow 2T$
6: $\quad$ **si** $m_i = 1$ **entonces**
7: $\quad\quad f \leftarrow f \cdot g_{T,P}$
8: $\quad\quad T \leftarrow T + P$
9: $\quad$ **fin si**
10: **fin para**
11: **devolver** f

En particular, si $P \in E[m]$, entonces $div(f_P) = m(P) - m(\mathcal{O})$. Nótese que esta función racional verifica las condiciones dadas para f_A (respectivamente f_B) en la Definición 4.10 o, si se prefiere, las de f_P (respectivamente f_Q) en el Teorema 4.12.

Demostración. Esto no es más que un algoritmo estándar del tipo *square and multiply*, igual en esencia al Algoritmo 1 visto anteriormente. La clave reside en esta ocasión en las funciones dadas en el Teorema 4.17, que nos indica que las funciones $g_{T,T}$ y $g_{T,P}$ tienen divisores

$$div(g_{T,T}) = 2(T) - (2T) - (\mathcal{O}) \quad \text{y} \quad div(g_{T,P}) = (T) + (P) - (T+P) - (\mathcal{O}).$$

Daremos solo un bosquejo de la demostración, que consiste en considerar el efecto de cada iteración i-ésima del bucle *para* `(3:)-(10:)`. Lo que sucede entonces, denotando mediante el superíndice *fin* el valor al concluir una iteración e *ini* al comienzo de ésta, es:

$$\begin{gathered} T_i^{fin} = 2T_i^{ini} + m_i P, \\ f_i^{fin} = (f_i^{ini})^2 \cdot g_{T_i^{ini},T_i^{ini}} \cdot (g_{2T_i^{ini},P})^{m_i} \\ div(f_i^{fin}) = 2div(f_i^{ini}) + 2(T_i^{ini}) - (T_i^{fin}) + m_i(P) - (1+m_i)(\mathcal{O}). \end{gathered}$$

Los valores finales de T y f después de una iteración i-ésima dada son justamente los valores de inicio de la siguiente iteración, es decir, $T_i^{fin} = T_{i-1}^{ini}$ y $f_i^{fin} = f_{i-1}^{ini}$. Se plantean entonces unas ecuaciones en recurrencias para T y $div(f)$, que resultan ser una suma telescópica con la que calculamos T_0^{fin} y, así, $div(f_0^{fin})$. Para un detalle mayor, puede consultarse [Sil09, Teorema XI.8.1]. □

Sean $P, Q \in E[m]$. El algoritmo de Miller nos permite computar una función f_P (respectivamente f_Q) de divisor $div(f_P) = m(P) - m(\mathcal{O})$ (resp. $div(f_Q) = m(Q) - m(\mathcal{O})$). Además, si R es otro punto cualquiera de E, podemos computar $f_P(R)$ directamente si evaluamos las funciones $g_{T,T}(R)$ y $g_{T,P}(R)$ cada vez que ejecutamos los pasos `(4:)` y `(7:)` del algoritmo. Obsérvese que esto nos permitiría evaluar el pairing $e_m(P, Q)$ de forma mucho más rápida, pues tomando un punto $S \notin \{\mathcal{O}, P, -Q, P - Q\}$, R podría ser cualquiera de entre $\{Q + S, S, P - S, -S\}$. De este modo, si en una ejecución del algoritmo se calculan simultáneamente $f_P(Q + S)$ y $f_P(S)$ y en otra ejecución hacemos lo mismo con $f_Q(P - S)$ y $f_Q(-S)$, podemos finalmente evaluar:

$$e_m(P,Q) = \frac{f_P(Q+S)/f_P(S)}{f_Q(P-S)/f_Q(-S)}.$$

4.19 Ejemplo. Sea $E : y^2 = x^3 + 30x + 34$ una curva elíptica sobre $\mathbb{F}_{631}$. La curva tiene $\#E(\mathbb{F}_{631}) = 650 = 2 \cdot 5^2 \cdot 13$ puntos, de los cuales comprobamos que

$$P = (36, 60) \qquad \text{y} \qquad Q = (121, 387)$$

son de orden 5, linealmente independientes y por lo tanto generan $E(\mathbb{F}_{631})[5]$. Por el Teorema 2.20, entonces necesariamente $E(\mathbb{F}_{631}) \cong \mathbb{Z}/5\mathbb{Z} \times \mathbb{Z}/130\mathbb{Z}$. Resulta trivial que el grado de inmersión de $E/\mathbb{F}_{631}$ respecto a 5 es $k = 1$. Para computar el pairing de Weil mediante el algoritmo de Miller, buscamos un punto S que no esté en el subgrupo generado por P y Q. Elegimos $S = (0, 36)$, de orden 130. El algoritmo de Miller nos da, entonces:

$$\frac{f_P(Q+S)}{f_P(S)} = \frac{103}{219} = 473 \in \mathbb{F}_{631}.$$

Invirtiendo los papeles de P y Q y reemplazando S por $-S$, el algoritmo de Miller nos permite calcular también:

$$\frac{f_Q(P-S)}{f_Q(-S)} = \frac{284}{204} = 88 \in \mathbb{F}_{631}.$$

Finalmente, si calculamos el cociente de estos dos valores, obtenemos:

$$e_5(P,Q) = \frac{473}{88} = 242 \in \mathbb{F}_{631}.$$

Si en la misma curva consideramos los puntos $P' = 3P = (617, 5)$ y $Q' = 4Q = (121, 244)$, procediendo del mismo modo obtenemos:

$$\frac{f_{P'}(Q'+S)}{f_{P'}(S)} = \frac{326}{523} = 219 \in \mathbb{F}_{631} \qquad \text{y} \qquad \frac{f_{Q'}(P'-S)}{f_{Q'}(-S)} = \frac{483}{576} = 83 \in \mathbb{F}_{631},$$

valores que nos permiten concluir

$$e_5(P',Q') = \frac{219}{83} = 512 \in \mathbb{F}_{631}.$$

Si operamos, comprobamos que $242^5 \equiv 1 \pmod{631}$ y $512^5 \equiv 1 \pmod{631}$, luego efectivamente tanto $e_5(P,Q)$ como $e_5(P',Q')$ son raíces quintas de la unidad en $\mathbb{F}_{631}$. Finalmente, podemos también verificar la bilinealidad del pairing de Weil:

$$512 = e_5(P',Q') = e_5(3P,4Q) = e_5(P,Q)^{12} = 242^{12} = 512 \in \mathbb{F}_{631}.$$

Capítulo 5

Criptología basada en pairings

El uso de los pairings en criptología estuvo, hasta el año 2000, limitado al de herramienta criptoanalítica contra los protocolos basados en el Problema del Logaritmo Discreto Elíptico. Sin embargo, desde que en dicho año Antoine Joux propusiera su protocolo de acuerdo tripartito de claves, se desarrolló un enorme interés en la construcción de protocolos criptográficos basados en pairings. El ejemplo más espectacular fue el desarrollo de un esquema de criptografía basado en la identidad por parte de Boneh y Franklin en 2001, lo cual se trataba de un problema abierto desde su proposición por Shamir en 1984.

Desde entonces el papel de los pairings en criptografía no hizo sino aumentar su protagonismo, penetrando también en otras áreas como los esquemas de firma. Gracias a ellos y al trabajo de Boneh, Lynn y Shacham, se logró la construcción de firmas cortas capaces de otorgar la misma seguridad que firmas de más del doble de longitud basadas en otros modelos.

A lo largo del capítulo trataremos de revisar todos estos avances, de gran relevancia histórica y práctica, así como un esquema de firma ciega que nos será de utilidad para el protocolo de voto electrónico del Capítulo 6.

5.1. Criptoanálisis basado en pairings

Los pairings permiten un nuevo tipo de ataque al logaritmo discreto elíptico (PLDE), los denominados algoritmos de reducción. En 1993, Menezes, Okamoto y Vanstone demuestran con su ataque cómo trasladar, utilizando el pairing de Weil, el PLDE en una curva $E(\mathbb{F}_q)$ al PLD en una extensión $\mathbb{F}_{q^k}$ de su cuerpo de definición, siendo k es el grado de inmersión correspondiente.

Supongamos la siguiente situación. Sea E una curva elíptica sobre $\mathbb{F}_q$ y sea $P \in E(\mathbb{F}_q)[\ell]$, donde ℓ es un número primo grande. Sea k el grado de inmersión respecto a ℓ y supongamos que conocemos cómo resolver el PLD en el cuerpo $\mathbb{F}_{q^k}$. Sea $Q \in E(\mathbb{F}_q)$ un punto que es múltiplo de P. Entonces el Algoritmo 3, debido a Menezes, Okamoto y Vanstone, resuelve el PLDE para P y Q.

La utilidad del algoritmo MOV depende fuertemente del valor de k. Recordemos que el logaritmo discreto sobre el grupo multiplicativo $\mathbb{F}_{q^k}^*$ es resoluble en tiempo subexponencial mediante el algoritmo *Index Calculus* [HR83], mientras que el PLDE es inmune al mismo, lo que posibilita emplear claves mucho menores. Es por ello que claves de 210 bits en el caso elíptico ofrecen la misma seguridad que claves de 2048 bits en el caso clásico.

Algoritmo 3 Ataque Menezes, Okamoto, Vanstone (MOV)

Entrada: $P, Q \in \langle P \rangle$, $Q \neq 0$.
Salida: n tal que $nP = Q$.

