Centro de Investigación y de Estudios Avanzados del
Instituto Politécnico Nacional

Unidad Zacatenco
Departamento de Computación

El ataque gGHS aplicado a curvas de Galbraith-Lin-Scott

Tesis que presenta

Jesús Javier Chi Domínguez

Para obtener el grado de

Maestro en Ciencias en Computación

Director de tesis

Dr. Francisco José Rambó Rodríguez Henríquez

México, D. F. Diciembre, 2015

Agradecimientos

Al Consejo Nacional de Ciencia y Tecnología (CONACyT) por haberme brindado el apoyo económico, permitiéndome el haber culminado mis estudios de posgrado en el Cinvestav Unidad Zacatenco.

A Gora Adj, Thomaz Oliveira, Osmanbey Uzumkol, al Dr. Francisco (mi asesor), a la Dra. Adriana Lara y al Dr. Guillermo Morales (ambos mis sinodales), por sus comentarios, correciones y recomendaciones.

A Sofía Reza, Erika Ríos y Felipa Rosas, por su ayuda en cuestiones administrativas.

Esta tesis es dedicada a mi hermano Alex, a mis padres, a mis mejores amigos y amiga (que para mi son como hermanos y hermana): Arreola, Karen, Rafa, Toni, William, y Yasbedh. Y no olvidemos, al amor de mi vida: Argentina y a mi futura descendencia.

Resumen

En la actualidad, la criptografía de curvas elípticas (ECC) juega un papel muy importante en las aplicaciones de seguridad informática y, con el paso del tiempo, nuevos "ataques criptográficos" han surgido con el objetivo de quebrantar los servicios de seguridad proporcionados por estos sistemas criptográficos: confidencialidad, integridad, autenticación y no repudio.

El ataque gGHS es usado en la criptografía de curvas elípticas binarias para reducir instancias del Problema del Logaritmo Discreto (DLP) en una curva elíptica hacia el *jacobiano* de una curva hiperelíptica. Se dice que un criptosistema basado en curvas elípticas es vulnerable ante este ataque si es mucho más sencillo resolver esta nueva instancia del Problema del Logaritmo Discreto.

En esta tesis se presenta una implementación en Magma para resolver el problema del logaritmo discreto sobre una curva binaria GLS; construyendo así una curva vulnerable ante el ataque gGHS y adaptando el algoritmo propuesto por Enge y Gaudry para la resolución del Problema del Logaritmo Discreto en el *jacobiano* de una curva hiperelíptica. Este método es efectivo con todas las curvas definidas sobre campos binarios y puede ser aplicado a cada elemento de la clase de isogenias.

Abstract

Nowadays, the elliptic curve cryptography (ECC) plays an important role in the informatic security applications and with pass of time, new cryptographic attacks have being born with the main objective to break the security services given by these cryptosystems: confidentiality, integrity, authentication and non-repudiation.

The gGHS attack is used on the binary elliptic curve cryptography for reducing an instace of the discret logarithm problem (DLP) on an elliptic curve into the *Jacobian* of a hyperelliptic curve. A cryptosystem based on elliptic curves is called vulnerable against this attack if it is much easier to solve this new instace.

In this thesis it is presented a Magma implementation for solving the DLP on a binary GLS curve. For this purpose, we constructed a curve vulnerable against the gGHS Weil descent attack and adapted the small-genus algorithm proposed by Enge-Gaudry to solve the DLP on the *Jacobian* of a hyperelliptic curve. Furthermore, we give an efficient mechanism to check whether a randomly selected binary GLS curve is vulnerable against the gGHS attack. The method is suitable for all curves defined over binary fields and can be applied to each element of the isogeny class.

Índice general

Índice de figuras

Índice de cuadros

Índice de algoritmos

Lista de códigos

Abreviaciones y notación estándar

Notación básica

$:$, $\mid$, $\wedge$	tal que, tal que (usado en conjuntos), y
$\equiv$, $\cong$, $\neq$	congruente ó equivalente, isomorfo, diferente
$\exists$, $\nexists$, $\exists!$	existe, no existe, existe un único
$\forall$, $\in$, $\notin$	para todo, pertenece, no pertenece
$\mathbb{N}$, $\mathbb{Z}$, $\mathbb{Q}$, $\mathbb{R}$, $\mathbb{C}$	conjunto de los número naturales, enteros, racionales, reales y complejos
$\mathbb{Z}_n$, $\mathbb{Z}_n^\times$, $n\mathbb{Z}$	los números enteros módulo n, el conjunto $\mathbb{Z}_n - \{0\}$, el conjunto de los números enteros que son múltiplos de n
$\mathcal{M}$, $\mathcal{K}$, $\mathcal{C}$, $\mathcal{G}$	espacio de mensajes, espacio de claves, espacio de textos cifrados y algoritmo generador de claves
$\text{divmod}(a, b)$	función que calcula $q, r \in \mathbb{N} \cup \{0\}$ tales que $a = qb + r$ con $0 \leq r < b$
$\text{grad}(f)$	grado del polinomio f
`A`, `B`, `E`	Alicia, Beto, Eva
`a`, `b`, $\xleftarrow{\$}$	números enteros positivos seleccionados aleatoriamente, selección aleatoria
$\Rightarrow$, $\Leftrightarrow$	entonces, si y sólo si
$\subset$, $\subseteq$, $\lvert \cdot \rvert$	contensión propia de conjuntos, contensión de conjuntos con posible igualdad, cardinalidad (número de elementos) de un conjunto dado
$\widetilde{O}(g(n))$	$O(\log^k g(n))$ para algún $k \in \mathbb{N} \cup \{0\}$.

$A \setminus B$, $A \cup B$	el conjunto de los elementos que están en A pero que no están en B, el conjunto de los elementos que están en A o B
A, B, $\emptyset$	A y B denotan conjuntos, conjunto vacío
$L_x[\alpha, c]$	$e^{\left(c(\log x)^{\alpha}(\log\log x)^{1-\alpha}\right)}$
$O(g(n))$	$\{f\colon \mathbb{N} \to \mathbb{R} \mid \exists c \in \mathbb{R}^+ \; \exists n_0 \in \mathbb{N}\colon 0 \leq f(n) \leq cg(n) \; \forall n \geq n_0\}$
$o(g(n))$	$\{f\colon \mathbb{N} \to \mathbb{R} \mid \forall c \in \mathbb{R}^+ \; \exists n_0 \in \mathbb{N}\colon 0 \leq f(n) < cg(n) \; \forall n \geq n_0\}$
mcd(x, y)	máximo común divisor de x e y (x e y pueden ser números enteros o polinomios)
mcm(x, y)	mínimo común múltiplo de x e y (x e y pueden ser números enteros o polinomios)

Notación relacionada con curvas elípticas e hiperelípticas

$\mathcal{E}$, $\mathcal{H}$	curva elíptica e hiperelíptica
$\mathcal{E}$(K), $\mathcal{H}$(K)	conjunto de puntos K racionales de la curva elíptica $\mathcal{E}$, conjunto de puntos K racionales de la curva hiperelíptica $\mathcal{H}$
div(u, v)	divisor constiuido por los polinomios u y v
$\mathrm{Jac}_{\mathrm{k}}(\mathcal{H})$	jacobiano de la curva hiperelíptica $\mathcal{H}$ sobre el campo k
K($\mathcal{E}$), K($\mathcal{H}$)	campo de funciones racionales de la curva elíptica $\mathcal{E}$, campo de funciones racionales de la curva hiperelíptica $\mathcal{H}$
$\mathrm{Tr}_{\mathrm{K/k}}(\gamma)$	traza de γ sobre el campo K con respecto al subcampo k
G, $\mathtt{d_G}$	base de factores, cota de suavidad
φ, div(φ)	función racional de un campo de funciones racionales, divisor asociado a φ
a, b, g	parámetros de una curva elíptica dada por $\mathcal{E}\colon y^2 + xy = x^3 + ax^2 + b$, género de una curva hiperelíptica
P, (P)	punto sobre una curva elíptica o hiperelíptica, divisor del punto P

Notación relacionada con teoría de anillos, grupos y campos

$\mathbb{F}_q$, $\mathbb{F}_q^{\times}$, σ	campo finito con $q = p^n$ elementos, $\mathbb{F}_q - \{0\}$, automorfismo de Frobenius: $\sigma(x) = x^q$
dim(V)	dimensión del espacio vectorial V
F, $\overline{\mathrm{F}}$, K, k	extensión de campo, cerradura algebraica del campo F, campo y subcampo

G/H, R/I	grupo cociente, anillo cociente
G, H, N	grupo, subgrupo y subgrupo normal
$\mathrm{K}(\alpha_1, \ldots, \alpha_n)$	Extensión más pequeña del campo K que contiene a los elementos $\alpha_1, \ldots, \alpha_n \in \overline{\mathrm{K}}$
$\mathrm{K}[x_1, \ldots, x_n]$	anillo de polinomios en las variables $x_1, \ldots, x_n$ sobre el campo K
R, I	anillo e ideal
S, $V(\mathrm{S})$	conjunto de polinomios, variedad algebraica del conjunto S
V, $\mathcal{L}_{\mathrm{K}}(\mathtt{C})$	espacio vectorial, K-espacio vectorial generado por el conjunto $\mathtt{C}$
f^σ	aplicación del automorfismo de Frobenius: si $f(x) = \sum f_i x^i \in \mathbb{F}_2[x]$ entonces $f^\sigma(x) = \sum f_i \sigma^i(x) = \sum f_i x^{q^i}$

Abreviaciones

Enc, Dec	algoritmo de cifrado y descifrado
ataque eGHS	extensión del ataque GHS
ataque gGHS	generalización del ataque GHS
DLP	Problema del Logaritmo Discreto (*Discrete Logarithm Problem*)
GHS	Gaudry-Hess-Smart
GLS	Galbraith-Lin-Scott
Magma	Magma Computational Algebra System
MOV	Menezes, Okamoto y Vanstone
QPA	algoritmo cuasi-polinomial (*quasi-polynomial algorithm*)

Capítulo 1

Introducción

> *– Este hombre, por una parte, cree que sabe algo, mientras que no sabe nada. Por otra parte, yo, que igualmente no sé nada, no creo saber algo.*
>
> ***Apología de Sócrates* - Platón**

Con la era de la información vigente, la necesidad de proteger información mediante métodos, cada vez más sofisticados, ha tomado gran importancia en nuestras vidas. Un claro ejemplo de este fenómeno, es el aumento en la dependencia que nuestra sociedad tiene con respecto a los sistemas eléctronicos, como consecuencia el uso de las redes sociales, transacaciones financieras, firmas y dinero electrónico requieren que el intercambio y acceso seguro a la información se realice a través de los siguientes servicios de seguridad [Hankerson et al., 2003]:

- *Confidencialidad*: Mantener todos los datos en secreto excepto para las personas que tengan autorización.
- *Integridad de datos*: Asegurar que los datos recibidos no han sido alterados por alguna entidad no autorizada.
- *Autenticación del origen de datos*: Corroborar que el mensaje proviene del emisor correcto y que no se trata de un impostor.
- *Autenticación de entidades*: Asegurar que las entidades son quienes dicen ser.
- *No repudio*: Impedir la negación de cualquier acto realizado en la comunicación por cualquiera de las entidades participantes.

1.1. Criptografía de clave pública

La criptografía es la disciplina encargada del diseño y análisis de técnicas que permiten realizar comunicaciones seguras en presencia de adversarios maliciosos. En un modelo de comunicación básico en el cual dos entidades, `A` (Alicia) y

`B` (Beto), se comunican a través de un canal inseguro y ante la presencia de un posible adversario, `E` (Eva), cuyo objetivo es quebrantar cualquier servicio de seguridad proporcionado por `A` y `B`, se desea que Alicia y Beto mantengan comunicación sin que Eva pueda comprender ni alterar el mensaje enviado.