1: Computar el número de puntos $N = \#E(\mathbb{F}_{q^k})$. Existen para ello varios algoritmos de tiempo polinómico [BSS05, Capítulo VI]. Obsérvese que $\ell | N$ por la hipótesis de que $P \in E(\mathbb{F}_q)[\ell]$.
2: Escoger al azar $T \in E(\mathbb{F}_{q^k})$ que verifique $T \notin E(\mathbb{F}_q)$.
3: Computar $T' = (N/\ell)T$.
4: **mientras** $T' = \mathcal{O}$ **hacer**
5: Escoger al azar $T \in E(\mathbb{F}_{q^k})$ que verifique $T \notin E(\mathbb{F}_q)$.
6: Computar $T' = (N/\ell)T$.
7: **fin mientras**
8: $T' \in E[\ell]$. Utilizar un algoritmo como el de Miller (Algoritmo 2) para computar los siguientes valores del pairing de Weil:

$$\alpha = e_\ell(P, T') \in \mathbb{F}_{q^k}^* \quad \text{y} \quad \beta = e_\ell(Q, T') \in \mathbb{F}_{q^k}^*$$

9: Resolver el PLD para α y β en $\mathbb{F}_{q^k}^*$, es decir, encontrar $n \in \mathbb{Z}$ tal que $\beta = \alpha^n$.
10: **devolver** n

Debemos observar, sin embargo, que la longitud binaria de q^k es k veces la longitud binaria de q y por tanto para k grande el tamaño del cuerpo $\mathbb{F}_{q^k}$ es lo suficientemente elevado como para resistir los ataques basados en el *Index Calculus*. No obstante, como ya hemos demostrado, para ciertas curvas elípticas, como las supersingulares, el valor de k es muy reducido, lo que las convierte en vulnerables para criptosistemas basados en el Problema del Logaritmo Discreto Elíptico.

Nota. Un algoritmo de reducción similar, utilizando el pairing de Tate, fue propuesto por Frey y Rück en 1994 [FR94]. En el caso concreto de las curvas anómalas, existen algoritmos específicos de reducción debidos a Satoh y Araki [SA98], Semaev [Sem98] y Smart [Sma99] que resuelven el PLDE en tiempo lineal.

5.2. Criptografía basada en pairings

La bilinealidad de los pairings ofrece numerosas posibilidades para la construcción de aplicaciones criptográficas. Concretamente, la propiedad $e(aP, Q) = e(P, Q)^a = e(P, aQ)$ puede ser usada para transportar el (potencialmente secreto) valor a de una coordenada a la otra, sin la necesidad de conocerlo. En consecuencia, la seguridad de la mayoría de los protocolos basados en pairings depende de un nuevo tipo de problema de Diffie-Hellman, similar a los vistos en la Sección 1.3.3, conocido como Problema Bilineal de Diffie-Hellman.

Definimos este problema, al igual que todos los protocolos de esta sección, en términos del pairing de Weil modificado $\hat{e}_\ell : E[\ell] \times E[\ell] \longrightarrow \mu_\ell$, pero podrían utilizarse alternativamente otros pairings que fuesen no alternados.

5.1 Problema de Bilineal de Diffie-Hellman (PBDH). Para $a, b, c \in \mathbb{Z}_\ell^*$, dados (P, aP, bP, cP), el Problema Bilineal de Diffie-Hellman consiste en computar $\hat{e}_\ell(P, P)^{abc}$. Le asignamos la notación $\mathrm{BDH}_P(aP, bP, cP) = \hat{e}_\ell(P, P)^{abc}$.

Es obvio que el PBDH es a lo sumo tan difícil como el PCDH en $E[\ell]$, ya que teniendo la solución $Q = \mathrm{CDH}_P(aP, bP)$, uno puede computar fácilmente $\hat{e}_\ell(Q, cP) = \hat{e}_\ell(P, P)^{abc}$, o

análogamente para $Q = \mathrm{CDH}_P(aP, cP)$, $Q = \mathrm{CDH}_P(bP, cP)$. Además, la dificultad del PBDH está también limitada por la del PCDH en μ_ℓ. Por ejemplo, si denominamos $g = \hat{e}_\ell(P, P)$, el Problema Computacional de Diffie-Hellman para g, $g^a = \hat{e}_\ell(P, aP)$ y $g^{bc} = \hat{e}_\ell(bP, cP)$ pide encontrar $\mathrm{CDH}_g(g^a, g^{bc}) = g^{abc} = \hat{e}_\ell(P, P)^{abc}$, lo que es justamente la solución al PBDH.

La siguiente imagen muestra la relación entre todos estos problemas y el PLD en un grupo Grieta Diffie-Hellman. La línea de puntos separa los problemas de gran diferencia de dificultad, y A $\longrightarrow$ B significa que el problema A es al menos tan difícil como el problema B. Los subíndices indican el grupo donde se plantea el problema:

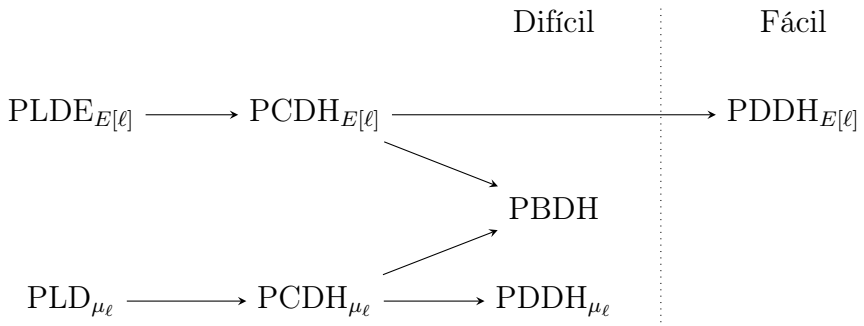

Figura 5.1: PBDH en un Grupo Grieta Diffie-Hellman.

Aunque no se sabe nada sobre la implicación recíproca, es decir, si la dificultad del PCDH en $E[\ell]$ y μ_ℓ implica la dificultad del PBDH, suele considerarse que se da la equivalencia.

5.2.1. Acuerdo tripartito de claves

El siguiente protocolo, propuesto por Antoine Joux en el año 2000 [Jou00], permite en una sola ronda el acuerdo de una clave común entre tres usuarios. Denominémosles A, B y C. El primer paso que han de dar es ponerse de acuerdo en una curva elíptica E y un punto $P \in E(\mathbb{F}_q)[\ell]$ de orden primo tal que exista una aplicación de ℓ-distorsión para P. Denotemos $\hat{e}_\ell$ al pairing de Weil modificado asociado a dicha aplicación de distorsión. El Algoritmo 4 da entonces la forma de obtener una clave común para los tres.

Algoritmo 4 Acuerdo tripartito de clave basado en pairings

1: **Intercambio de puntos:** Los usuarios A, B y C realizan en privado:
2: A escoge secretamente al azar un n_A: $1 < n_A < \ell$, y calcula $P_A = n_A P$.
3: B escoge secretamente al azar un n_B: $1 < n_B < \ell$, y calcula $P_B = n_B P$.
4: C escoge secretamente al azar un n_C: $1 < n_C < \ell$, y calcula $P_C = n_C P$.
5: A, B y C intercambian los valores de P_A, P_B y P_C.
6: **Acuerdo de clave:** Los participantes A, B y C calculan la clave común K_{ABC}:
7: A, utilizando su clave privada n_A y los parámetros públicos P_B y P_C, obtiene:

$$K_{ABC} = \hat{e}_\ell(P_B, P_C)^{n_A} = \hat{e}_\ell(P, P)^{n_A n_B n_C}$$

8: B, utilizando su clave privada n_B y los parámetros públicos P_A y P_C, obtiene:

$$K_{ABC} = \hat{e}_\ell(P_A, P_C)^{n_B} = \hat{e}_\ell(P, P)^{n_A n_B n_C}$$

9: C, utilizando su clave privada n_C y los parámetros públicos P_A y P_B, obtiene:

$$K_{ABC} = \hat{e}_\ell(P_A, P_B)^{n_C} = \hat{e}_\ell(P, P)^{n_A n_B n_C}$$

Si un atacante fuera capaz de resolver el PLDE podría entonces romper el algoritmo de acuerdo tripartito de clave, ya que con recuperar sólo un n_A, n_B o n_C le bastaría para calcular K_{ABC}. Sin embargo, existe un ataque más sencillo. Supongamos que un atacante utiliza los valores públicos P y P_A para calcular los valores:

$$\hat{e}_\ell(P,P), \qquad \hat{e}_\ell(P_A,P) = \hat{e}_\ell(n_A P,P) = \hat{e}_\ell(P,P)^{n_A}.$$

En tal caso, el atacante podría obtener n_A si lograra resolver la ecuación $a^{n_A} = b$, donde $a = \hat{e}_\ell(P,P)$, $b = \hat{e}_\ell(P_A,P)$. En otras palabras, es suficiente resolver el Problema del Logaritmo Discreto en un subgrupo de orden ℓ de $\mathbb{F}_q^*$ para romper el esquema. Como para ello disponemos del ataque Index Calculus, que resuelve dicho problema en tiempo subexponencial [HR83], usar el acuerdo tripartito de clave de manera segura requiere un cuerpo $\mathbb{F}_q$ mucho mayor que el acuerdo bipartito. Hay que destacar, sin embargo, que a la fecha sigue sin conocerse ningún otro método para alcanzar un secreto común entre tres entidades en una sola ronda.

5.2 Ejemplo. Ilustremos el acuerdo tripartito de clave con un ejemplo numérico. Sea la curva elíptica $E : y^2 = x^3 + x$ sobre $\mathbb{F}_{1303}$. Su cardinal es $\#E(\mathbb{F}_{1303}) = 1304 = 2^3 \cdot 163$. El punto $P = (334, 920) \in E(\mathbb{F}_{1303})$ es de 163-torsión. Es sencillo comprobar que el grado de inmersión de $E/\mathbb{F}_{1303}$ respecto a 163 es 2. Los usuarios A, B, C escogen al azar y en secreto:

$$n_A = 126, \quad n_B = 71, \quad n_C = 3.$$

Y computan y publican, respectivamente:

$$P_A = n_A P = (196, 815), \quad P_B = n_B P = (1279, 1171), \quad P_C = n_C P = (872, 515).$$

La aplicación distorsión $\phi : (x, y) \longrightarrow (-x, iy)$ proporciona el pairing de Weil modificado $\hat{e}_{163}$. Finalmente, utilizando cada usuario su clave privada y los puntos públicos:

$$\begin{array}{ll} \text{A computa:} & \hat{e}_{163}(P_B, P_C)^{126} = (282 + 173i)^{126} = 768 + 662i. \\ \text{B computa:} & \hat{e}_{163}(P_A, P_C)^{71} = (172 + 256i)^{71} = 768 + 662i. \\ \text{C computa:} & \hat{e}_{163}(P_A, P_B)^{3} = (1227 + 206i)^{3} = 768 + 662i. \end{array}$$

Su clave común K_{ABC} es 768+662i.