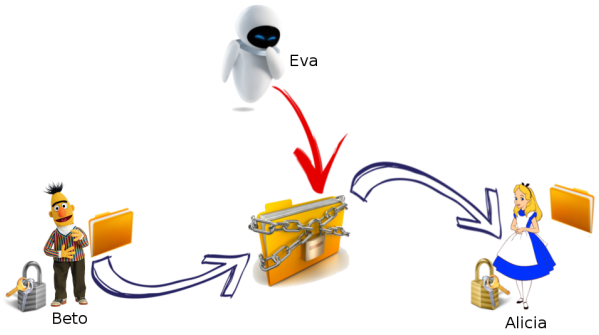

Figura 1.1: Modelo básico de comunicación.

En este contexto, el mensaje a enviar es conocido como texto en claro, mientras que la codificación del mensaje se denomina como texto cifrado. Esta codificación, que es incomprensible para Eva, es realizada mediante el uso de un algoritmo de cifrado. El papel de Beto es enviar el texto cifrado a Alicia, quién utiliza un algoritmo de descifrado para obtener el mensaje original. La clave del éxito en este esquema radica en que los algoritmos de cifrado y descifrado sean operaciones inversas, es decir, que del texto en claro se pueda obtener el texto cifrado y viceversa.

Formalmente, sean $\mathcal{K}$, $\mathcal{M}$ y $\mathcal{C}$ los conjuntos de claves, mensajes y mesajes cifrados, respectivamente. Entonces, un criptosistema o sistema criptográfico consiste de una tupla

$$(\mathrm{n}_{\mathrm{bits}}, \mathrm{Enc}, \mathrm{Dec}, \mathcal{G}, \mathcal{K}, \mathcal{M}, \mathcal{C}) \tag{1.1}$$

donde $\mathcal{G}\colon \mathbb{N} \to \mathcal{K}$ determina un algoritmo de generación de claves, el cual toma como entrada un entero positivo, n, dando una clave $k \in \mathcal{K}$ de longitud en bits igual a $O(\mathrm{n})$. Las funciones $\mathrm{Enc}\colon \mathcal{K} \times \mathcal{M} \to \mathcal{C}$ y $\mathrm{Dec}\colon \mathcal{K} \times \mathcal{C} \to \mathcal{M}$ determinan los algoritmos de cifrado y descifrado, respectivamente, y son tales que para cada clave $k \in \mathcal{K}$ se cumple

$$\mathrm{Dec}\left(k, \mathrm{Enc}(k, m)\right) = m \quad \forall m \in \mathcal{M} \tag{1.2}$$

Los sistemas criptográficos actuales pueden ser clasificados en dos clases, la *criptografía de clave privada* o *criptografía simétrica*, y la *criptografía de clave pública* o *criptografía asimétrica*. La criptografía de clave privada es utilizada para cifrar grandes cantidades de información a grandes velocidades, debido a

la simpleza de las operaciones que involucra. Su característica principal consiste en que utiliza una misma clave para cifrar y descifrar los mensajes.

Por su parte, en la criptografía de clave pública cada entidad posee un par de claves, una clave pública de cifrado y una clave privada de descifrado. Es importante mencionar que todas las entidades conocen la clave pública, por lo que debe ser imposible, computacionalmente hablando, deducir la clave privada a partir de la clave pública, de ahí que este tipo de criptosistemas estén basados en el uso de *funciones con trampa* [1] a problemas matemáticos difíciles.

A pesar de que la criptografía de clave pública dio pie a la elaboración de nuevos protocolos criptográficos, tiene la desventaja de que el tamaño de las claves es mayor y las operaciones de cifrado y descifrado son considerablemente más costosas que las involucradas en criptografía simétrica. Por ello, es común usar sistemas de clave pública para establecer una clave que es usada en un sistema de clave privada para cifrar y descifrar la información.

1.2. Antecedentes

La *criptografía de clave pública* fue introducida por los investigadores Diffie y Hellman (1976), cuyo trabajo, considerado como el desarrollo más impactante en la criptografía, propone una manera interesante de resolver el problema de intercambio de claves. Dos años después, Adleman, Sharmir y Rivest (1978) dieron a conocer el primer esquema de clave pública (conocido como RSA), cuya seguridad se basa en la dificultad de la factorización de números enteros.

Posteriormente, Elgamal (1985) propuso un esquema de clave pública (conocido como Elgamal), cuya seguridad descansa en la dificultad de cómo resolver el problema del logaritmo discreto en campos finitos. En ese mismo año, Miller (1985) y Koblitz (1985) propusieron, de manera independiente, el uso de curvas elípticas para el diseño de criptosistemas de clave pública, *basando ambas su seguridad en la dificultad de resolver el Problema del Logaritmo Discreto en curvas elípticas*. Estas propuestas dieron pie a la elaboración de una gran gama de protocolos basados en la criptografía de curvas elípticas, que hoy en día es uno de los temas más investigados en la disciplina de seguridad informática.

En una conferencia impartida en el *Workshop on Elliptic Curve Cryptography* (ECC '98), Frey (1998) propuso una manera de relacionar puntos de una curva elíptica definida sobre un campo finito $\mathbb{F}_p$, siendo p un número primo, con puntos de una variedad abeliana. Cuatro años más tarde, Gaudry et al. (2002) aplicaron esta idea a curvas elípticas definidas sobre $\mathbb{F}_{2^m}$, donde m es un entero positivo, reduciendo así el Problema del Logaritmo Discreto de esta curva elíptica, a una curva hiperelíptica definida sobre un subcampo de $\mathbb{F}_{2^m}$; siendo conocido como el *ataque del descenso de Weil de Gaudry-Hess-Smart* ó como el *ataque GHS*. En ese mismo año, Galbraith et al. (2002) extendieron

[1] Una **función con trampa** consiste en una función matemática cuyo cálculo directo es sencillo, pero en la que el cálculo de la función inversa se torna muy complejo cuando se desconoce un *secreto*, es decir, involucra un elevado número (por ejemplo, exponencial) de operaciones [Katz y Lindell, 2007].

este ataque mediante el uso de isogenias expandiendo el número de posibles curvas vulnerables. Más tarde, Hess (2004) generalizó el ataque mediante el uso de extensiones de Artin-Schreier.

Dicha reducción, sembró la curiosidad de preguntarse si acaso el Problema del Logaritmo Discreto en la curva hiperlíptica obtenida es, o no, más sencillo de resolver que el Problema del Logaritmo Discreto en la curva elíptica. Pregunta que será retomada a lo largo de este trabajo de tesis.

1.3. Estado del arte

En los últimos años diversos investigadores han intentado responder dicha pregunta, abordando curvas elípticas definidas sobre campos finitos muy particulares, es decir, aún no existe una respuesta general a la pregunta planteada.

Menezes y Qu (2001) realizaron un estudio para determinar la vulnerabilidad, ante el ataque GHS, en curvas elípticas definidas sobre campos finitos binarios de la forma $\mathbb{F}_{2^n}$ donde $n \in [160, 600]$ es un número entero; concluyendo que este ataque es impráctico cuando n es un número primo. Este análisis fue continuado por Maurer et al. (2001), en el cual afirman la existencia de curvas elípticas vulnerables ante este ataque para el caso cuando n es un número compuesto [1], con el detalle de que no se determina la estructura de dichas curvas.

Por otro lado, Hankerson et al. (2011) llevaron a cabo este análisis de seguridad a una familia de curvas elípticas conocidas como curvas elípticas GLS [Galbraith et al., 2009] definidas sobre campos finitos binarios de la forma $\mathbb{F}_{2^{2\ell}}$, donde ℓ es un número primo. Dicho análisis se realizó considerando valores de ℓ mayores a 80 y menores a 256; obteniendo como resultado que $\ell = 127$ es el único número primo, en ese intervalo, que genera una familia de curvas elípticas vulnerables ante el ataque. Este resultado es de gran importancia debido a que hoy en día las curvas elípticas GLS se encuentran en su auge de uso, por el hecho de que brindan una mayor velocidad en implementaciones aunque se cree que brindan un nivel relativamente "bajo" de seguridad (véase el cuadro 1.1), y no existe "hasta ahora" un mecanismo que determine si dada una curva elíptica binarias GLS seleccionada aleatoriamente es, o no, vulnerable ante este ataque.

Curvas elípticas [2]	Velocidad [3]	Seguridad [3]
Primas	☆☆☆	☆☆☆☆☆
Binarias	☆☆☆☆	☆☆☆☆
GLS	☆☆☆☆☆	¿☆☆☆?

Cuadro 1.1: Comparativa entre curvas binarias y primas

[1] véase la definición 2.1.5

[2] Las curvas elípticas definidas sobre campos finitos $\mathbb{F}_{p^n}$, siendo $p > 2$ un número primo y n un entero positivo, son conocidas como curvas elípticas primas y curvas elípticas binarias si $p = 2$.

[3] En la simbología empleada en este cuadro, a mayor cantidad de ☆, mayor la velocidad y el nivel de seguridad.

1.4. Organización de la tesis

La tesis ha sido organizada en siete capítulos: el capítulo 2 tiene como objetivo introducir los conceptos básicos y generales que son indispensables para la comprensión del mecanimos de vulnerabilidad ante el ataque gGHS que se propone en este trabajo tesis.

Posteriormente, los capítulos 3 y 4 abordan a detalle el tema de curvas elípticas e hiperelípticas binarias, describiendo los conceptos principales, como el Problema del Logaritmo Discreto asociado a dichas curvas. Por su parte, el capítulo 5 desglosa la idea general tanto el ataque GHS, como su extensión y generalización.

En el capítulo 6 se analizan algunas propiedades importantes de las curvas GLS y las dos posibles maneras de determinar una posible vulnerabilidad ante el ataque gGHS y eGHS. De igual manera, se describe el mecanismo de vulnerabilidad mediante una serie de teoremas y corolarios necesarios para la justificación, y demostración, del porqué dicho mecanismo funciona. Para finalmente concluir con el capítulo 7 donde se enuncian algunos puntos importantes en el trabajo a futuro.

Observación 1.4.1 *La notación establecidad en el capítulo 2 será usada a lo largo de todo este trabajo de tesis. La notación establecida en los capítulos 3, 4 y 5 será usada en los capítulos 6 y 7.*

Capítulo 2

Conceptos algebraicos básicos

Algunos misterios siempre escaparán a la mente humana. Para convencernos de ello, sólo hay que echar un vistazo a las tablas de los números primos, y ver que no reina ni orden, ni reglas.

Évariste Galois

Estructuras algebraicas como los grupos, anillos y campos son básicas para el desarrollo de esquemas criptográficos. Por ende, es de suma importacia conocer y comprender los conceptos más usados de estas estructuras algebraicas para luego aplicarlas en el área de la criptografía. El contenido de este capítulo ha sido organizado en cuatro secciones: la sección 2.1 introduce, de manera breve y concisa, la aritmética modular, la cual es la base matemática de la criptografía de curvas elípticas e hiperelípticas. Posteriormente, las secciones 2.2 y 2.3 describen los conceptos principales, y de interés criptográfico, de teoría de grupos y campos finitos. Además en la sección 2.2 también se mencionan los dos problemas derivados del Problema del Logaritmo Discreto en grupos. En la sección 2.4 se detalla de manera breve los conceptos requeridos "relacionados con curvas algebraicas" para el entendimiento del ataque GHS. El material a ser discutido en este capítulo ha sido tomado de los libros de texto [Lidl y Niederreiter, 1997, Friedberg et al., 2002, Hoffstein et al., 2008, Cormen et al., 2009, Guerrero y Pérez, 2010, Cohen et al., 2012, Judson, 2014].

2.1. Aritmética modular

Las siguientes definiciones, ejemplos y teoremas fueron obtenidos de las secciones 1.3 y 1.4 de [Hoffstein et al., 2008].

Dado un conjunto S, la **cardinalidad** de S, denotada por |S|, está definida como el número de elementos que pertenecen al conjunto S. Cuando dicho

número de elementos es infinito, se dice que la cardinalidad es infinita y es denotado mediante $|\mathrm{S}| = \infty$, mientras que en el caso contrario $|\mathrm{S}| < \infty$; si sucede que $|\mathrm{S}| = 0$, entonces se dice que S es un conjunto **vacío** y es denotado por $\mathrm{S} = \emptyset$.