Nota. A lo largo del ejemplo hemos decidido denotar como i al valor de $\sqrt{-1}$ en $\mathbb{F}^*_{1303^2}$.

5.2.2. Criptografía basada en la identidad

Uno de los asuntos que motivaron a Diffie y a Hellman cuando desarrollaron la criptografía de clave pública fue el problema de la gestión y distribución de claves entre los usuarios de los criptosistemas clásicos, de clave privada.

El esquema de intercambio de claves de Diffie-Hellman, y todos los surgidos gracias a la criptografía de clave pública, permitían resolver dicho problema al poder alcanzar una clave común secreta mediante comunicaciones en un canal inseguro. Para ello, era necesario mantener en secreto las claves privadas de los usuarios, mientras que sus claves públicas circulaban por tales canales. Un nuevo problema salió entonces a primera plana: Una clave pública que se diga perteneciente a un usuario A, ¿es realmente perteneciente a él, o a un atacante que se haga pasar por A?

La solución propuesta a este tipo de ataques, llamados de impersonación (o en inglés, *man-in-the-middle*) fue la de garantizar la autenticidad de dichas claves mediante un sistema de protocolos de firma digital, certificados y autoridades de certificación, lo que generó complicadas Infraestructuras de Clave Pública, conocidas como PKI (del inglés *Public Key Infrastructure*).

En 1984, Adi Shamir [Sha85] propuso un paradigma alternativo, en el cual se pudieran obtener de manera segura claves públicas a partir de cualquier cadena de de texto dada, permitiendo así utilizar como clave la propia *identidad* del usuario. Por ejemplo, un usuario cualquiera podría usar su propia dirección de correo electrónico como su clave pública basada en la identidad. De este modo, cualquiera capaz de enviarle un e-mail sería del mismo modo capaz de salvaguardar el contenido de éste de ojos indiscretos y asegurarse de que no ha habido una suplantación de la identidad.

La criptografía basada en la identidad, al igual que las PKI, implica la existencia de una *autoridad de confianza* (AC), también conocida en inglés como *trusted third party* (TTP). Esta autoridad selecciona los parámetros comunes a todos los participantes y se encarga de proporcionarles sus claves privadas. En lo que sigue supondremos que la AC ha seleccionado y hecho públicas, al menos, una curva elíptica $E/\mathbb{F}_q$ un punto $P \in E(\mathbb{F}_q)[\ell]$ de orden primo y un pairing no alternado como el pairing de Weil modificado $\hat{e}_\ell$. Supondremos también que ha tomado aleatoriamente un $s \in \mathbb{Z}_\ell^*$ que le servirá para generar tanto su clave pública P_{AC} como las claves privadas de los participantes.

En [Maa04, Capítulo 8] puede encontrarse un elaborado análisis comparativo entre las Infraestructuras de Clave Pública y los Esquemas Basados en la Identidad, con sus respectivas ventajas e inconvenientes en cada caso.

Acuerdo bipartito de claves basado en la identidad

Adicionalmente a lo expuesto en el preámbulo, para este protocolo suponemos que la AC ha elegido y hechos públicos también una función resumen h que transforma cualquier secuencia binaria en un punto de $\langle P \rangle$. Son por todos conocidos las identidades Id_A, Id_B de los dos usuarios. La clave común entre ambos se obtiene mediante el siguiente algoritmo:

Algoritmo 5 Acuerdo bipartito de claves basado en pairings

1: **Claves privadas de A y B:** Ambas partes solicitan a la AC sus claves, quien:
2: Envía a A el punto $S_A = sP_A \in \langle P \rangle$, donde $P_A = h(Id_A)$.
3: Envía a B el punto $S_B = sP_B \in \langle P \rangle$, donde $P_B = h(Id_B)$.
4: **Acuerdo de clave:** Los participantes A y B calculan la clave común K_{AB}:
5: A, utilizando su clave privada S_A y el parámetro público P_B, obtiene:

$$K_{AB} = \hat{e}_\ell(S_A, P_B) = \hat{e}_\ell(sP_A, P_B) = \hat{e}_\ell(P_A, P_B)^s \in \mu_\ell.$$

6: B, utilizando su clave privada S_B y el parámetro público P_A, obtiene:

$$K_{AB} = \hat{e}_\ell(P_A, S_B) = \hat{e}_\ell(P_A, sP_B) = \hat{e}_\ell(P_A, P_B)^s \in \mu_\ell.$$

Cifrado basado en la identidad

El primer criptosistema eficaz basado en la identidad fue propuesto por Boneh y Franklin en 2001. Fueron, en realidad, dos versiones de uno mismo: El esquema básico expuesto en el

Algoritmo 6 y el esquema completo del Algoritmo 7, pues no se suponía al primero lo suficientemente seguro. La seguridad del modelo completo de Boneh y Franklin se considera equiparable a la del Problema Bilineal de Diffie-Hellman.

Algoritmo 6 Esquema básico de Boneh y Franklin

Parámetros: La autoridad de confianza (AC):
Elige un cuerpo finito $\mathbb{F}_q$, una curva elíptica E y un punto $P \in E(\mathbb{F}_q)[\ell]$ de orden primo de tal forma que exista una aplicación de ℓ-distorsión para P, con $\hat{e}_\ell$ el pairing de Weil modificado asociado.
Publica una función resumen $h_1 : \{Identidades\} \longrightarrow E(\mathbb{F}_q)$ que permite asignar a la identidad de cada usuario A una clave pública $P_A = h_1(Id_A) \in \langle P \rangle$.
Publica una función resumen $h_2 : \mu_\ell \longrightarrow \mathcal{M} = \{0,1\}^n$ que permite asignar a cada elemento de μ_ℓ un mensaje en claro M, que será un elemento del conjunto $\mathcal{M}$ de secuencias binarias de longitud prefijada n.
Elige en secreto un entero s no nulo módulo ℓ y publica el punto $P_{AC} = sP \in E(\mathbb{F}_q)$, que será su clave pública.
Para cada usuario A calcula su correspondiente clave privada $S_A = sP_A \in E(\mathbb{F}_q)$ y se la envía de forma segura.
Cifrado: Si B desea enviar a A un mensaje $M \in \mathcal{M}$:
Computa $P_A = h_1(Id_A) \in \langle P \rangle$.
Elige aleatoriamente $r \not\equiv 0 \pmod{\ell}$ y computa ($\oplus$ indica la suma bit a bit o XOR):

$$C_1 = rP, \qquad C_2 = M \oplus h_2(\hat{e}_\ell(P_A, P_{AC})^r)$$

Envía a A el criptotexto $C = (C_1, C_2)$.
Descifrado: Cuando A recibe el criptotexto $C = (C_1, C_2)$:
Calcula $\hat{e}_\ell(S_A, C_1) = \hat{e}_\ell(sP_A, rP) = \hat{e}_\ell(P_A, P)^{rs} = \hat{e}_\ell(P_A, sP)^r = \hat{e}_\ell(P_A, P_{AC})^r$.
Obtiene el mensaje original M realizando:

$$C_2 \oplus h_2(\hat{e}_\ell(S_A, C_1)) = M \oplus h_2(\hat{e}_\ell(P_A, P_{AC})^r) \oplus h_2(\hat{e}_\ell(P_A, P_{AC})^r) = M.$$

Algoritmo 7 Esquema completo de Boneh y Franklin

Parámetros: Además de los parámetros del esquema básico, la autoridad de confianza:
Publica una función resumen $h_3 : \{0,1\}^{2n} \longrightarrow \{2, 3, ...\ell - 1\}$.
Publica una función resumen $h_4 : \{0,1\}^n \longrightarrow \{0,1\}^n$.
Cifrado: Si B desea enviar a A un mensaje $M \in \mathcal{M}$:
Computa $P_A = h_1(Id_A) \in \langle P \rangle$.
Toma $S \in \{0,1\}^n$ aleatoriamente.
Calcula $r = h_3(S, M)$
Calcula $C = (C_1, C_2, C_3)$ y se lo envía a A, donde:

$$C_1 = rP, \qquad C_2 = S \oplus h_2(\hat{e}_\ell(P_A, P_{AC})^r), \qquad C_3 = M \oplus h_4(S).$$

Descifrado: Cuando A recibe el criptotexto $C = (C_1, C_2, C_3)$:
Calcula $S' = C_2 \oplus h_2(\hat{e}_\ell(S_A, C_1))$
Calcula $M' = C_3 \oplus h_4(S')$.
Calcula $r' = h_3(S', M')$.
si $C_1 = r'P$ **entonces**
    A acepta como válido $M' = M$.
si no
    A rechaza el mensaje recibido.
fin si

5.3. Esquemas de firma basados en pairings

La firma digital de un mensaje M es el análogo criptográfico de la firma ordinaria, con la diferencia de que la firma digital depende del mensaje concreto M. Permite demostrar, incluso con valor legal, la identidad de quien ha producido la firma de M, puesto que en un esquema de firma bien propuesto el intento de falsificar la firma por un adversario es computacionalmente imposible.

En esta sección definiremos distintos conceptos de firmas basadas en pairings, que sirven por lo tanto cada uno a distintos objetivos, junto con sus nociones de seguridad asociadas. Antes de entrar en casos particulares, damos el concepto general:

5.3 Definición. Un esquema de firma S consiste en tres algoritmos $\mathscr{K}, \mathscr{S}$ y $\mathscr{V}$.

- El algoritmo aleatorio de *generación de claves* $\mathscr{K}$ toma cierta información pública I y da como salida un par (sk, pk) de claves privada y pública respectivamente.
- El algoritmo de *generación de firma* $\mathscr{S}$ toma un mensaje M a firmar, la información pública I y la clave secreta sk para dar como salida M junto con su firma σ.
- Un algoritmo determinista de *verificación de firma* $\mathscr{V}$ utiliza la clave pública pk, un mensaje M y su firma σ y da como salida 1 (acepta) si la firma es válida ó 0 (rechaza) en caso contrario. Se le exige a este último algoritmo que $\mathscr{V}(pk, \mathscr{S}(I, sk, M)) = 1\ \forall M \in \mathcal{M}$.