Definición 2.1.1 *Sean* S_1 *y* S_2 *dos conjuntos. Entonces,* S_1 *es un* ***subconjunto*** *de* S_2*, denotado por* $\mathrm{S}_1 \subseteq \mathrm{S}_2$*, si cada elemento* $s_1 \in \mathrm{S}_1$ *satisface que* $s_1 \in \mathrm{S}_2$*. Si existe* $s_2 \in \mathrm{S}_2$ *tal que* $s_2 \notin \mathrm{S}_1$*, entonces se dice que* S_1 *es un* ***subconjunto propio*** *de* S_2 *y es denotado por* $\mathrm{S}_1 \subset \mathrm{S}_2$*.*

Definición 2.1.2 *El conjunto de los* ***números enteros*** *está definido como el conjunto* $\mathbb{Z} = \{\ldots, -2, -1, 0, 1, 2, 3, \ldots\}$*, mientras que el subconjunto de los números enteros positivos está definido como el conjunto de los* ***números naturales****,* $\mathbb{N} = \{1, 2, 3, \ldots\}$*.*

Definición 2.1.3 *Sean* $a, b \in \mathbb{Z}$ *con* $a \neq 0$*, se dice que* a ***divide*** *a* b*, denotado por* $a|b$*, si existe* $c \in \mathbb{Z}$ *tal que* $b = ac$*. El número* a *es llamado* ***factor*** *o* ***divisor*** *de* b*,* y b *es un* ***múltiplo*** *de* a*.*

Definición 2.1.4 *Sean* $a, b \in \mathbb{Z} \setminus \{0\}$*, el* ***máximo común divisor****, abreviado por* ***mcd****, de* a y b *está definido como el entero positivo* d *que satisface las siguientes dos propiedades:*

1. $d|a$ *y* $d|b$*,*
2. $\forall c \in \mathbb{Z}$ *tal que* $c|a$ *y* $c|b$*, se tiene que* $c|d$*.*

Teorema 2.1.1 *(****Teorema de la división****) Sean* $a \in \mathbb{Z}$ y $b \in \mathbb{N}$*, entonces existen dos únicos números* $q, r \in \mathbb{Z}$ *tales que* $a = qb + r$ *y* $0 \leq r < b$*.*

Teorema 2.1.2 *Sean* $a, b \in \mathbb{Z} \setminus \{0\}$*. Entonces,* $\mathrm{mcd}(a, b) = \mathrm{mcd}(b, r)$ *donde* r *es el residuo obtenido de la división de* a *por* b*.*

Teorema 2.1.3 *Sean* $a, b \in \mathbb{Z} \setminus \{0\}$ y $d = \mathrm{mcd}(a, b)$*. Entonces,* d *es el menor entero positivo que puede expresarse en la forma* $ax + by$ *con* $x, y \in \mathbb{Z}$*.*

Input: $a, b \in \mathbb{Z} \setminus \{0\}$
Output: $\mathrm{mcd}(a, b)$
1 $r_{-1} \leftarrow a$, $r_0 \leftarrow b$, $i \leftarrow 0$;
2 **while** $r_i \neq 0$ **do**
3 $\quad i \leftarrow i + 1$;
4 $\quad q_i, r_i \leftarrow \mathrm{divmod}(r_{i-2}, r_{i-1})$
5 **end**
6 **return** r_{i-1}

Algoritmo 2.1: Algoritmo de Euclides

El **algoritmo de Euclides**, el cual es descrito mediante el algoritmo 2.1, también nos proporciona un método para calcular dos valores enteros x e y tales que $\text{mcd}(a,b) = ax + by$. Este método, consiste en ir despejando el residuo de la última división, el que nos da el valor $\text{mcd}(a,b)$, hacia atrás hasta llegar a los valores a y b de partida. En la literatura matemática, este método es conocido como el **algoritmo extendido de Euclides** y puede ser descrito mediante el algoritmo 2.2.

Input: $a, b \in \mathbb{Z} \setminus \{0\}$
Output: $d = \text{mcd}(a,b), x, y \in \mathbb{Z}'$ tales que $d = ax + by$

$r_{-1} \leftarrow a$, $r_0 \leftarrow b$, $i \leftarrow 0$;
$x_1 \leftarrow 0$, $x_2 \leftarrow 1$, $y_1 \leftarrow 1$, $y_2 \leftarrow 0$;
while $r_i \neq 0$ **do**
    $i \leftarrow i + 1$;
    $q_i, r_i \leftarrow \text{divmod}(r_{i-2}, r_{i-1})$;
    $x \leftarrow x_2 - q_i x_1$, $x \leftarrow y_2 - q_i y_1$;
    $x_2 \leftarrow x_1$, $x_1 \leftarrow x$, $y_2 \leftarrow y_1$, $y_1 \leftarrow y$;
end
return (r_{i-1}, x_2, y_2)

Algoritmo 2.2: Algoritmo extendido de Euclides

Ejemplo 2.1.1 *Para los enteros* $a = 217$ *y* $b = 21$ *obtenemos lo siguiente:*

$$\begin{aligned} 217 &= 10 \cdot 21 + 7 \\ 21 &= 2 \cdot 7 + 0 \end{aligned}$$

en consecuencia, $\text{mcd}(217, 21) = 7$ *y se puede verificar que* $217 \cdot 1 + 21 \cdot (-10) = 7$*, es decir,* $x = 1$ *e* $y = -10$*.*

Ejemplo 2.1.2 *Si en el ejemplo anterior cambiamos el entero* b *por* 39*, obtenemos lo siguiente:*

$$\begin{aligned} 217 &= 5 \cdot 39 + 22 \\ 39 &= 1 \cdot 22 + 17 \\ 22 &= 1 \cdot 17 + 5 \\ 17 &= 3 \cdot 5 + 2 \\ 5 &= 2 \cdot 2 + 1 \\ 2 &= 2 \cdot 1 + 0 \end{aligned}$$

$$\begin{aligned} 5 \cdot 1 + 2 \cdot (-2) &= 1 \\ 5 \cdot 7 + 17 \cdot (-2) &= 1 \\ 22 \cdot 7 + 17 \cdot (-9) &= 1 \\ 22 \cdot 16 + 39 \cdot (-9) &= 1 \\ 217 \cdot 16 + 2 \cdot (-89) &= 1 \end{aligned}$$

en consecuencia, $\text{mcd}(217, 39) = 1$ *y se puede verificar que* $217 \cdot 16 + 21 \cdot (-89) = 1$*.*

Definición 2.1.5 *Se dice que un número entero positivo $p > 1$ es un* ***número primo*** *si sus únicos divisores positivos son 1 y p. Si un número entero positivo dado es diferente de uno y no es primo, entonces se denomina* ***número compuesto****.*

Ejemplo 2.1.3 *El número 7 es un número primo debido a que los números enteros 2, 3, 4, 5 y 6 no dividen a 7; mientras que el número 21 es número compuesto por el hecho de que $21 = 3 \cdot 7$.*

Definición 2.1.6 *Sea $1 < n \in \mathbb{N}$, se dice que $a, b \in \mathbb{Z}$ son congruentes módulo n, denotado por $a \equiv b \mod n$, si y sólo si $n|(a - b)$.*

Esta relación de congruencia es una relación de equivalencia para todo $n \in \mathbb{N}$, es decir, se satisfacen las siguientes tres propiedades:

1. **Reflexividad**: $\forall a \in \mathbb{N}$, $a \equiv a \mod n$.
2. **Simetría**: $\forall a, b \in \mathbb{N}$: $a \equiv b \mod n \Rightarrow b \equiv a \mod n$.
3. **Transitividad**: $\forall a, b, c \in \mathbb{N}$: $a \equiv b \mod n \wedge b \equiv c \mod n \Rightarrow a \equiv c \mod n$.

Como en toda relación de equivalencia, podemos definir el conjunto cociente de las clases de equivalencia originadas por la relación de congruencia. En este caso, la relación clasifica a cualquier entero a según el residuo obtenido al dividirlo por el módulo n. Llamaremos $\mathbb{Z}_n$ al conjunto cociente de $\mathbb{Z}$ respecto de la relación de congruencia módulo n, y denotaremos a la clase de equivalencia de un entero a, mediante el conjunto:

$$[a] = \{b \colon a \equiv b \mod n\}$$

Puesto que para todo $a \in \mathbb{Z}$ se cumple que [a] = [r] en $\mathbb{Z}_n$ donde r es el residuo de dividir a entre n, entonces $\mathbb{Z}_n$ puede ser descrito de la siguiente manera:

$$\mathbb{Z}_n = \{[0], [1], ..., [n-1]\}$$

A este conjunto cociente se le conoce como el conjunto de residuos módulo m.

Definición 2.1.7 *Para todo $[a]$ y $[b]$ en $\mathbb{Z}_n$, la adición y la multiplicación modular están definidas de la siguiente manera:*

- ***Adición****: $[a] + [b] = [a + b]$.*
- ***Producto****: $[a] \cdot [b] = [a \cdot b]$.*

Definición 2.1.8 *Un elemento $[a] \in \mathbb{Z}_n$ es* ***invertible*** *si existe un elemento $[b] \in \mathbb{Z}_n$ tal que $[a] \cdot [b] = [1]$. El elemento $[b]$ es llamado el* ***inverso*** *de $[a]$ en $\mathbb{Z}_n$ y es denotado por $[a]^{-1}$.*

Teorema 2.1.4 *$[a]$ es invertible en $\mathbb{Z}_n$, si y sólo si* $\text{mcd}(a, n) = 1$.

En la literatura matemática, los elementos de $\mathbb{Z}_n$ son escritos omitiendo los corchetes, es decir, $\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}$.

Ejemplo 2.1.4 *Sea $n = 3$, entonces tenemos que $\mathbb{Z}_3 = \{0, 1, 2\}$ y que $1^{-1} \equiv 1$* mód 3 *y $2^{-1} \equiv 2$* mód 3.

$+$	0	1	2
0	0	1	2
1	1	2	0
2	2	0	1

$\cdot$	0	1	2
0	0	0	0
1	0	1	2
2	0	2	1

Cuadro 2.1: Adición y multiplicación en $\mathbb{Z}_3$

Ejemplo 2.1.5 *Sea $n = 217$ y $a = 21$, entonces* 21 *no es invertible módulo* 217 *debido a que* $\text{mcd}(217, 21) = 7$.

Ejemplo 2.1.6 *Sea p un número primo, entonces $\mathbb{Z}_p$ tiene la propiedad que todo elemento distinto de* 0 *es invertible módulo p.*

Cabe mencionar que la aritmética modular en los enteros, descrita en esta sección, puede ser generalizada "de manera natural" para el conjunto de los polinomios en una variable sobre un campo dado.

2.2. Funciones

Las siguientes definiciones, ejemplos y teoremas fueron obtenidos de la sección 0.2 de [Judson, 2014] y de la sección 3.1 de [Cormen et al., 2009].

Dados dos conjuntos no vacíos A y B, una **función** f entre ellos, denotada por $f\colon A \to B$, es una asignación que relaciona cada elemento $a \in A$ con un único elemento $b = f(a) \in B$. Los conjuntos A y $f(A) = \{f(a)\colon a \in A\} \subseteq B$ son llamados **dominio** e **imagen** de la función, respectivamente.