La noción de seguridad más ampliamente aceptada por la comunidad criptográfica para los esquemas de firma es la *seguridad contra falsificación en ataques adaptativos con texto en claro elegido*.

5.4 Definición. Sea S un esquema de firma. Considérese el siguiente desafío donde el retador da al adversario una clave pública pk generada aleatoriamente mediante $\mathscr{K}$. Al adversario se le permite realizar la cantidad que quiera de solicitudes de firma al retador. Una solicitud de firma consiste en un mensaje libremente elegido por el adversario, a la cual el retador responde con la firma de dicho mensaje mediante sk, la clave privada correspondiente a pk. El adversario puede realizar estas solicitudes "adaptativamente", es decir, basándose en las respuestas anteriores. Resuelve el desafío si puede dar un par válido de mensaje y firma para la clave sk, con el requisito de que el mensaje no haya sido una de las solicitudes de firma previas. Un esquema de firma S se dice *seguro contra falsificación en ataques adaptativos con texto en claro elegido* (abreviado en inglés EUF-CMA) si ningún adversario puede resolver el desafío en tiempo polinómico con probabilidad no despreciable.

5.3.1. Esquemas de firma basados en grupos Grieta Diffie-Hellman

Como comentamos en la Sección 1.3.3, la definición de un grupo Grieta Diffie-Hellman sugiere de forma natural la definición de un esquema de firma. Por conveniencia y ya que a la fecha sólo se han logrado construcciones de grupos Grieta Diffie-Hellman mediante el uso de pairings, definimos a continuación dicho esquema usando la notación habitual del grupo de puntos de una curva elíptica (aditiva) en lugar de una notación genérica para grupos abstractos.

5.5 Esquema de firma GS. Sea $(G, +)$ un grupo Grieta Diffie-Hellman de orden ℓ primo con generador P. Sea H una función resumen que transforme mensajes de longitud arbitraria en elementos de G^*. La información pública es $I = (\ell, P, H)$. El esquema de firma GS, que denotaremos $GS[G] = (\mathscr{K}, \mathscr{S}, \mathscr{V})$ es como sigue:

- $\mathscr{K}(I)$: Tomar aleatoriamente $sk = s \in \mathbb{Z}_\ell^*$ y computar $pk = P_{PUB} = sP$. Devolver $\{pk = P_{PUB}, sk = s\}$.
- $\mathscr{S}(I, sk, M)$: Computar $Q = H(M)$ y la firma $\sigma = sQ$. Devolver $\{M, \sigma\}$.
- $\mathscr{V}(I, pk, M, \sigma)$: Computar $Q = H(M)$ y $\mathscr{V}(I, pk, M, \sigma) = \mathrm{DDH}_P(pk, Q, \sigma)$ y aceptar la firma si el resultado es 1. En caso contrario, ésta se rechaza.

La idea principal es que el firmante utiliza su clave privada s para convertir el punto Q resumen del mensaje en una firma σ tal que (P, P_{PUB}, Q, σ) es una cuádrupla válida de Diffie-Hellman. La computación de σ sin el conocimiento de s se reduce a resolver el PCDH, mientras que la verificación de la firma consiste en resolver el PDDH.

Construcción de grupos Grieta Diffie-Hellman mediante pairings

Sea $P \in E(\mathbb{F}_q)$ de orden primo ℓ y sea k el grado de inmersión de $E/\mathbb{F}_q$ respecto a ℓ. Supongamos que $\ell \nmid q - 1$. Un resultado de Balasubramanian y Koblitz [BK98] muestra que $E(\mathbb{F}_{q^k})$ contiene un punto Q que es linealmente independiente de P. Tal punto $Q \in E(\mathbb{F}_{q^k})$, también de orden ℓ, puede encontrarse de manera efectiva, y su independencia lineal con respecto a P puede ser verificada mediante el pairing de Weil como describimos en la Proposición 4.14.

Nos encontramos ahora con todas las herramientas necesarias para construir un oráculo de Decisión de Diffie-Hellman. Con los puntos P y Q descritos, el pairing de Weil nos permite determinar si la cuádrupla (P, aP, Q, bQ) es tal que $a \equiv b \pmod{\ell}$, de hecho:

$$a \equiv b \pmod{\ell} \iff e_\ell(P, bQ) = e_\ell(aP, Q)$$

Si además suponemos que existe ϕ aplicación de ℓ-distorsión para P, podemos construir el pairing de Weil modificado $\hat{e}_\ell$, que nos permite determinar si (P, aP, bP, cP) es una cuádrupla de Diffie-Hellman válida, es decir, si es tal que $ab \equiv c \pmod{\ell}$:

$$ab \equiv c \pmod{\ell} \iff \hat{e}_\ell(P, cP) = \hat{e}_\ell(aP, bP)$$

El grupo Grieta de Diffie-Hellman es $\langle P \rangle$ y su oráculo de Decisión de Diffie-Hellman es justamente el pairing de Weil modificado $\hat{e}_\ell$ relativo a ϕ. Observemos que el algoritmo utilizado requiere de dos evaluaciones del pairing de Weil para puntos de $E(\mathbb{F}_{q^k})$. Debido a los ataques de reducción descritos previamente, sabemos que k no puede ser muy pequeño si queremos que el PCDH sea difícil en $\langle P \rangle$. Sin embargo, para que el algoritmo del oráculo de Decisión de Diffie-Hellman sea eficiente, necesitamos que k no sea demasiado grande.

5.3.2. Firmas cortas: La firma de Boneh-Lynn-Shacham

Boneh, Lynn y Shacham [BLS01] observaron que una firma GS es un único elemento de G^* y se dedicaron por lo tanto a la búsqueda de grupos Grieta Diffie-Hellman en los cuales los elementos tuvieran una representación corta. Se debe a ellos la construcción recién expuesta, para la cual utilizaron el pairing de Weil modificado $\hat{e}_\ell$ en una familia de curvas supersingulares con grado de inmersión $k = 6$. De este modo, lograron firmas de la mitad de longitud que las utilizadas habitualmente hasta entonces (DSA, ECDSA) capaces de proporcionar la misma seguridad. Destacaron además que cuando curvas con un grado de inmersión mayor fueran utilizadas, la longitud de la firma podría reducirse aún más manteniendo dichos niveles de seguridad. Ello conllevaría forzosamente el uso de curvas ordinarias, pues las supersingulares tienen un grado de inmersión máximo de seis. Sin embargo, como comentamos en la Sección 4.2.1, para tales curvas ordinarias no existen aplicaciones de distorsión y habría que utilizar un pairing adecuado que nos permitiera la construcción de un Grupo Grieta Diffie-Hellman.

La firma corta de Boneh-Lynn-Shacham (conocida como firma BLS) consiste pues en aplicar el esquema de firma GS 5.5 a una curva elíptica adecuada como las descritas en [BLS01]. Su seguridad está basada en el ya mencionado modelo del oráculo aleatorio [BR93]. La demostración del siguiente teorema, ajena a las líneas generales de este trabajo, puede asimismo encontrarse en [BLS01].

5.6 Teorema. *Sea H una función resumen que actúe como un oráculo aleatorio. Supongamos que un algoritmo $\mathcal{A}$ de un adversario se ejecuta en un tiempo máximo t, realiza como máximo q_S solicitudes de firma al oráculo de firma S y como máximo q_H solicitudes a la función resumen. Supongamos que $\mathcal{A}$ resuelve con éxito el EUF-CMA (Definición 5.4) con probabilidad ϵ no despreciable. Entonces existe un algoritmo $\mathcal{B}$ que resuelve el PCDH en G con probabilidad al menos $\epsilon/(2q_S e)$ en un tiempo de ejecución de como máximo $t + 2(q_H + q_S)c_{\mathcal{A}} log(\ell)$, donde $c_{\mathcal{A}}$ es una constante pequeña (en la práctica $\leqslant 2$) y e la base del logaritmo natural.*

Hemos decidido, en esta ocasión, enunciar el teorema de un modo que si bien a priori puede parecer más oscuro, permite comprender el porqué de la definición de los Problemas de Diffie-Hellman y los modelos e hipótesis teóricos de seguridad. Esta subdisciplina de la criptografía, conocida como *seguridad demostrable*, trata de explicar los distintos protocolos criptográficos reduciendo los problemas en los que basan su seguridad a su forma más básica posible. Se puede entonces explicar el sistema completo en función a esos problemas básicos, relacionándolos entre ellos.

Lo que, en resumen, quiere decir el último resultado, es que el esquema de firma es seguro contra falsificación en ataques adaptativos con texto en claro elegido (EUF-CMA), ya que de resolverse el desafío de dicho modelo en tiempo polinómico con probabilidad no despreciable, entonces el PCDH se resolvería asimismo en tiempo polinómico con probabilidad no despreciable, lo cual se asume generalmente falso como hipótesis. En la siguiente subsección, en la que hablaremos sobre firmas ciegas, enunciaremos el teorema referente a su seguridad de manera más sencilla.

5.3.3. Firmas ciegas: La firma ciega de Boldyreva

Las firmas ciegas son una de las herramientas básicas utilizadas en ciertos esquemas de dinero electrónico y protocolos de voto electrónico. Su objetivo es el de permitir a un *usuario*

conseguir una firma de un *firmante* (por ejemplo, una autoridad de confianza) de tal forma que quien firma no obtenga ninguna información sobre el mensaje que acaba de firmar y que el usuario no pueda obtener más de una firma válida tras una interacción con el firmante.