Ejemplo 2.2.1 *Supongamos que $A = \{1, 2, 3\}$ y $B = \{a, b, c\}$. Entonces, en la figura 2.1 la relación $f\colon A \to B$ determina una función mientras que la relación $g\colon A \to B$ no, debido a que el elemento $1 \in A$ no tiene una única asignación en B, es decir, $g(1) = a$ y $g(1) = b$.*

Definición 2.2.1 *Sea $f\colon A \to B$ una función entre los conjuntos A y B. Entonces, f es una función:*

i. ***sobreyectiva****, si $f(A) = B$,*

ii. ***inyectiva****, si $\forall a_1, a_2 \in A\colon\ f(a_1) = f(a_2) \Rightarrow a_1 = a_2$,*

iii. ***biyectiva****, si f es una función inyectiva y sobreyectiva.*

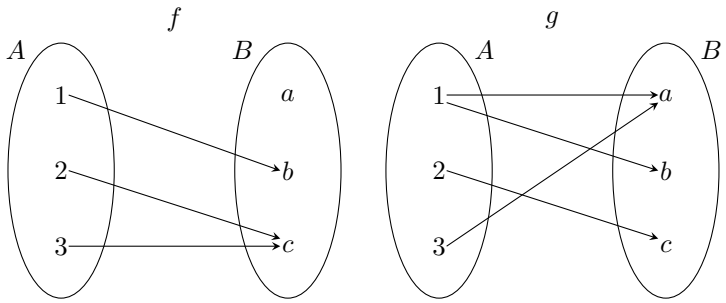

Figura 2.1: Ejemplo de una función

Ejemplo 2.2.2 *La asignación* $f\colon \mathbb{Q} \to \mathbb{Z}$ *dada por* $f(\frac{a}{b}) = a$*, es una función sobreyectiva que no es inyectiva debido a que* $f(\frac{3}{7}) = 3 = f(\frac{3}{17})$ *y* $\frac{3}{7} \neq \frac{3}{17}$*.*

Ejemplo 2.2.3 *La asignación* $f\colon \mathbb{Z}^+ \to \mathbb{R}^+$ *dada por* $f(a) = \sqrt{a}$*, es una función inyectiva debido a que si* $a, b \in \mathbb{Z}^+$ *y* $f(a) = f(b)$*, entonces* $a = (\sqrt{a})^2 = f(a)^2 = f(b)^2 = (\sqrt{b})^2 = b$*. Además, f no es una función sobreyectiva por el hecho de que no existe* $a \in \mathbb{Z}^+$ *tal que* $\sqrt{a} = \sqrt[3]{7}$*, ya que si suponemos que existe tal* $a \in \mathbb{Z}^+$*, entonces tendría que suceder que* $a = \sqrt[3]{7 \cdot 7} \notin \mathbb{Z}^+$ *(lo cual es una contradicción).*

Ejemplo 2.2.4 *La asignación* $f\colon \mathbb{Z} \to \mathbb{Z}$ *dada por* $f(a) = a+1$*, es una función biyectiva. Esto es debido a que si* $a, b \in \mathbb{Z}$ *son tales que* $a + 1 = f(a) = f(b) = b + 1$*, entonces* $a = b$ *(inyectividad de la función). De igual manera, si* $c \in \mathbb{Z}$ *entonces* $f(c - 1) = c$ *(sobreyectividad de la función).*

Definición 2.2.2 *Sea* $g\colon \mathbb{N} \to \mathbb{R}$ *una función. Entonces,* $O(g(n))$ *denota el siguiente conjunto de funciones*

$$\left\{ f\colon \mathbb{N} \to \mathbb{R} \mid \exists c \in \mathbb{R}^+ \; \exists n_0 \in \mathbb{N}\colon 0 \leq f(n) \leq cg(n) \; \forall n \geq n_0 \right\}. \tag{2.1}$$

Definición 2.2.3 *Sea* $g\colon \mathbb{N} \to \mathbb{R}$ *una función. Entonces, se define* $\widetilde{O}(g(n))$ *como* $O\left(g(n) \log^k g(n)\right)$ *para algún* $k \in \mathbb{N} \cup \{0\}$*.*

Definición 2.2.4 *Sea* $g\colon \mathbb{N} \to \mathbb{R}$ *una función. Entonces,* $o(g(n))$ *denota el siguiente conjunto de funciones*

$$\left\{ f\colon \mathbb{N} \to \mathbb{R} \mid \forall c \in \mathbb{R}^+ \; \exists n_0 \in \mathbb{N}\colon 0 \leq f(n) < cg(n) \; \forall n \geq n_0 \right\}. \tag{2.2}$$

En la literatura matemática, se suele escribir $f(n) = O(g(n))$, $f(n) = \widetilde{O}(g(n))$ y $f(n) = o(g(n))$ para referirse a $f(n) \in O(g(n))$, $f(n) \in \widetilde{O}(g(n))$ y $f(n) \in o(g(n))$, respectivamente.

Ejemplo 2.2.5 *Las funciones* $f, g, h\colon \mathbb{N} \to \mathbb{R}$ *dadas por* $f(n) = 6n^4 - 2n^3 + 5$*,* $g(n) = n^4$ *y* $h(n) = 2n$ *satisfacen* $f(n) = O(n^4)$*,* $g(n) = O(4 \cdot n^4 \log n) = \widetilde{O}(n^4)$ *y* $h(n) = o(n^2)$*.*

Definición 2.2.5 *Sean* $c \in \mathbb{R}^+$, $\alpha \in (0,1)$ *y* $x \in \mathbb{N}$. *Entonces,* $\mathrm{L}_x[\alpha, c]$ *denota el conjunto de funciones* $e^{\left(c(\log x)^{\alpha}(\log\log x)^{1-\alpha}\right)}$ [Enge y Gaudry, 2002].

Observación 2.2.1 *Nótese que* $\mathrm{L}_x[0,c] = (\log n)^c$ *es una función polinomial y* $\mathrm{L}_x[1,c] = n^c$ *es una función exponencial, ambas en* $\log n$. *Por consiguiente,* $(\log n)^c < \mathrm{L}_x[\alpha, c] < n^c$, *es decir,* $\mathrm{L}_x[\alpha, c]$ *es una función sub-exponencial en* $\log n$.

2.3. Grupos

Las siguientes definiciones, ejemplos y teoremas fueron obtenidos del capítulo 1 de [Guerrero y Pérez, 2010].

Un **grupo** $(\mathrm{G}, \star)$, abreviado por G, es una estructura algebraica conformada por un conjunto G no vacío y una operación binaria en G, con las siguientes propiedades:

1. **Cerradura** o **Clausura**. La operación $\star$ es cerrada sobre los elementos de G, es decir:
$$\forall a, b \in \mathrm{G},\ a \star b \in \mathrm{G}.$$
2. **Asociatividad**. La operación $\star$ es asociativa sobre los elementos de G, es decir:
$$\forall a, b, c \in \mathrm{G},\ (a \star b) \star c = a \star (b \star c).$$
3. **Elemento neutro**. $\exists! e \in \mathrm{G}\colon\ \forall a \in \mathrm{G},\ a \star e = e \star a = a$.
4. **Elemento inverso**. $\forall a \in \mathrm{G}\ \exists \overline{a} \in \mathrm{G}\colon\ a \star \overline{a} = \overline{a} \star a = e$.

El grupo G es llamado un grupo abeliano si la operación $\star$ es conmutativa, es decir, $\forall a, b \in \mathrm{G}$ se satisface:

$$a \star b = b \star a.$$

Cabe mencionar que si solamente se satisfacen las dos primeras propiedades, entonces G es llamado un **semigrupo**; si G es un semigrupo y satisface la propiedad del elemento neutro, entonces G es denominado **monoide** (analogamente, existen semigrupos y monoides abelianos).

Definición 2.3.1 *Sea* $\mathrm{G} = (\mathrm{G}, \star, e)$ *un grupo, el orden del grupo* G *está definido como la cardinalidad del conjunto* G. *Para cada elemento* $g \in \mathrm{G}$, *el orden de* g, *denotado por* $ord(g)$, *está definido como el mínimo entero positivo,* r, *que satisface la siguiente propiedad*

$$\star^r(g) = \underbrace{g \star \ldots \star g}_{r\ veces} = e.$$

Si $|\mathrm{G}| = \infty$ *y* $g \in \mathrm{G}$ *satisface que no existe dicho mínimo entero positivo, entonces se dice que el elemento* g *es de orden infinito.*

Ejemplo 2.3.1 *Sea* $\mathrm{G} = \mathbb{Z}_4 = \{0, 1, 2, 3\}$. *Entonces,* $(\mathrm{G}, +)$ *forma un grupo abeliano de orden* 4*, donde 0 es de orden 1, 2 de orden 2, 1 y 3 de orden 4, y* $+$ *denota la adición modular en* $\mathbb{Z}_4$ *dada por la tabla de Cayley que es descrita por el cuadro 2.2.*

$+$	0	1	2	3
0	0	1	2	3
1	1	2	3	0
2	2	3	0	1
3	3	0	1	2

Cuadro 2.2: El grupo $\mathbb{Z}_4$

Ejemplo 2.3.2 *Sea* $\mathrm{G} = \mathbb{Z}_4 = \{0, 1, 2, 3\}$. *Entonces* $(\mathrm{G}, \star)$ *forma un grupo abeliano de orden* 4*, donde 0 es de orden 1, mientras que 1, 2 y 3 son de orden 4, y* $\star$ *está definida mediante la tabla de Cayley que es descrita por el cuadro 2.3. Este grupo es conocido como el grupo de Klein y es denotado por* K_4*.*

$\star$	0	1	2	3
0	0	1	2	3
1	1	0	3	2
2	2	3	0	1
3	3	2	1	0

Cuadro 2.3: El grupo K_4 de Klein

Ejemplo 2.3.3 *El conjunto de todas las matrices cuadradas de orden* $n \times n$ *cuyas entradas son números reales, y la multiplicación de matrices, forman un grupo (no-abeliano) el cual es de orden infinito.*

2.3.1. Subgrupos

Dado un grupo G, y $\mathrm{H} \neq \emptyset$ un subconjunto de G. Se dice que H es un **subgrupo** de G, denotado por $\mathrm{H} \leq \mathrm{G}$, si la operación binaria de G restringida a H hace de este un grupo.

Definición 2.3.2 *Sea* G *un grupo y* $S \neq \emptyset$ *un subconjunto de* G *cuya cardinalidad es igual a* n *(i.e., S consta de* n *elementos distintos). Entonces, el* ***subgrupo generado*** *por S está definido como el conjunto* $\langle S \rangle = \{s_1^{i_1} \dots s_n^{i_n} : s_i \in S, i_j \in \mathbb{Z}\}$ *donde el elemento neutro y la operación binaria coinciden con los del grupo* G*. Se dice que* G *es* ***finitamente generado****, abreviado por f.g., si existe un conjunto S tal que* $\mathrm{G} = \langle S \rangle$*. En particular, si* G *es f.g. por un solo elemento entonces* G *es llamado* ***ciclíco****.*

Teorema 2.3.1 (**de Lagrange**) *Sea* G *un grupo de orden finito y* $\mathrm{H} \leq \mathrm{G}$*, entonces el orden de* H *divide al orden de* G*.*

Ejemplo 2.3.4 *Sea* $(\mathrm{G}, \cdot)$ *con* $\mathrm{G} = \mathbb{Z}_7^\times = \{1, 2, 3, 4, 5, 6\}$ *y* $\cdot$ *el producto modular en* $\mathbb{Z}_7$ *restringido a* G*, el cual es defindo mediante la tabla de Cayley que es descrita mediante el cuadro 2.4. Entonces,* $\langle 2\rangle = \{1, 2, 4\} = \langle 4\rangle$, $\langle 3\rangle = \mathbb{Z}_7^\times = \langle 5\rangle$, $\langle 6\rangle = \{1, 6\}$*; y claramente el Teorema de Lagrange se cumple.*

$\cdot$	1	2	3	4	5	6
1	1	2	3	4	5	6
2	2	4	6	1	3	5
3	3	6	2	5	1	4
4	4	1	5	2	6	3
5	5	3	1	6	4	2
6	6	5	4	3	2	1

Cuadro 2.4: El grupo $\mathbb{Z}_7^\times$

Definición 2.3.3 *Sean* G *un grupo de orden finito y* $\mathrm{H} \leq \mathrm{G}$*. Sean* a *y* b *dos elementos del grupo* G

- *Se dice que* a *y* b *son* ***congruentes por la izquierda****, denotado por* $a \equiv_i b \mod \mathrm{H}$*, si* $\overline{a} \star b \in \mathrm{H}$*. La* ***clase lateral izquierda*** *que contiene al elemento* a *está definida como el conjunto* $a \star \mathrm{H} = \{a \star h | h \in \mathrm{H}\}$*, es decir,* $x \in a \star \mathrm{H} \Leftrightarrow x \equiv_i a \mod \mathrm{H}$*.*
- *Se dice que* a *y* b *son* ***congruentes por la derecha****, denotado por* $a \equiv_d b$ mód H*, si* $a \star \overline{b} \in \mathrm{H}$*. La* ***clase lateral derecha*** *que contiene al elemento* a *está definida como el conjunto* $\mathrm{H} \star a = \{h \star a | h \in \mathrm{H}\}$*, es decir,* $x \in \mathrm{H} \star a \Leftrightarrow x \equiv_d a \mod \mathrm{H}$*.*

Ambas expresiones son relaciones de equivalencia y dividen al grupo G en clases de equivalencia disjuntas por pares.