Un esquema de firma ciega $CS = (\mathscr{CK}, \mathscr{CS}, \mathscr{CV})$ utiliza los mismos algoritmos $\mathscr{CK}$ de generación de claves y $\mathscr{CV}$ de verificación de firma que su esquema de firma regular correspondiente. La diferencia radica en que el *algoritmo de generación de firma ciega* $\mathscr{CS}$ es un protocolo interactivo entre el usuario y el firmante, de los cuales el primero conoce la clave pública (del firmante), el segundo la clave privada correspondiente y todos conocen la información pública. Al final del protocolo, el usuario obtiene un par válido (M, σ) de mensaje y firma, es decir, tal que $\mathscr{CV}(I, pk, M, \sigma) = 1$.

Describimos a continuación el esquema de firma ciega de Boldyreva [Bol02], basado en el esquema de firma GS 5.5. Una vez más, nos reduciremos a la notación de grupo aditivo correspondiente a las curvas elípticas:

5.7 CGS, el esquema de firma ciega GS. Sea $(G, +)$ un grupo Grieta Diffie-Hellman de orden ℓ primo con generador P. Sea H una función resumen que transforme mensajes de longitud arbitraria en elementos de G^*. La información pública es $I = (\ell, P, H)$. Sea $GS[G]$ el esquema de firma GS 5.5. El esquema de firma ciega GS, $CGS[G] = (\mathscr{CK}, \mathscr{CS}, \mathscr{CV})$ es como sigue:

- $\mathscr{K}(I)$: Tomar aleatoriamente $sk = s \in \mathbb{Z}_\ell^*$ y computar $pk = P_{PUB} = sP$. Devolver $\{pk = P_{PUB}, sk = s\}$.
- El protocolo interactivo $\mathscr{CS}$ se define como sigue: El usuario dispone de una clave pública $pk = P_{PUB}$ y de la información pública I. Para "cegar" su mensaje $Q = H(M)$, el usuario toma al azar un $r \in \mathbb{Z}_\ell^*$, computa $\bar{Q} = Q + rP$ y se lo envía al firmante, que conoce I y la clave privada $sk = s$. El firmante computa $\bar{\sigma} = s\bar{M}$ y se lo envía al usuario. Este último computa $\sigma = \bar{\sigma} - rP_{PUB}$ y muestra como salida el par $\{M, \sigma\}$.
- $\mathscr{V}(I, pk, M, \sigma)$: Computar $Q = H(M)$ y $\mathscr{V}(I, pk, M, \sigma) = \mathrm{DDH}_P(pk, Q, \sigma)$ y aceptar la firma si el resultado es 1. En caso contrario, ésta se rechaza.

Nota. Observemos que el par (M, σ) obtenido en $\mathscr{CS}$ es efectivamente una firma válida para M: $\sigma = \bar{\sigma} - rP_{PUB} = s\bar{Q} - rsP = s(Q + rP) - rsP = sQ$

La noción de seguridad en las firmas ciegas hace referencia a dos propiedades. La primera es la "ceguera", es decir, la cualidad de que el firmante no obtenga ninguna información sobre el mensaje del que se obtiene la firma. El esquema CGS verifica esta propiedad, ya que el firmante recibe solo elementos aleatorios de G que son independientes de las salidas mostradas por el usuario. El segundo concepto es una forma especial de seguridad contra la falsificación que exige que un usuario que haya estado involucrado en x rondas del protocolo de firma ciega no sea capaz de producir más de x firmas. La noción estándar de *seguridad contra falsificación en ataques adaptativos con texto en claro elegido* (Definición 5.4) no puede ser utilizada para los esquemas de firma ciega, ya que por su propia construcción, el usuario tiene que ser capaz de producir una firma válida (σ a partir de $\bar{\sigma}$) para un mensaje no firmado previamente. No obstante, una sutil modificación nos permitirá definir el concepto adecuado:

5.8 Definición. Sea S un esquema de firma y sea $CS = (\mathscr{CK}, \mathscr{CS}, \mathscr{CV})$ el correspondiente esquema de firma ciega. Consideremos el siguiente desafío, donde el retador otorga al adversario una clave pública aleatoria generada mediante $\mathscr{CK}$. Al adversario se le permite jugar el rol de

un usuario durante múltiples rondas del protocolo de firma ciega $\mathscr{CS}$. El adversario resuelve con éxito el desafío si es capaz de dar como salida un conjunto L de pares mensaje-firma válidos tales que el número de veces que ha utilizado el protocolo $\mathscr{CS}$ es estrictamente menor que el cardinal de L. Un esquema de firma ciega CS se dice *seguro interactivamente contra falsificación en ataques adaptativos con texto en claro elegido* (en inglés, *one-more forgery under chosen message attacks* [PS00]) si ningún adversario puede resolver el desafío en tiempo polinómico con probabilidad no despreciable.

La seguridad de la firma CGS, dada su naturaleza, tampoco puede basarse en el modelo del oraculo aleatorio. En vez de ello lo hace en la *asunción de objetivos elegidos para el PCDH*, cuya definición puede consultarse en [Bol02, Sección 5]. La demostración del resultado que exponemos a continuación puede encontrarse en el mismo lugar.

5.9 Teorema. *Supongamos que la asunción de objetivos elegidos para el PCDH es cierta en G. Entonces el esquema de firma ciega $CGS[G]$ es seguro interactivamente contra falsificación en ataques adaptativos con texto en claro elegido*

Capítulo 6

Un protocolo de voto electrónico basado en pairings

Los procesos de toma de decisiones son un asunto problemático sobre el que se ha escrito mucho y desde muchos enfoques, incluido el matemático en muchas de sus vertientes. El objetivo de este capítulo no es el de aportar todas estas visiones, sino el de limitarnos a mostrar cómo la criptografía puede ser una herramienta interesante para resolver los problemas de los que adolecen dichos procesos, en el contexto de votaciones de un proceso electoral.

Merece ser destacado que el texto que sigue no pretende ser un ataque ni una defensa de unos métodos u otros, sino un acercamiento crítico a la cuestión que sirva al lector poco familiarizado con ella para entender un poco mejor el debate sobre el voto electrónico: cómo funciona, por qué se introduce y qué problemas trata de resolver.

A tales efectos, haremos en primer lugar un repaso histórico a nivel nacional sobre algunas experiencias relevantes de voto electrónico, para contextualizar el problema y reflejar la tendencia actual hacia sistemas de tal naturaleza. A continuación, enumeraremos las características más ampliamente consensuadas que ha de poseer un sistema de votación de cualquier naturaleza para poder, en función de ellas, evaluar los sistemas de voto electrónico en contraposición a los sistemas de papel a los que estamos más acostumbrados en España.

Para finalizar, estudiaremos un protocolo de voto electrónico de publicación reciente, obra de López García, Domínguez Pérez y Rodríguez Henríquez [LGDPRH13], que servirá como aplicación final y concreta de gran parte de los conceptos y herramientas que hemos expuesto a lo largo del Trabajo de Fin de Grado.

6.1. Evolución histórica

El objetivo inicial de la introducción de tecnologías más sofisticadas, ya fueran mecánicas o electrónicas, en los sistemas de voto habituales, fue el de tratar de automatizar los diferentes procesos de las elecciones a fin de lograr una mayor eficiencia que aumentase la precisión y ahorrase costes.

Los primeros sistemas automatizados han estado enfocados al escrutinio de los votos. Sin embargo, cada vez más, se están utilizando herramientas electrónicas en el resto de las fases de

un sistema electoral, por ejemplo para el registro de los votantes o para la fase de votación. A continuación mostramos una breve reseña de la evolución de los sistemas de voto electrónico a nivel estatal, con algunos casos especialmente relevantes. En [MR+09] puede encontrarse información sobre algunos de los países en los que ha habido experiencias de voto electrónico.

La lista general incluye, aunque no se limita, a: Alemania, Australia, Bélgica, Brasil, Canadá, Estados Unidos, Estonia, Filipinas, Finlandia, Francia, Grecia, India, Irlanda, Israel, Italia, Kazajistán, Noruega, Reino Unido, Rumanía y Suiza.

6.1.1. Experiencias en España

En noviembre del 2003, más de 23.000 catalanes residentes en el extranjero fueron invitados a participar en una experiencia piloto de voto electrónico [BR04a], en paralelo a las elecciones del parlamento de Cataluña. Los votantes podían votar de manera vinculante exclusivamente por correo postal, no obstante, el material electoral destinado a parte de los residentes en México no pudo ser entregado a tiempo debido a problemas con el servicio postal. Por esta razón, la participación de voto por Internet en dicho país excedió en más de un 200 % al número de votos postales enviados.

El Ayuntamiento de Madrid organizó en junio de 2004 una consulta ciudadana destinada a probar diferentes aspectos del voto electrónico remoto [BR04b]. Para este propósito, se escogió un censo de aproximadamente 100.000 personas que tuvieron la oportunidad de participar usando su teléfono móvil o internet como canales de votación. El número de personas que se registraron para participar en el proceso fue de 1.351 y el de votos emitidos fue de 882.

Posteriormente, entre el 1 y el 18 de febrero del 2005, cerca de dos millones de votantes provenientes de un municipio de cada una de las 52 provincias tuvieron la oportunidad de participar en un ensayo no vinculante de voto por Internet [Ele05]. La experiencia se realizó en paralelo a un sistema convencional de voto vinculante basado en papel para un referéndum de la Constitución Europea. De acuerdo a informes de prensa, 10.543 votantes enviaron su voto por Internet.

En 2010, el ayuntamiento de Barcelona sometió a consulta la modificación de la Avenida Diagonal, con una escasa participación que defendió además el mantenimiento de su configuración en dicha fecha.

En los primeros meses de 2014, dos hechos pueden ser tomados como indicadores de un considerable aumento de la confianza por parte de la ciudadanía española en los medios de voto electrónico, ambos dentro del contexto de las Elecciones al Parlamento Europeo.

El primero de ellos es la demanda por parte de colectivos de emigrantes de una transición al voto electrónico, debido a los problemas con el sistema actual de voto postal para electores residentes en el extranjero. Según datos del Instituto Nacional de Estadística, sólo 81.039 solicitudes de voto fueron aceptadas del total de 1.691.367 electores censados [CE14]. En segundo lugar, múltiples formaciones políticas, algunas de las cuales lograron representación en el Parlamento Europeo, abogaron por la realización de elecciones primarias abiertas mediante sistemas de voto electrónico. Según informes de prensa, al menos un total de 56.014 votos individuales fueron emitidos para la totalidad de dichas formaciones.