Definición 2.3.4 *Sea* G *un grupo de orden finito y* $\mathrm{N} \leq \mathrm{G}$*. Entonces* N *es un* ***subgrupo normal*** *de* G*, denotado por* $\mathrm{N} \trianglelefteq \mathrm{G}$*, si las clases laterales derechas e izquierdas coinciden, es decir,* $g \star \mathrm{N} = \mathrm{N} \star g$ *para todo* $g \in \mathrm{G}$*.*

Ejemplo 2.3.5 *En vista de que* $(\mathbb{Z}_7^\times, \cdot)$ *es un grupo abeliano, entonces se puede ver fácilmente que el grupo generado por el número* 2*,* $\langle 2\rangle$*, es un subgrupo normal de* $\mathbb{Z}_7^\times$*.*

Definición 2.3.5 *Sea* G *un grupo de orden finito y* $\mathrm{N} \trianglelefteq \mathrm{G}$*. El* ***grupo cociente*** *de* G *sobre* N *está definido como el grupo* $\mathrm{G/N} = \{g \star \mathrm{N} \colon g \in \mathrm{G}\}$ *donde la operación binaria está dada por la relación* $(a \star \mathrm{N}) \star (b \star \mathrm{N}) = (a \star b) \star N$ *para todo* $a \star \mathrm{N}$ *y* $b \star \mathrm{N}$ *en* G/N*.*

Ejemplo 2.3.6 *Retomando el grupo* $(\mathbb{Z}_7^\times, \cdot)$*, y tomando el subgrupo normal* $\langle 2\rangle$*, obtenemos las siguientes clases de equivalencia para cada elemento:*

$$\langle 2\rangle = 2 \cdot \langle 2\rangle = 4 \cdot \langle 2\rangle = \{1, 2, 4\}$$
$$3 \cdot \langle 2\rangle = 5 \cdot \langle 2\rangle = 6 \cdot \langle 2\rangle = \{3, 6, 5\}$$

cuya operación asociada está dada mediante la tabla de Cayley que es descrita por el cuadro 2.5.

$+$	$\langle 2\rangle$	$3\cdot\langle 2\rangle$
$\langle 2\rangle$	$\langle 2\rangle$	$3\cdot\langle 2\rangle$
$3\cdot\langle 2\rangle$	$3\cdot\langle 2\rangle$	$\langle 2\rangle$

Cuadro 2.5: El grupo $\mathbb{Z}_7^\times/\langle 2\rangle$

2.3.2. Homomorfismos

Dados dos grupos $(\mathrm{G}_1,\star)$ y $(\mathrm{G}_2,*)$, una función $f\colon \mathrm{G}_1 \to \mathrm{G}_2$ es llamado **homomorfismo** de grupos (o simplemente homomorfismo), si $\forall a,b \in \mathrm{G}_1$ se cumple lo siguiente:

$$f(a\star b) = f(a) * f(b).$$

La **imagen** del homomorfismo f, denotada por $\mathrm{Im}(f)$, está definida como el conjunto $\{f(g_1)\colon g_1 \in \mathrm{G}_1\}$, mientras que el **núcleo** del homomorfismo f, denotado por $\mathrm{Ker}(f)$, está definido como el conjunto $\{g_1 \in G_1\colon f(g_1) = e_{\mathrm{G}_2}\}$. Tanto la imagen como el núcleo del homomorfismo f son subgrupos de G_2 y G_1, respectivamente; en particular, $\mathrm{Ker}(f) \trianglelefteq \mathrm{G}_1$. Los homomorfismos se clasifican como sigue:

- **Monomorfismo**, si es una función inyectiva.
- **Epimorfismo**, si es una función sobreyectiva.
- **Isomorfismo**, si es una función biyectiva.
- **Endomorfismo**, si G_1 coincide con G_2.
- **Automorfismo**, si es un endomorfismo e isomorfismo al mismo tiempo.

Ejemplo 2.3.7 *Sean* G *un grupo finito y* $\mathrm{N} \trianglelefteq \mathrm{G}$ *un subgrupo normal, entonces el siguiente mapeo, conocido como la* ***proyección canónica****, es un epimorfismo de grupos.*

$$\begin{aligned} \varphi\colon \mathrm{G} &\to \mathrm{G/N} \\ g &\mapsto g\star N. \end{aligned}$$

Ejemplo 2.3.8 *Sean* $(\mathbb{Q}^\times,*)$ *el grupo de los números racionales y* $(\mathbb{R}^\times,\cdot)$ *el grupo de los número reales, ambos sin el elemento cero y con la multiplicación como operación binaria, entonces*

$$\begin{aligned} f\colon \mathbb{Q}^\times &\to \mathbb{R}^\times \\ x &\mapsto x \end{aligned}$$

es un monomorfismo de grupos que no es un epimorfismo de grupos.

Ejemplo 2.3.9 *Sean* $(\mathbb{R}, +)$ *el grupo de los números reales con la adición como operación binaria y* $(\mathbb{R}^+, \cdot)$ *el grupo de los número reales positivos con la multiplicación como operación binaria, entonces*

$$\begin{array}{rcl} f\colon \mathbb{R} & \to & \mathbb{R}^+ \\ x & \mapsto & e^x \end{array}$$

es un isomorfismo de grupos.

Ejemplo 2.3.10 *Sean* $(\mathbb{Q}^\times, \cdot)$ *el grupo de los números racionales con la multiplicación como operación binaria, entonces*

$$\begin{array}{rcl} f\colon \mathbb{Q} & \to & \mathbb{Q} \\ \dfrac{x}{y} & \mapsto & x \end{array}$$

es un endomorfismo de grupos (en particular, no es mono ni epimorfismo de grupos).

Ejemplo 2.3.11 *Sean* $(\mathbb{R}, +)$ *el grupo de los números reales con la adición como operación binaria, entonces*

$$\begin{array}{rcl} f\colon \mathbb{R} & \to & \mathbb{R} \\ x & \mapsto & \dfrac{3}{7}x \end{array}$$

es un automorfismo de grupos.

Definición 2.3.6 *Sean* G_1 *y* G_2 *dos grupos, se dice que* G_1 *es* ***isomorfo*** *a* G_2*, denotado por* $\mathrm{G}_1 \cong \mathrm{G}_2$*, si existe un isomorfismo entre* G_1 *y* G_2*.*

Ejemplo 2.3.12 $\mathbb{Z}_7^\times / \langle 2 \rangle \cong \mathbb{Z}_2$ *ya que definiendo* $f\colon \mathbb{Z}_7^\times / \langle 2 \rangle \to \mathbb{Z}_2$ *mediante la relación* $f(\langle 2 \rangle) = 0$ *y* $f(3 \cdot \langle 2 \rangle) = 1$ *vemos que dicho mapeo es un isomorfismo de grupos.*

El siguiente teorema es muy importante en la teoría de grupos (cuya demostración es basada en el uso de **los tres teoremas de isomorfimos**).

Teorema 2.3.2 (**Fundamental de Homomorfismos**) *Sean* $f\colon \mathrm{G} \to \mathrm{H}$ *un homomorfismo de grupos y* $\mathrm{N} \triangleleft \mathrm{G}$ *un subgrupo normal contenido en* $\mathrm{Ker}(f)$*. Entonces, existe un único homomorfismo* $\psi\colon \mathrm{G/N} \to \mathrm{H}$ *tal que* $\psi \circ \varphi = f$ *y* $\varphi\colon \mathrm{G} \to \mathrm{G/N}$ *es la proyección canónica.*

2.3.3. Problema del Logaritmo Discreto en grupos

Dado un grupo cíclico $\mathrm{G} = \langle g \rangle$, el Problema del Logaritmo Discreto consiste en encontrar la solución a la ecuación $\star^x(g) = h \in \mathrm{G}$, denotada por $x = \log_g(h)$. Usando la notación multiplicativa, la anterior ecuación queda expresada de la siguiente manera:

$$g^x = h,$$

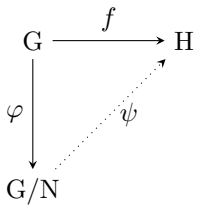

Figura 2.2: Teorema Fundamental de Homomorfismos.

mientras que en notación aditiva

$$xg = h.$$

Existen dos problemas que se derivan del Problema del Logaritmo Discreto, los cuales (mediante la notación multiplicativa) están definidos como siguen:

- **El problema de Diffie-Hellman**. Dados $g, g^a, g^b \in \mathrm{G}$, calcular g^{ab}.
- **El problema de decisión de Diffie-Hellman**. Dados $g, g^a, g^b, g^c \in \mathrm{G}$ decidir si acaso $g^{ab} = g^c$.

2.4. Anillos

Las siguientes definiciones, ejemplos y teoremas provienen de los capítulo 3 y 4 de [Guerrero y Pérez, 2010], de la sección 3.2 de [Lidl y Niederreiter, 1997] y del capítulo 1 de [Friedberg et al., 2002].

Un **anillo** $(\mathrm{R}, +, \cdot)$, abreviado por R, es un estructura algebraica, conformada por un conjunto $\mathrm{R} \neq \emptyset$ y dos operaciones binarias, adición y multiplicación, que satisface las siguientes condiciones:

1. $(\mathrm{R}, +)$ es un grupo abeliano teniendo como elemento neutro "aditivo" el "0_R".
2. $(\mathrm{R}, \cdot)$ es un semigrupo.
3. El producto se distribuye con respecto a la adición, es decir, $\forall a, b, c \in \mathrm{K}$ se cumple que

$$\begin{aligned} a \cdot (b + c) &= a \cdot b + a \cdot c, \\ (b + c) \cdot a &= b \cdot a + c \cdot a. \end{aligned}$$

El anillo R es llamado conmutativo si $(\mathrm{R}, \cdot)$ es un semigrupo abeliano. Cabe mencionar que si $(\mathrm{R}, \cdot)$ es un monoide, entonces R es denominado un **anillo unitario**, con elemento neutro "multiplicativo" denotado por "1_R". De manera análoga al caso de grupos, un **subanillo** de un anillo R es un anillo S, denotado

por $\mathrm{S} < \mathrm{R}$, tal que $\mathrm{S} \subseteq \mathrm{R}$ y las operaciones binarias asociadas de S coinciden con las operaciones binarias de R restringidas al subconjunto S. De aquí en adelante, se supondrá que todo anillo será un anillo unitario (se omitirá la palabra unitario).