6.2. Requisitos de un sistema de votación

De forma general, un sistema de voto electrónico tiene que satisfacer las siguientes condiciones para ser considerado como seguro:

1. *Integridad:* El resultado del voto debe satisfacer la intención de voto del votante. Concretamente, el voto pretendido, el emitido y el depositado han de coincidir y éste último ha de contarse de forma precisa en el recuento final.
2. *Secreto del voto:* Nadie puede ser capaz de asociar a un votante con un voto, tanto durante como después de la votación. Ni siquiera el propio votante, si trata de demostrar su voto a otra persona (*no-coerción*).
3. *Autenticación del votante:* Solamente los votantes autorizados pueden emitir votos, y cada uno de ellos puede votar sólo el número de veces correspondiente.
4. *Accesibilidad:* Todos los votantes autorizados tienen que tener una oportunidad real de votar (colectivos con discapacidad, votantes desplazados, etc.).

Aunque la última propiedad pudiera parecer a priori ajena a la categoría de seguridad, desalentar o en última instancia imposibilitar el ejercimiento del derecho a voto a determinadas personas, poseedoras de una intención de voto distinta a la del atacante, es un tipo de ataque real. En entornos virtuales, por ejemplo, hay que estar prevenidos contra ataques de denegación de servicios (DoS, de las siglas en inglés *Denial of Service*).

Otras propiedades, no relacionadas con la seguridad pero también relevantes, son:

5. *Disponibilidad:* El sistema tiene que ser capaz de aceptar todos los votos en el calendario estipulado y proceder a su resolución en un tiempo adecuado.
6. *Inteligibilidad:* La manera en que se tratan y se recuentan los votos debe ser comprensible y perceptible como segura para todos los votantes.

Cualquier intento de ataque a una de estas seis propiedades suele clasificarse según dos criterios simultáneamente. El primero es si es producido por la autoridad de la elección o por personas ajenas a ella. El segundo, si modifica algún voto individual o se trata de un ataque a gran escala que modifique directamente el resultado global.

La satisfacción, ya sea plena o en buena medida, de las seis características, resulta un interesante reto para los diseñadores de sistemas de votación debido a las tensiones inherentes que surgen entre algunas de ellas. Es el caso por ejemplo de la Autenticación y la Accesibilidad: Solicitar a los electores llevar consigo originales y fotocopias de su DNI y Pasaporte, además una muestra de ADN a cotejar *in situ* aumentaría notablemente la Autenticación, pero reduciría la Accesibilidad debido a la cantidad de tiempo y esfuerzo que costarían tanto al elector como a la autoridad de la elección.

Para cumplir con la mayoría estos requisitos se han propuesto muchos protocolos de voto electrónico. En la literatura encontramos principalmente cuatro clases de soluciones: Los protocolos basados en papeletas precifradas, los esquemas basados en mix-nets, los protocolos basados en funciones homomórficas y los protocolos basados en firmas ciegas. El esquema que expondremos en la Sección 6.4 corresponde a esta última categoría. En [MR+09] se puede encontrar un análisis comparativo de los cuatro tipos de soluciones.

6.3. Comparación con otros sistemas

Si bien es cierto que en múltiples ocasiones los sistemas de voto electrónico propuestos en el pasado han sufrido serias dificultades para satisfacer algunos de los requisitos arriba expuestos, los sistemas contemporáneos también representan diversos riesgos. Para ser justos, mostramos a continuación algunos de dichos inconvenientes en los sistemas de voto presencial y voto remoto habituales.

6.3.1. Voto Presencial: Voto en papel

En un entorno de voto presencial clásico, el votante deposita su papeleta en una urna física. A partir de ese momento, el votante debe confiar en que su voto será incluido en el escrutinio ya que no existe la manera de verificarlo, que sí permiten algunas herramientas criptográficas. La autoridad de la elección usualmente publica los resultados locales por recinto, sin embargo dicha información no le confirma al votante que su voto ha sido contado, o que no ha sido manipulado antes del escrutinio. Nos encontramos, por lo tanto, ante un problema de Integridad/Inteligibilidad. Esta carencia trata de suplirse parcialmente otorgando a los ciudadanos la posibilidad de acudir durante el recuento para controlar cualquier irregularidad, lo cual puede resultar poco accesible para determinados colectivos y/o en diversas zonas. También existen en ocasiones ataques directos, como el que mostramos a continuación.

6.1 Ejemplo. *Cadena de votos.* Los ataque de coerción o venta de votos podrían pensarse exclusivos de los entornos remotos, pero no lo son. En entornos presenciales, un ataque de cadena de votos como el descrito en [Jon05], requiere únicamente de la obtención de una papeleta de votación en blanco por parte del atacante. Dicha papeleta es marcada con las opciones de voto deseadas por el atacante, que la entrega a un votante coaccionado o que desea vender su voto. El votante entra en el recinto de votación ocultando la papeleta. Una vez que un oficial de la elección entrega al votante una papeleta en blanco, el votante accede a la cabina de votación y de manera privada intercambia las papeletas: Deposita en la urna la que le fue entregada por el atacante y sale del recinto de votación ocultando la papeleta en blanco. Esta papeleta en blanco será la prueba ante el atacante de que la papeleta depositada en la urna ha sido la encomendada. El atacante tiene entonces una nueva papeleta en blanco, por lo que puede seguir llevando el ataque con el mismo procedimiento tantas veces como personas coaccionadas o vendedoras de su voto tenga disponibles.

6.3.2. Voto Remoto: Voto postal

En el voto postal, los votantes usualmente introducen su voto en un sobre, que es a su vez introducido en un segundo sobre junto con un certificado del votante que contiene los datos de autenticación. En ocasiones el votante debe incluir su firma manuscrita como prueba de identificación. El segundo sobre es enviado a la autoridad de la elección a través de correo postal.

Tal como se puede ver en algunas experiencias descritas en [AHR07], [Uhl05] y los casos ya comentados en la Sección 6.1.1, en el voto postal existe el riesgo de retrasos tanto en el material que se envía al votante como en el voto enviado por el votante a la autoridad de la elección. Estos retrasos se deben principalmente a problemas en el servicio postal o una mala actuación por parte de la autoridad de la elección. Si el material de votación no llega a tiempo

al votante, éste no tendrá la opción de enviar su voto y posiblemente sea tarde para llevar a cabo su votación de otra manera. Si el retraso se presenta en la entrega de la papeleta a la autoridad electoral, dicha papeleta no será incluida en el escrutinio de los votos. El problema de Accesibilidad resulta por lo tanto grave y más común de lo deseado. En la búsqueda de soluciones, se ha propuesto la utilización de códigos de seguimiento para verificar si el voto ha llegado la autoridad de la elección, lo cual sigue sin solucionar los problemas logísticos.

Finalmente, además de los problemas ya mencionados, los votos enviados por correo postal están expuestos a manipulaciones durante su transporte, pudiendo ser modificados o incluso eliminados por adversarios que logren tener acceso a ellos.

6.4. El protocolo de voto electrónico LDR

Lo que sigue es una descripción del esquema de voto electrónico propuesto por López García, Domínguez Pérez y Rodríguez Henríquez [LGDPRH13], al que denotaremos por las siglas de sus primeros apellidos. La primera tarea cuando se diseña un sistema de voto electrónico es establecer el número de entidades que requiere el sistema. A pesar de que un único servidor sería a priori suficiente para encargarse de todo el proceso, surge el problema de que éste debería autenticar al votante antes de emitir su voto, lo cual dificulta el requisito del secreto de voto. Por ello, los autores deciden emular el comité electoral mediante dos autoridades: El Servidor de Autenticación (AS) y el Servidor de Votación (VS).

Además de otras herramientas ya vistas a lo largo del Trabajo de Fin de Grado, los autores utilizan un tipo de pairing concreto, conocido como *optimal ate pairing*, por motivos de eficiencia computacional [BGDM$^+$10]. Este pairing $\tilde{e} : G_2 \times G_1 \to \mu_\ell$, opera sobre curvas elípticas ordinarias enviando dos puntos racionales linealmente independientes pertenecientes a grupos G_1, G_2 de orden ℓ al grupo de raíces primitivas ℓ-ésimas de la unidad.

La familia de curvas elípticas ordinarias que utilizan (y sobre las que actúa dicho pairing, por lo tanto) son las conocidas como curvas de Barreto-Naehrig o curvas BN [BN06]. Las curvas BN sobre un cuerpo primo $\mathbb{F}_p$ vienen definidas por la ecuación $E : y^2 = x^3 + b$, donde $b \neq 0$, y tienen grado de inmersión $k = 12$. La característica p, el orden del grupo ℓ y la traza de Frobenius t_ℓ de tales curvas viene parametrizado por:

$$\begin{aligned} p(\delta) &= 36\delta^4 + 36\delta^3 + 24\delta^2 + 6\delta + 1, \\ \ell(\delta) &= 36\delta^4 + 36\delta^3 + 18\delta^2 + 6\delta + 1, \\ t_\ell(\delta) &= 6\delta^2 + 1, \end{aligned} \tag{6.1}$$

donde $\delta \in \mathbb{Z}$ es un entero de tal forma que los ℓ, p definidos según la ecuación (6.1) sean números primos.

6.4.1. Proceso electoral

El protocolo entre el votante y las autoridades electorales a lo largo de cada fase de la elección viene explicado a continuación, y resumido en los algoritmos que acompañan al texto. La siguiente tabla servirá como apoyo a la lectura en caso de olvidar la notación utilizada.