Ejemplo 2.4.1 *El conjunto* $\mathbb{C}$ *de los números complejos junto con la adición y la multiplicación usual forma un anillo.*

Ejemplo 2.4.2 *El conjunto* $\mathbb{Q}$ *de los números racionales junto con la adición y la multiplicación usual es un subanillo del anillo de los números reales* $\mathbb{R}$*.*

Definición 2.4.1 *Sea* R *un anillo e* $\mathrm{I} < \mathrm{R}$ *un subanillo de* R*, entonces* I *es un* ***ideal*** *de* R*, denotado por* $\mathrm{I} \trianglelefteq \mathrm{R}$*, si* $a \cdot r \in I$ *y* $r \cdot a \in I$*,* $\forall r \in \mathrm{R}$ *y* $\forall a \in \mathrm{I}$*.*

Definición 2.4.2 *Sea* R *un anillo, un ideal* ***maximal*** *es un ideal* $\mathrm{M} \triangleleft \mathrm{R}$*, el cual satisface que los únicos ideales que lo contienen son* R *y él mismo.*

Ejemplo 2.4.3 *El conjunto* $2\mathbb{Z}$ *de los enteros que son multiplos de* 2*, es decir, el conjunto de los números pares, junto con la adición y la multiplicación usual forma un ideal maximal del anillo* $\mathbb{Z}$*. El conjunto* $\mathrm{R} = \mathbb{Z} \setminus 2\mathbb{Z} = \{2k+1 \colon k \in \mathbb{Z}\}$ *no forma un ideal debido a que el producto de dos números enteros, uno impar y otro par, da como resultado un número par, es decir, si* $2k_1 \in 2\mathbb{Z} \subset \mathbb{Z}$ *y* $2k_2 + 1 \in \mathrm{R}$ *entonces* $2k_1\,(2k_2+1) = 2\,(2k_1k_2 + k_1) = 2k_3 \notin \mathrm{R}$*. En particular,* R *no forma un anillo debido que la suma de dos números enteros impares da como resultado un número entero par.*

Ejemplo 2.4.4 *El conjunto* $4\mathbb{Z}$ *de los múltiplos de* 4 *no es un ideal maximal debido a que* $4\mathbb{Z} \subset 2\mathbb{Z}$*.*

Definición 2.4.3 *Sea* R *un anillo e* $\mathrm{I} \trianglelefteq \mathrm{R}$ *un ideal de* R*, entonces el* ***anillo cociente*** *está definido como el anillo* $\mathrm{R/I} = \{r + \mathrm{I} \colon r \in \mathrm{R}\}$ *donde las operaciones de adición y multiplicación están dadas por las siguientes dos relaciones:*

$$\begin{aligned} (r_1 + \mathrm{I}) + (r_1 + \mathrm{I}) &= (r_1 + r_2) + \mathrm{I}, \quad \forall r_1, r_2 \in \mathrm{R}, \\ (r_1 + \mathrm{I}) \cdot (r_1 + \mathrm{I}) &= (r_1 \cdot r_2) + \mathrm{I}, \quad \forall r_1, r_2 \in \mathrm{R}. \end{aligned}$$

Ejemplo 2.4.5 *El anillo cociente* $\mathbb{Z}/2\mathbb{Z}$ *es determinado por el conjunto* $\{0 + 2\mathbb{Z}, 1 + 2\mathbb{Z}\}$ *y las operaciones binarias dadas por las tablas de Catley que son descrita por el cuadro 2.6.*

$+$	$0+2\mathbb{Z}$	$0+2\mathbb{Z}$
$0+2\mathbb{Z}$	$0+2\mathbb{Z}$	$1+2\mathbb{Z}$
$1+2\mathbb{Z}$	$1+2\mathbb{Z}$	$0+2\mathbb{Z}$

$\cdot$	$0+2\mathbb{Z}$	$0+2\mathbb{Z}$
$0+2\mathbb{Z}$	$0+2\mathbb{Z}$	$0+2\mathbb{Z}$
$1+2\mathbb{Z}$	$0+2\mathbb{Z}$	$1+2\mathbb{Z}$

Cuadro 2.6: Adición y multiplicación en $\mathbb{Z}/2\mathbb{Z}$

Definición 2.4.4 *Un **homomorfismo de anillos** es definido como una función que es un homomorfismo de grupos para ambas operaciones binarias, por lo que tanto el **núcleo** e **imagen** de dicho homomorfismo están definidos de la misma manera que en el caso de grupos. Por consiguiente, los diferentes tipos de homomorfismos de anillos que existen son los mismos que en el caso de grupos, implicando así que dos anillos* R_1 *y* R_2 *son **isomorfos** si existe un isomorfismo entre ellos, y de igual manera que en el caso de grupos es denotado por* $\mathrm{R}_1 \cong \mathrm{R}_2$.

2.4.1. Campos

Un **campo** K es un anillo conmutativo con la propiedad de que $(\mathrm{K}, \cdot)$ es un grupo. La característica de un campo K, está definida como el mínimo entero positivo, n, que cumple $n1_{\mathrm{K}} = 0_{\mathrm{K}}$. Si no existe tal n, se dice que la característica de K es 0. Un **subcampo** del campo K es un subanillo k con la propiedad de que también es un campo.

Teorema 2.4.1 *Sean* R *un anillo y* $\mathrm{M} \trianglelefteq \mathrm{R}$*, entonces* M *es un ideal maximal si y sólo si* R/M *es un campo.*

Igual que en el caso de grupos, un campo K es un campo finito si K consta de un número finito de elementos. Un resultado conocido, gracias a la teoría de Galois, es el hecho de que todo campo finito tiene exactamente p^m elementos, para algún número p, que coincide con la característica del campo, y un entero positivo m. En la literatura matemática, un campo finito con p^m elementos es denotado por $\mathbb{F}_{p^m}$, y los elementos neutro aditivo y multiplicativo son denotados por 0 y 1, respectivamente.

2.4.2. Anillos de polinomios sobre campos

Dado un campo K, un **polinomio** f en la variable x con coeficientes en el campo K, es una combinación lineal finita "con coeficientes en K" de los monomios x^α , es decir,

$$f(x) = \sum_{\alpha \in \mathrm{A}} a_\alpha x^\alpha, \quad a_\alpha \in \mathrm{K}$$

donde $\mathrm{A} \subset \mathbb{N} \cup \{0\}$ y $|A| < \infty$. El conjunto de todos los polinomios en la variable x con coeficientes en K es denotado por $\mathrm{K}[x]$. El **grado** de un polinomio $f \in \mathrm{K}[x]$, denotado por $\mathrm{grad} f$, está definido como el máximo del comjunto A, por lo cual si el polinomio f tiene grado n, entonces f puede ser escrito con una suma finita de la forma

$$f(x) = \sum_{i=0}^{n} a_i x^i, \quad a_i \in \mathrm{K}.$$

Definición 2.4.5 *Sean* $f(x) = \sum_{i=0}^{n} a_i f^i$ *y* $g(x) = \sum_{i=0}^{m} b_i g^i$ *dos polinomios en* $\mathrm{K}[x]$ *de grado* n *y* m*, respectivamente, entonces la **suma** de los polinomios*

f y g está definida mediante el siguiente polinomio de grado menor o igual a $\max\{n, m\}$*:*

$$(f+g)(x) = f(x) + g(x) = \sum_{i=0}^{\max\{n,m\}} (a_i + b_i)x^i$$

mientras que el ***producto*** *de los polinomios f y g está definido mediante el siguiente polinomio de grado* $m+n$*:*

$$(f \cdot g)(x) = f(x) \cdot g(x) = \sum_{i=0}^{n+m} \left(\sum_{j=0}^{i} a_j b_{i-j} \right) x^i.$$

Teorema 2.4.2 *Sea* K *un campo, entonces* $\mathrm{K}[x]$ *forma un anillo abeliano mediante la adición y el producto de polinomios en* $\mathrm{K}[x]$*. A este anillo se le conoce como* ***anillo de polinomios*** *en una variable sobre el campo* K*.*

Ejemplo 2.4.6 *Sean* $f(x) = x^2 + 1$ *y* $g(x) = x^3 + x + 1$ *dos polinomios en* $\mathbb{Z}[x]$*, entonces* $(f \cdot g)(x) = x^5 + 2x^3 + x^2 + x + 1$ *y* $(f+g)(x) = x^3 + x^2 + x + 2$*.*

Definición 2.4.6 *Un polinomio* $f \in \mathrm{K}[x]$ *de grado n es un* ***irreducible*** *si no puede ser expresado como el producto de polinomios de grado menor a n en* $\mathrm{K}[x]$*.*

Ejemplo 2.4.7 *El polinomio* $x^2 + 1 \in \mathbb{R}[x]$ *es irreducible.*

Ejemplo 2.4.8 *El polinomio* $x^2 + x + 1 \in \mathbb{F}_2[x]$ *es irreducible, mientras que* $x^2 + 1 \in \mathbb{F}_2[x]$ *no es irreducible debido a que* $x^2 + 1 = (x+1)^2$*.*

Teorema 2.4.3 *Si* $f \in \mathrm{K}[x]$ *es un polinomio irreducible, entonces el ideal generado por f es un ideal maximal. Por consiguiente,* $\mathrm{K}[x]/\langle f \rangle$ *es un campo.*

2.4.3. Espacios vectoriales

Dado un campo K, un **espacio vectorial** sobre el campo K (o **K-espacio vectorial**), es un conjunto $\mathrm{V} \neq \emptyset$, dotado de dos operaciones binarias: la *adición* ($\forall x, y \in \mathrm{V}$, $\exists! x + y \in \mathrm{V}$) y la *multiplicación escalar* ($\forall a \in \mathrm{K}\ \forall x \in \mathrm{V}$, $\exists! ax \in \mathrm{V}$) que satisfacen las siguientes condiciones:

1. $\forall x, y \in \mathrm{V}$, $x + y = y + x$.
2. $\forall x, y, z \in \mathrm{V}$, $(x + y) + z = x + (y + z)$.
3. $\exists 0 \in \mathrm{V} \colon\ x + 0 = x\ \forall x \in \mathrm{V}$.
4. $\forall x \in \mathrm{V}$, $\exists y \in \mathrm{V} \colon\ x + y = 0$.
5. $\forall x \in \mathrm{V}$, $1x = x$.

6. $\forall a, b \in \mathrm{K}$ y $\forall x \in \mathrm{V}$, $a(bx) = (ab)x$.

7. $\forall a \in \mathrm{K}$ y $\forall x, y \in \mathrm{V}$, $a(x + y) = ax + ayx$.

8. $\forall a, b \in \mathrm{K}$ y $\forall x \in \mathrm{V}$, $(a + b)x = ax + bx$.