Notación	
E :	Una curva elíptica de Barreto-Naehrig.
$\{G_1, G_2\}$:	Grupos generados por $P, Q \in E(\mathbb{F}_q)[\ell]$ respectivamente, con P linealmente independiente de Q.
$H_1 : \{0,1\}^* \to G_1$:	Función resumen *map-to-point*.
$m2s : G_2 \to \{0,1\}^{4r}$:	Función que envía un punto de G_2 a una cadena de bits de longitud $4r$, donde r es el tamaño de ℓ en bits.
$\tilde{e} : G_2 \times G_1 \to \mu_\ell$:	*Optimal ate pairing*.
$\{d_{AS}, V_{AS}\}$:	Par de claves privada-pública del AS.
$\{d_{VS}, V_{VS}\}$:	Par de claves privada-pública del VS.
$\{ID_V, d_V, V_V\}$:	Identidad electoral y par de claves privada-pública del Votante.
t :	Una marca de tiempo.
$\{d_t, V_t\}$:	Par de claves pseudónimas privada-pública del Votante.
$b \in \mathbb{Z}_\ell^*$:	Factor aleatorio de cegado.
$m\|\|m'$:	Concatenación de las cadenas de texto m y m'.

Fase de registro

Antes de comenzar el intercambio de mensajes entre el votante y las entidades electorales es necesario realizar un listado nominal que permita realizar la Autenticación de los votantes. Tal registro, que debe contener todos los votantes válidos junto con su identidad electoral, recibe el nombre de *censo electoral*.

No daremos, al igual que tampoco lo hacen los autores en el artículo, un procedimiento específico para realizar este censo electoral, que puede realizarse mediante una Infraestructura de Clave Pública (PKI) o un Esquema Basado en la Identidad. Cada votante ha de poseer finalmente una clave privada d_V de su conocimiento exclusivo y hacer conjuntamente públicas su clave pública V_V y su identidad electoral ID_V.

Adicionalmente, los Servidores de Autenticación y Votación calculan sus respectivas claves privadas d_{AS}, d_{VS} y publican sendas claves públicas V_{AS}, V_{VS}.

Fase de autenticación

La Integridad del voto se consigue a traves de firmas digitales. No obstante, si los votantes usasen sus claves privadas para firmar perderían automáticamente su anonimato, ya que la autoridad electoral tendría que usar su clave pública correspondiente para verificar la firma.

Para evitar este problema, la solución de la propuesta pasa por generar aleatoriamente unas claves *pseudónimas* (d_t, V_t), que llamaremos clave *pseudoprivada* y clave *pseudopública* respectivamente. La generación de estas claves permite desvincular la identidad del votante de la firma en que se utilicen. El problema que surge entonces es el de evitar que un votante malicioso genere varios pseudónimos para votar en múltiples ocasiones. Cuando cada votante genera su firma pseudopública, le aplica un cegado para que el Servidor de Autenticación la firme. Cada votante produce, pues, un par (d_t, V_t) donde V_t es el "mensaje" al que aplicar la firma ciega de Boldyreva, que vimos en la Sección 5.3.3. El motivo de la utilización de una marca de tiempo t por todas las partes es el de evitar los conocidos como *ataques replay* o de

reinyección, que podrían resultar en una suplantación de identidad o ataque de denegación de servicios.

Antes de la generación de la firma ciega, el AS debe autenticar al votante, verificar si se encuentra en el censo electoral y si ninguna papeleta ha sido generada previamente para él. Si todas las verificaciones son correctas, el AS aplicará entonces una única firma ciega para V_t, tras la cual marcará al votante en la lista para evitar que solicite la firma de ninguna otra clave pseudopública. El votante pasa a poseer entonces una papeleta electrónica "en blanco", que consiste únicamente en su clave pseudopública V_t junto con su correspondiente firma ciega S_{V_t}.

Algoritmo 8 LDR - Fase de Autenticación

1: El votante genera aleatoriamente $b, d_t \in \mathbb{Z}_\ell^*$.
2: El votante genera su clave pseudopública, $V_t = d_t Q \in G_2$.
3: El votante comienza el protocolo interactivo de la firma ciega de Boldyreva:
4: $m = m2s(V_t) \in \{0,1\}^{4r}$.
5: $\tilde{M} = bH_1(m) \in G_1$.
6: $S_{\tilde{M}} = d_V \tilde{M} \in G_1$.
7: El votante envía $\{ID_V, t, \tilde{M}, S_{\tilde{M}}\}$ al Servidor de Autenticación.
8: **si** $\tilde{e}(Q, S_{\tilde{M}}) = \tilde{e}(V_V, \tilde{M})$ **entonces**
9: El Servidor de Autenticación firma: $\tilde{S} = d_{AS}\tilde{M} \in G_1$.
10: El Servidor de Autenticación envía $\{t, \tilde{S}\}$ al votante.
11: **si no**
12: Se rechaza la firma por no ser un votante válido.
13: **fin si**
14: El votante obtiene la firma ciega de su clave pseudopública: $S_{V_t} = b^{-1}\tilde{S} \in G_1$.

Fase de votación

El votante genera una firma corta para su voto $\mathcal{V}$ utilizando su clave pseudoprivada d_t. Sigue para ello el procedimiento de la firma corta BLS que describimos en la Sección 5.3.2. Cómo conseguir un Grupo Grieta Diffie-Hellman en las curvas Barreto-Naehrig (que son ordinarias y no disponen por lo tanto de aplicación distorsión) es una tarea compleja en la que no profundizaremos, pero puede consultarse en [BN06]. La papeleta electrónica final $\mathcal{B}$ consiste en V_t junto con su firma ciega S_{V_t} y el voto $\mathcal{V}$ junto con su firma corta $S_{\mathcal{V}}$. El votante envía la papeleta $\mathcal{B}$ al servidor de verificación, que realiza dos verificaciones: En primer lugar, si la firma ciega de V_t es de la autoría del AS, para lo cual se vale de la clave pública V_{AS}. En segundo lugar, si la firma corta de $\mathcal{V}$ es válida, utilizando V_t como la firma pública correspondiente. Si ambas verificaciones son correctas, el Servidor de Verificación genera un valor aleatorio a, que concatena a la información de la papeleta para luego aplicarle una función resumen. Firma el resultado de esta operación y le envía un comprobante ACK al votante, tras lo cual la papeleta electrónica pasa a ser almacenada.

Algoritmo 9 LDR - Fase de Votación

1: El votante calcula la firma corta $S_{\mathcal{V}} = d_t H_1(\mathcal{V})$.
2: El votante envía la papeleta $\mathcal{B} = \{V_t, S_{V_t}, \mathcal{V}, S_{\mathcal{V}}\}$ al servidor de verificación.
3: El servidor de verificación calcula $m = m2s(V_t)$.
4: **si** $\tilde{e}(Q, S_{V_t}) = \tilde{e}(V_{AS}, H_1(M))$ **y** $\tilde{e}(Q, S_{\mathcal{V}}) = \tilde{e}(V_t, H_1(\mathcal{V}))$ **entonces**
5: El servidor de verificación genera aleatoriamente $a \in \mathbb{Z}_\ell^*$.
6: El servidor de verificación genera el comprobante $ACK = H(V_t||S_{V_t}||\mathcal{V}||S_{\mathcal{V}}||a)$.
7: El servidor de verificación genera la firma del comprobante $S_{ACK} = d_{VS} H_1(ACK)$
8: El servidor de verificación envía $\{ACK, S_{ACK}\}$ al votante y almacena la papeleta.
9: **si no**
10: Se rechaza la papeleta.
11: **fin si**
12: El votante comprueba que su papeleta ha sido firmada por el Servidor de Verificación $\tilde{e}(Q, S_{ACK}) = \tilde{e}(V_{VS}, H_1(ACK))$

Fase de recuento

Una vez ha concluido el periodo electoral, el servidor de verificación verifica que no existe ninguna duplicación ni falsificación de papeletas comprobando cada firma y mensaje presente en todas las papeletas. Asumiendo que cada votante tiene un par (d_t, V_t) distinto, las firmas de V_t y $\mathcal{V}$ deben ser únicas. Si el VS encontrara dos papeletas con alguna de sus firmas iguales, dicho par se consideraría fraudulento y los autores proponen tomar entonces únicamente una papeleta como válida, marcando la otra como fraudulenta.

Finalmente, el VS cuenta los votos de todas las papeletas válidas y publica dos listas: una con los comprobantes de todos los votos válidos y otra con los comprobantes de todos los votos fraudulentos. De este modo, cada votante puede verificar si su comprobante se encuentra o no en una de las listas.

6.4.2. Críticas y análisis de seguridad

Completamos la exposición con un análisis crítico personal del protocolo expuesto, atendiendo a los criterios de evaluación de la Sección 6.2.

Integridad

Buena parte depende de la implementación real del protocolo. Resulta especialmente desaconsejable su uso en ordenadores personales para votaciones de trascendencia, debido a la posibilidad de la existencia de *malware* en el ordenador del votante.

Durante la fase de recuento, la hipótesis de que cada par de votantes tiene un par (d_t, V_t) distinto puede resultar demasiado fuerte, y habría que estudiar la probabilidad de que esto ocurra en función del tamaño del censo electoral y los $\ell - 1$ valores posibles de d_t. De no ser abrumadoramente alta, incurriría en un problema de falta de Integridad.

Secreto del voto

Se logra gracias a la generación aleatoria de las claves pseudónimas d_t y V_t, que no tienen ninguna relación con el votante, y por el hecho de que la firma de V_t es ciega. Por lo tanto, si el Servidor de Votación o las autoridades electorales quisieran relacionar los votantes con sus papeletas, tendrían que romper la firma ciega de Boldyreva. Esto garantiza el secreto de voto tanto durante como después de la votación.

Si el Servidor de Autenticación tratase de vincular V_t con ID_V tendría dos opciones, conocidos $\tilde{M}$ y $S_{\tilde{M}}$. La primera sería resolver el PLDE en G_1 para tratar de obtener d_V de la igualdad $S_{\tilde{M}} = d_V\tilde{M}$. La segunda, aún más complicada, se trata de obtener V_t de la igualdad $\tilde{M} = bH_1(m2s(V_t))$, donde $b \in \mathbb{Z}_\ell^*$ es aleatorio y desconocido y la función resumen H_1 es resistente al cálculo de contraimágenes.

El comprobante ACK publicado tras la fase de recuento no permite al votante demostrar a quién voto, debido al valor aleatorio $a \in \mathbb{Z}_\ell^*$ concatenado al resto de su papeleta antes de aplicarle la función resumen. Se presume, por supuesto, un ℓ lo suficientemente grande.

Autenticación del votante

En la fase de autenticación, el votante solicita una papeleta a la que el Servidor de Autenticación aplica una firma ciega. Éste, antes de producirla, autentica al votante verificando la firma $S_{\tilde{M}}$ mediante la clave pública V_V del solicitante, tras lo cual lo elimina de la lista de votantes válidos.

En la fase de votación sólo es posible votar de poseer una de las papeletas firmadas por el AS, luego no hay riesgo de votos múltiples salvo que un votante válido fuera coercionado para entregar su papeleta a otra persona. Existen diversas medidas para evitar que esto ocurra, como hacer al votante emitir su voto en un entorno aislado justo tras obtener su papeleta.

Habría que estudiar, en función del método utilizado para la fase de registro, la posibilidad de realizar entonces un ataque para conseguir múltiples identidades electorales ID_V. Parece no obstante enormemente complicado, más si se opta por un Esquema Basado en la Identidad.

Accesibilidad

Depende completamente de la implementación que se haga del protocolo.

Disponibilidad

En el articulo original [LGDPRH13] los autores hablan sobre mecanismos de optimización a la hora de implementar su protocolo y realizan un estudio de eficiencia computacional, que luego comparan con otros ya existentes obteniendo resultados muy satisfactorios.

Inteligibilidad

Como en cualquier protocolo de voto electrónico, la percepción de seguridad depende enormemente del votante. La utilización del comprobante ACK, que le permite saber si su voto fue considerado o no, aumenta esta confianza. Con objetivo de evitar problemas de coerción, no obstante, resulta imposible verificar si el voto fue contado del mismo modo que fue emitido, para evitar al votante la posibilidad de demostrárselo a terceras personas.

Adicionalmente, los autores dan un método para auditar el sistema.

Bibliografía

[AHR07] R Michael Alvarez, Thad E Hall, and Brian F Roberts. Military voting and the law: procedural and technological solutions to the ballot transit problem. *Fordham Urb. LJ*, 34:935, 2007.

[BGDM+10] Jean-Luc Beuchat, Jorge E González-Díaz, Shigeo Mitsunari, Eiji Okamoto, Francisco Rodríguez-Henríquez, and Tadanori Teruya. High-speed software implementation of the optimal ate pairing over Barreto–Naehrig curves. In *Pairing-Based Cryptography-Pairing 2010*, pages 21–39. Springer, 2010.

[BK98] Ramachandran Balasubramanian and Neal Koblitz. The improbability that an elliptic curve has subexponential discrete log problem under the Menezes–Okamoto–Vanstone algorithm. *Journal of cryptology*, 11(2):141–145, 1998.

[BLS01] Dan Boneh, Ben Lynn, and Hovav Shacham. Short signatures from the Weil pairing. In *Advances in Cryptology—ASIACRYPT 2001*, pages 514–532. Springer, 2001.

[BN06] Paulo SLM Barreto and Michael Naehrig. Pairing-friendly elliptic curves of prime order. In *Selected areas in cryptography*, pages 319–331. Springer, 2006.

[Bol02] Alexandra Boldyreva. Threshold signatures, multisignatures and blind signatures based on the gap-Diffie-Hellman-group signature scheme. In *Public key cryptography—PKC 2003*, pages 31–46. Springer, 2002.

[BR93] Mihir Bellare and Phillip Rogaway. Random oracles are practical: A paradigm for designing efficient protocols. In *Proceedings of the 1st ACM conference on Computer and communications security*, pages 62–73. ACM, 1993.

[BR04a] Jordi Barrat and Josep M^{a} Reniu. Legal and social issues in electronic voting. Report on the catalan essays during the elections of november, 2003. *E-Government and E-Democracy: Progress and Challenges. México: IPN. Pág*, pages 129–136, 2004.

[BR04b] Jordi Barrat and Josep M^{a} Reniu. Democracia electrónica y participación ciudadana. *Informe sociológico y jurídico de la consulta ciudadana Madrid Participa. Ayuntamiento de Madrid/Scytl/Accenture, Madrid*, 2004.

[BSS05] Ian F Blake, Gadiel Seroussi, and Nigel P Smart. *Advances in elliptic curve cryptography*, volume 317. Cambridge University Press, 2005.

[BW05] Friederike Brezing and Annegret Weng. Elliptic curves suitable for pairing based cryptography. *Designs, Codes and Cryptography*, 37:133–141. Springer, 2005.

[CE14] Oficina del Censo Electoral. Elecciones al parlamento europeo de 25 de mayo de 2014 - lista de tablas. Technical report, Instituto Nacional de Estadística, 2014. Disponible electrónicamente en: goo.gl/BgLC1r.

[DH76] Whitfield Diffie and Martin E Hellman. New directions in cryptography. *Information Theory, IEEE Transactions on*, 22(6):644–654, 1976.

[Ele05] Observatorio Voto Electrónico. Informe 2m6: "Así, no". *Disponible electrónicamente en: http://www.votobit.org/archivos/PruebaVotoInternet2005.pdf*, 2005.

[FR94] Gerhard Frey and Hans-Georg Rück. A remark concerning m-divisibility and the discrete logarithm in the divisor class group of curves. *Mathematics of computation*, 62:865–874, 1994.

[FST10] David Freeman, Michael Scott, and Edlyn Teske. A taxonomy of pairing-friendly elliptic curves. *Journal of cryptology*, 23(2):224–280, 2010.

[HPS08] Jeffrey Hoffstein, Jill Pipher, and Joseph H Silverman. *An introduction to mathematical cryptography*. Springer, 2008.

[HR83] Martin E Hellman and Justin M Reyneri. Fast computation of discrete logarithms in GF(q). In *Advances in Cryptology*, pages 3–13. Springer, 1983.

[Jon05] D. Jones. Chain voting. *Disponible electrónicamente en: http://vote.nist.gov/threats/papers/ChainVoting.pdf*, 2005.

[Jou00] Antoine Joux. A one round protocol for tripartite Diffie–Hellman. In *Algorithmic number theory*, pages 385–393. Springer, 2000.

[Kah96] David Kahn. *The Codebreakers: The comprehensive history of secret communication from ancient times to the internet*. Simon and Schuster, 1996.

[Kob87] Neal Koblitz. Elliptic curve cryptosystems. *Mathematics of computation*, 48(177):203–209, 1987.

[LGDPRH13] Lourdes López-García, Luis J Dominguez Perez, and Francisco Rodríguez-Henríquez. A pairing-based blind signature e-voting scheme. *The Computer Journal*, page bxt069, 2013.

[Maa04] Martijn Maas. *Pairing-based cryptography*. Master's thesis, Technische Universiteit Eindhoven, 2004.

[Men93] Alfred J Menezes. *Elliptic curve public key cryptosystems*, volume 234. Springer, 1993.

[Mil86] Victor S Miller. Use of elliptic curves in cryptography. In *Advances in Cryptology—CRYPTO'85 Proceedings*, pages 417–426. Springer, 1986.

[Miy93] Atsuko Miyaji. Elliptic curves over $\mathbb{F}_p$ suitable for cryptosystems. In *Advances in Cryptology—AUSCRYPT'92*, pages 477–491. Springer, 1993.

[MOV93] Alfred J Menezes, Tatsuaki Okamoto, and Scott A Vanstone. Reducing elliptic curve logarithms to logarithms in a finite field. *Information Theory, IEEE Transactions on*, 39:1639–1646, 1993.

[MR+09] Víctor Manuel Morales Rocha et al. Seguridad en los procesos de voto electrónico remoto: registro, votación, consolidación de resultados y auditoría. 2009.

[MST] Josep M. Miret, Daniel Sadornil, and Juan G Tena. Elliptic curves with $j = 0, 1728$ and low embedding degree. *Unpublished*.

[MT93] Carlos Munuera and Juan G Tena. An algorithm to compute the number of points on elliptic curves of j-invariant 0 or 1728 over a finite field. *Rendiconti del Circolo Matematico di Palermo*. 42(1): 106–116. Springer, 1993.

[MVOV96] Alfred J Menezes, Paul C Van Oorschot, and Scott A Vanstone. *Handbook of applied cryptography*. CRC press, 1996.

[PS00] David Pointcheval and Jacques Stern. Security arguments for digital signatures and blind signatures. *Journal of cryptology*, 13(3):361–396, 2000.

[SA98] Takakazu Satoh and K Araki. Fermat quotients and the polynomial time discrete log algorithm for anomalous elliptic curves. *Commentarii Math. Univ. Sancti Pauli*, 47(1):81–92, 1998.

[Sem98] Igor Semaev. Evaluation of discrete logarithms in a group of p-torsion points of an elliptic curve in characteristic p. *Mathematics of Computation of the American Mathematical Society*, 67(221):353–356, 1998.

[Sha85] Adi Shamir. Identity-based cryptosystems and signature schemes. In *Advances in cryptology*, pages 47–53. Springer, 1985.

[Sho97] Victor Shoup. Lower bounds for discrete logarithms and related problems. In *Advances in Cryptology—EUROCRYPT'97*, pages 256–266. Springer, 1997.

[Sil09] Joseph H Silverman. *The arithmetic of elliptic curves*, volume 106. Springer, 2009.

[Sin11] Simon Singh. *The code book: the science of secrecy from ancient Egypt to quantum cryptography*. Random House LLC, 2011.

[Sma99] Nigel P Smart. The discrete logarithm problem on elliptic curves of trace one. *Journal of cryptology*, 12(3):193–196, 1999.

[SZ03] Susanne Schmitt and Horst G Zimmer. *Elliptic Curves: A computational approach*, volume 31. Walter de Gruyter, 2003.

[Uhl05] Chris Uhlmann. Polls apart. *About the House (Canberra, ACT)*, (24):48, 2005.