Ejemplo 2.4.9 *Sea* K *un campo, entonces el anillo de polinomios* $\mathrm{K}[x]$ *es un* K*-espacio vectorial.*

Ejemplo 2.4.10 *Sea* $\mathbb{C}$ *el conjunto de los número complejos. Entonces,* $\mathbb{C}$ *es un* $\mathbb{R}$*-espacio vectorial.*

Definición 2.4.7 *Sea* V *un* K*-espacio vectorial y* $\emptyset \subset \mathrm{S} \subseteq \mathrm{V}$*. Entonces,* $u \in \mathrm{V}$ *es una* ***combinación lineal*** *de vectores en* S *si* $\exists u_1, \ldots, u_n \in \mathrm{S}$ *y* $\exists a_1, \ldots, a_n$ *tales que* $u = \sum_{i=1}^{n} a_i u_i$*. En este caso, decimos que* u *es una combinación lineal de los vectores* $u_1, \ldots, u_n$ *y los elementos* $a_1, \ldots, a_n$ *son llamados* ***coeficientes*** *de la combinación lineal.*

Definición 2.4.8 *Sea* V *un* K*-espacio vectorial y* $\emptyset \subset \mathrm{S} \subseteq \mathrm{V}$*. Entoces, el espacio vectorial* ***generado*** *por el conjunto* S*, denotado por* $\mathcal{L}_{\mathrm{K}}(\mathrm{S})$ *está definido como el conjunto de todas las combinacions lineales de los vectores en* S*. Como caso especial, se define* $\mathcal{L}_{\mathrm{K}}(\emptyset) = \{0\}$*.*

Ejemplo 2.4.11 *Sea* $\mathbb{C}$ *el* $\mathbb{R}$*-espacio vectorial de los números complejos. Entonces,* $\mathcal{L}_{\mathbb{C}}(\{1\}) = \mathbb{R}$ *y* $\mathcal{L}_{\mathbb{C}}(\{i\}) = i\mathbb{R}$ *donde* $i^2 = -1$*.*

Definición 2.4.9 *Sea* V *un* K*-espacio vectorial y* $\emptyset \subset \mathrm{S} \subseteq \mathrm{V}$*. Entonces,* S ***genera*** *a* V *si* $\mathcal{L}_{\mathrm{K}}(\mathrm{S}) = \mathrm{V}$*.*

Ejemplo 2.4.12 *Sea* $\mathbb{C}$ *el* $\mathbb{R}$*-espacio vectorial de los números complejos. Entonces,* $\{1, i\}$ *genera a* $\mathbb{C}$ *donde* $i^2 = -1$*.*

Definición 2.4.10 *Sea* V *un* K*-espacio vectorial y* $\emptyset \subset \mathrm{S} \subseteq \mathrm{V}$*. Entonces,* S *es* ***linealmente independiente*** *si* $\exists u_1, \ldots, u_n \in \mathrm{S}$ *y* $\exists a_1, \ldots, a_n$ *(con al menos un* $a_i \neq 0$ *para algún* $i \in \{1, \ldots, n\}$*) tales que* $\sum_{i=1}^{n} a_i u_i = 0$*. En este contexto,se dice que los elementos* $u_1, \ldots, u_n$ *son* ***linealmente dependientes****.*

Definición 2.4.11 *Sea* V *un* K*-espacio vectorial y* $\emptyset \subset \mathrm{S} \subseteq \mathrm{V}$*. Entonces,* S *es* ***linealmente independiente*** *si no es linealmente dependiente. De manera similar se definen elementos* ***linealmente independientes****.*

Ejemplo 2.4.13 *Sea* $\mathbb{C}$ *el* $\mathbb{R}$*-espacio vectorial de los números complejos. Entonces, el conjunto* $\{i, \frac{3}{5}i\}$ *es linealmente dependiente; mientras que el conjunto* $\{1, i\}$ *es linealmente independiente donde* $i^2 = -1$*.*

Definición 2.4.12 *Una* ***base*** β *de un* K*-espacio vectorial* V*, es un conjunto linealmente independiente que genera a* V*. Si* β *es una base de* V*, entonces se dice que los elementos de* β *forman una base de* V*.*

Un resultado muy importante, sobre espacios vectoriales, es el hecho de que toda base de un espacio vectorial tiene el mismo número de elementos [Friedberg et al., 2002]. A este único número se le denomina dimensión del espacio vectorial, es decir, si β es una base de V y β consta de n elementos, entonces se dice que V es un K-espacio vectorial de dimensión n, denotado por $\dim(\mathrm{V}) = n$.

Ejemplo 2.4.14 *Sea* $\mathbb{C}$ *el* $\mathbb{R}$*-espacio vectorial de los números complejos. Entonces, el conjunto* $\{1, i\}$ *es una base de* $\mathbb{C}$.

2.4.4. Extensiones de campos finitos

En el contexto general de campos, una **extensión** de un campo K está definida como un campo F que tiene como subcampo a K. El siguiente teorema nos da una caracterización de los campos finitos (resultado del teorema 2.5 de [Lidl y Niederreiter, 1997]).

Teorema 2.4.4 *Sea* K *un campo finito de característica p. Entonces* $|\mathrm{K}| = p^n$ *y cualquier otro campo finito con* p^n *elementos, es isomorfo a* K.

Teorema 2.4.5 *Sea* F *una extensión de grado n del campo* K*, entonces F es un espacio vectorial de dimensión n sobre el campo* K *cuya base canónica es* $\{1, x, \ldots, x^{n-1}\}$.

Ejemplo 2.4.15 *En vista de que* $x^2 + 1 \in \mathbb{R}[x]$ *es un polinimio irreducible, entonces* $\mathbb{R}[x]/\langle x^2+1\rangle$ *es una extensión del campo* $\mathbb{R}$*, más aún* $\mathbb{R}[x]/\langle x^2+1\rangle \cong \mathbb{C}$.

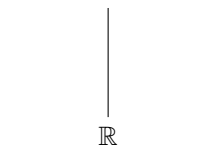

Figura 2.3: $\mathbb{C}$ visto como una extensión de grado 2 de $\mathbb{R}$

Definición 2.4.13 *Sea* F *una extensión de grado n del campo* K*, entonces se denomina* ***grado de la extensión*** F *a la dimensión de* F *como* K*-espacio vectorial.*

El siguiente resultado es una consecuencia de los teoremas 2.4.3 y 2.4.4 y del hecho de que para cada entero positivo n, existe un polinomio irreducible de grado n en $\mathrm{K}[x]$ [Lidl y Niederreiter, 1997].

Teorema 2.4.6 *Sean* $n \in \mathbb{N}$*, p un número primo y* $\mathbb{F}_{p^n}$ *(el campo finito con* p^n *elementos), se tiene que existe un polinomio* $f \in \mathbb{F}_p[x]$ *de grado n el cual es irreducible y satisface que* $\mathbb{F}_{p^n} \cong \mathbb{F}_p[x]/\langle f\rangle$*. En otras palabras,* $\mathbb{F}_{p^n}$ *puede ser representado por el conjunto de todos los polinomios de grado menor a n módulo un polinomio irreducible f de grado n.*

Corolario 2.4.1 *Sean $n, m \in \mathbb{N}$, p un número primo y $\mathrm{K} = \mathbb{F}_{p^m}$ (un campo finito con p^m elementos), entonces existe un polinomio $f \in \mathrm{K}[x]$ de grado n el cual es irreducible y satisface que $\mathbb{F}_{p^{nm}} \cong \mathrm{K}[x]/\langle f \rangle$.*

Ejemplo 2.4.16 *Dado que $x^2 + x + 1$, $x^3 + x + 1$ y $x^6 + x^4 + x^3 + x + 1$ son polinimios irreducibles en $\mathbb{F}_2[x]$, podemos ver que si $\mathrm{k}_1 = \mathbb{F}_2[x]/\langle x^2 + x + 1\rangle$ y $\mathrm{k}_2 = \mathbb{F}_2[x]/\langle x^3 + x + 1\rangle$, entonces $\mathrm{k}_1/\langle x^3 + x + 1\rangle \cong \mathrm{k}_2/\langle x^2 + x + 1\rangle \cong \mathbb{F}_2[x]/\langle x^6 + x^4 + x^3 + x + 1\rangle$ (véase la figura 2.4).*

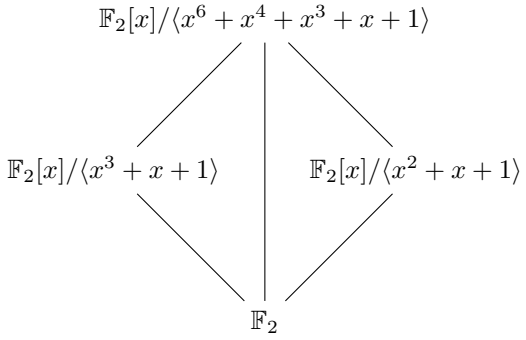

Figura 2.4: Torre de campos de grado 6 para $\mathbb{F}_2$.

Definición 2.4.14 *Sea* F *una extensión del campo* K*. Un elemento $\alpha \in$ F es* ***algebraico*** *sobre* F *si existe un polinomio distinto de cero $f \in \mathrm{K}[x]$ tal que $f(\alpha) = 0$. Si todo elemento en* F *es algebraico sobre* K*, entonces se dice que* F *es la* ***extensión algebraica****.*

Definición 2.4.15 *Un campo* K *es* ***algebraicamente cerrado*** *si cada polinomio irreducible sobre* K *tiene grado* 1*. Si* F *es una extensión algebraica de* K *y* F *es algebraicamente cerrado, entonces decimos que* F *es la* ***cerradura algebraica*** *de* K*. En la literatura matemática, la cerradura algebraica del campo* K *es denotada por $\overline{\mathrm{K}}$.*

Definición 2.4.16 *Sean* F *una extensión del campo* K*, $\alpha \in$ F un elemento algebraico sobre* K*. Entonces,* $\mathrm{K}(\alpha)$ *denota la extensión más pequeña del campo* K *que contiene al elemento α. En general, si se tiene un polinomio irreducible en* $\mathrm{K}[x]$ *de grado n, digamos f. Supongamos que $\alpha_1, \ldots, \alpha_n \in \overline{\mathrm{K}}$ denotan las raíces del polinomio f, entonces* $\mathrm{K}(\alpha_1, \ldots, \alpha_n)$ *está definido de manera análoga.*

El siguiente teorema es resultado del teorema 1.86 de [Lidl y Niederreiter, 1997].

Teorema 2.4.7 *Sean* F *una extensión del campo* K*, $\alpha \in$ F un elemento algebraico sobre* K *y* g *el polinomio irreducible de menor grado en* $\mathrm{K}[x]$ *que satisface* $\mathsf{g}(\alpha) = 0$*. Entonces,* $\mathrm{K}(\alpha) \cong \mathrm{K}[x]/\langle g \rangle$.

Retomando el contexto de campo finitos, la cerradura algebraica de un campo finito $\mathbb{F}_p$ está dada como la unión infinita de todas las extensiones finitas del campo $\mathbb{F}_p$:

$$\overline{\mathbb{F}}_p = \bigcup_{n>1} \mathbb{F}_{p^n}.$$

Definición 2.4.17 *Sea* $\mathrm{K} = \mathbb{F}_{q^n}$ *una extensión del campo finito* $\mathrm{k} = \mathbb{F}_q$ *de grado* n *donde* $q = p^l$ *para algún entero positivo* l*. Entonces, el* ***automorfismo de Frobenius****, denotado por* σ*, de* K *con respecto a* k *está definido de la siguiente manera:*

$$\begin{aligned} \sigma\colon \mathrm{K} &\mapsto \mathrm{K} \\ \gamma &\to \gamma^q \end{aligned}$$

Definición 2.4.18 *Sea* $\mathrm{K} = \mathbb{F}_{q^n}$ *una extensión del campo finito* $\mathrm{k} = \mathbb{F}_q$ *de grado* n *donde* $q = p^l$ *para algún entero positivo* l*. Entonces, la* ***traza*** *"con respecto al campo* k*" de un elemento* $\gamma \in \mathrm{K}$ *está definida de la siguiente manera:*

$$\mathrm{Tr}_{\mathrm{K/k}}(\gamma) = \sum_{i=0}^{n-1} \sigma^i(\gamma) = \sum_{i=0}^{n-1} \gamma^{q^i}.$$

Observación 2.4.1 *Nótese que* $\mathrm{Tr}_{\mathrm{K/k}}(\gamma) \in \mathrm{k}$ *para todo* $\gamma \in \mathrm{K}$*.*

2.5. Variedades algebraicas

Las siguientes definiciones, ejemplos y teoremas fueron obtenidos de las secciones 4.1 y 4.3 de [Cohen et al., 2012].

Dado un campo k, un **polinomio** f en las variables $x_1, \ldots, x_n$ con coeficientes en el campo k está definido de manera análoga que en el caso de una variable, es decir,

$$f(x) = \sum_{\alpha \in \mathrm{A}} a_\alpha x_1^{\alpha_1} \cdots x_n^{\alpha_n}, \quad a_\alpha \in \mathrm{k}$$

donde $\alpha = (\alpha_1, \ldots, \alpha_n)$, $|\mathrm{A}| < \infty$ y cada $\alpha_i \in \mathbb{N} \cup \{0\}$. El conjunto de todos los polinomios en las variables $x_1, \ldots, x_n$ con coeficientes en k, denotado por $\mathrm{k}[x_1, \ldots, x_n]$, forma un anillo mediante la adición y la multiplicación de polinomios.

Definición 2.5.1 *Sean* k *un campo y* $\mathrm{S} \subset \mathrm{k}[\mathrm{x_1}, \ldots, \mathrm{x_n}]$*,* ***la variedad afín*** *determinada por* S *está definida de la siguiente manera:*

$$V(\mathrm{S}) = \{a \in \mathrm{k}^n \colon f(a) = 0, f \in \mathrm{S}\},$$

es decir, $V(\mathrm{S})$ determina el conjunto de puntos en los que se anulan todos los polinomios que pertenecen a S.

Ejemplo 2.5.1 *Si* $\mathrm{S} = \{x^2 + y^2 - 1\} \subset \mathbb{R}[x, y]$ *entonces* $V(\mathrm{S}) = \{(\cos\theta, \mathrm{sen}\theta) \in \mathbb{R}^2 \colon \theta \in [0, 2\pi]\}$ *debido a que* $\cos(\theta + 2\pi) = \cos\theta$, $\mathrm{sen}(\theta + 2\pi) = \mathrm{sen}\theta$ *y* $\cos^2\theta + \mathrm{sen}^2\theta = 1$, $\forall\theta \in \mathbb{R}$.

Definición 2.5.2 *Una* ***variedad abeliana*** *de dimensión n es una variedad afín de dimensión n que tiene asociada una ley de grupo.*

Capítulo 3

Curvas elípticas

Todo el mundo sabe lo que es una curva, hasta que se ha estudiado las suficientes matemáticas para confundirse por la infinidad de posibles excepciones.

Felix Klein

De manera general, una curva elíptica está definida por una ecuación de dos variables x e y, conocida como ecuación de Weierstrass, con término cuadrático en la variable y y término cúbico en la variable x [Hankerson et al., 2003]. Ciertamente, es posible definir una operación binaria, de manera geométrica "natural", sobre el conjunto de puntos pertenecientes a la curva elíptica mediante la **regla de la secante** (suma de puntos) y la **regla de la tangente** (doblado de puntos). Por ejemplo, supongamos que tenemos una curva elíptica definida por la ecuación $y^2 = x^3 - 3x + 5$ sobre el campo de los número reales, $\mathbb{R}$, entonces la figura 3.1 muestra la idea geométrica de cómo está definida la operación binaria en la curva elíptica.

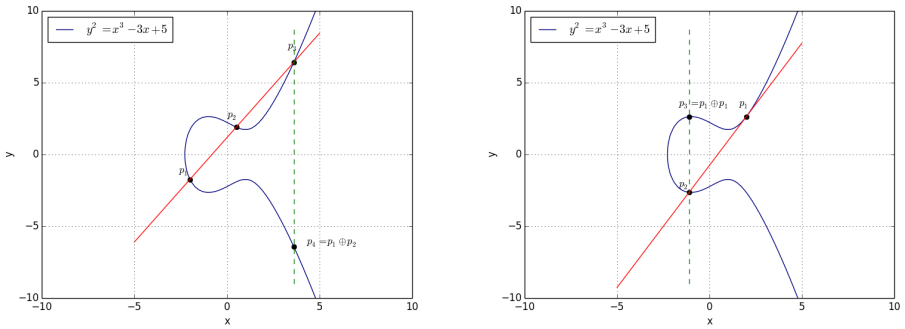

Figura 3.1: Suma y doblado de puntos en una curva elíptica dada por la ecuación $y^2 = x^3 - 3x + 5$ sobre $\mathbb{R}$.

Esta idea geométrica es aplicada algebraicamente, cuyo fin es expresar tanto la suma como el doblado de puntos mediante dos fórmulas. No obstante, para el caso de campos finitos la idea geométrica pierde un poco el sentido; por ejemplo, supongamos que tenemos una curva elíptica definida por la ecuación $y^2 = x^3 + 4x + 21$ sobre el campo finito $\mathbb{F}_{1777}$, entonces la figura 3.2 describe el conjunto de puntos pertenencientes a dicha curva y lo único que se puede observar, de manera natural, es la simetría en la figura.

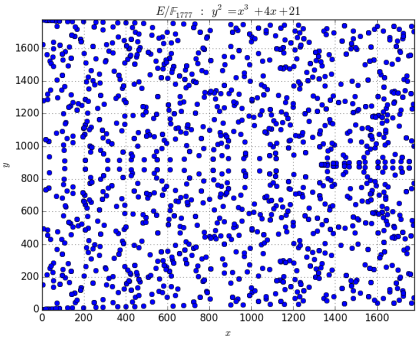

Figura 3.2: Curva elíptica dada por la ecuación $y^2 = x^3 + 4x + 21$ sobre $\mathbb{F}_{1777}$.

En la literatura criptográfica, una curva elíptica definida sobre un campo $\mathbb{F}_p$ es conocida como una **curva elíptica binaria** si $p = 2$; mientras que para el caso cuando $p > 2$, es denominada una **curva elíptica prima**.

El contenido de este capítulo ha sido organizado en las siguientes tres secciones: las secciones 3.1 y 3.2 describen tanto la familia de curvas elípticas que son de interés en esta tesis, como las propiedades que son de gran utilidad para el comprendimiento del ataque extendido gGHS; mientras que la sección 3.3 explica, de manera general, la aplicación de curvas elípticas hacia la criptografía de clave pública.

3.1. Curvas elípticas binarias

Las siguientes definiciones, ejemplos y teoremas provienen de la sección 3.1 de [Hankerson et al., 2003].

Dados dos números enteros positivos l, $q = 2^l$ y un campo binario $\mathrm{k} = \mathbb{F}_q$, una curva elíptica ordinaria $\mathcal{E}$ definida sobre el campo finito k, está dada por la siguiente ecuación simplificada de Weierstrass:

$$\mathcal{E}/\mathrm{k}\colon y^2 + xy = x^3 + ax^2 + b \tag{3.1}$$

donde $a \in \mathrm{k}$ y $b \in \mathrm{k}^\times$ (los elementos a y b son conocidos como **parámetros de la curva**). Sea K una extensión del campo k, entonces el conjunto $\mathcal{E}(\mathrm{K})$ de puntos K-racionales se define com el conjunto de puntos con coordenadas $x, y \in \mathrm{K}$ que satisfacen la ecuación 3.1 junto con un punto al infinito ∞, es decir,

$$\mathcal{E}(\mathrm{K}) = \{(x, y) \in \mathrm{K}^2 \colon y^2 + xy = x^3 + ax^2 + b\} \cup \{\infty\}.$$

El conjunto de puntos K-racionales forman un grupo abeliano teniendo como elemento identidad un punto al infinito ∞ y una ley de grupo definida por:

- **Suma de puntos.** Dados dos puntos distintos $P = (x_p, y_p)$ y $Q = (x_q, y_q)$ en $\mathcal{E}(\mathrm{K})$, la adición $P \oplus Q = (x, y) \in \mathcal{E}(\mathrm{K})$ se calcula como sigue:

$$\begin{aligned} \lambda &= \frac{y_p + y_q}{x_p + x_q}, \\ x &= \lambda^2 + \lambda + x_p + x_q + a, \\ y &= \lambda(x_p + x) + x + y_p. \end{aligned}$$

- **Doblado de puntos.** Dado un punto $P = (x_p, y_p) \in \mathcal{E}(\mathrm{K})$, el doblado $2P = P \oplus P = (x, y) \in \mathcal{E}(\mathrm{K})$ se calcula como sigue:

$$\begin{aligned} \lambda &= x_p + y_p/x_p, \\ x &= \lambda^2 + \lambda + a, \\ y &= x_p^2 + \lambda x + x. \end{aligned}$$

- **Inverso aditivo.** Dado un punto $P = (x_p, y_p) \in \mathcal{E}(\mathrm{K})$, el inverso aditivo de P está definido por $-P = (x_p, x_p + y_p) \in \mathcal{E}(\mathrm{K})$ y es tal que $P \oplus (-P) = \infty = (-P) \oplus P$.

- **Elemento identidad.** Para cualquier punto $P \in \mathcal{E}(\mathrm{K})$ en la curva elíptica $\mathcal{E}$, el punto al infinito ∞ es tal que $P \oplus \infty = P = \infty \oplus P$ y $\infty \oplus \infty = \infty$.

Dado un punto P en la curva elíptica $\mathcal{E}$ y un escalar $k \in \mathbb{N}$, la **multiplicación escalar** asociada está definida como:

$$kP = \underbrace{P \oplus \ldots \oplus P}_{k-1 \text{ veces}}.$$

Ejemplo 3.1.1 *Sean* $\mathrm{k} = \mathbb{F}_{2^7} = \mathbb{F}_2[u]/\langle u^7 + u + 1\rangle$ *y* $\mathcal{E}/\mathrm{k}$ *una curva elíptica binaria dada por la ecuación* $y^2 + xy = x^3 + (u^4 + 1)x^2 + (u^5 + u^4 + u)$. *Sean* $P = (u^6 + u^5 + u^2 + 1, u^3 + 1)$ *y* $Q = (u^4 + u^3 + 1, u^6 + u^5 + u^4 + u^3 + 1)$ *dos puntos* k*-racionales de la curva* $\mathcal{E}$*, entonces:*

- *Suma de puntos:* $P + Q = (u^6 + u + 1, u^6 + u^5 + u^4 + u^3 + u^2 + 1)$

$$\begin{aligned}
\lambda &= \frac{(u^3+1)+(u^6+u^5+u^4+u^3+1)}{(u^6+u^5+u^2+1)+(u^4+u^3+1)} = \frac{u^6+u^5+u^4}{u^6+u^5+u^4+u^3+u^2} \\
&= (u^6+u^5+u^4)(u^6+u^4+u^3+u) = u^5+u^4, \\
x &= (u^5+u^4)^2+(u^5+u^4)+(u^6+u^5+u^2+1)+(u^4+u^3+1)+(u^4+1) \\
&= (u^4+u^3+u^2+u)+(u^5+u^4)+(u^6+u^5+u^2+1)+(u^4+u^3+1) \\
&\quad +(u^4+1) = u^6+u+1, \\
y &= (u^5+u^4)\left((u^6+u^5+u^2+1)+(u^6+u+1)\right)+(u^6+u+1)+(u^3+1) \\
&= (u^5+u^4+u^2+u+1)+(u^6+u+1)+(u^3+1) \\
&= u^6+u^5+u^4+u^3+u^2+1.
\end{aligned}$$

- *Doblado de puntos:* $2P = (u^6 + u^5 + u^2 + u + 1, u^5 + u^4 + u^3 + u)$

$$\begin{aligned}
\lambda &= (u^6+u^5+u^2+1)+\frac{u^3+1}{u^6+u^5+u^2+1} \\
&= (u^6+u^5+u^2+1)+(u^3+1)(u^6+u^5+u^4+u^2+u) \\
&= (u^6+u^5+u^2+1)+(u^6+u^3+u^2+u+1) = u^5+u^3+u, \\
x &= (u^5+u^3+u)^2+(u^5+u^3+u)+(u^4+1) \\
&= (u^6+u^4+u^3+u^2)+(u^5+u^3+u)+(u^4+1) = u^6+u^5+u^2+u+1, \\
y &= (u^6+u^5+u^2+1)^2+(u^5+u^3+u)(u^6+u^5+u^2+u+1) \\
&\quad +(u^6+u^5+u^2+u+1) \\
&= (u^6+u^5+u^3+1)+(x^5+x^4+x^2)+(u^6+u^5+u^2+u+1) \\
&= u^5+u^4+u^3+u.
\end{aligned}$$

- *Multiplicación escalar:* $7Q = (u^6 + u^5 + u^4 + u^2 + u + 1, u^6 + u^5 + u^4 + u^2)$.

Teorema 3.1.1 (**Intervalo de Hasse**) *Sean l un enetero positivo, $q = 2^l$ y una curva elíptica binaria $\mathcal{E}/\mathbb{F}_q$, entonces:*

$$q + 1 - 2\sqrt{q} \leq |\mathcal{E}(\mathbb{F}_q)| \leq q + 1 + 2\sqrt{q}.$$

Ejemplo 3.1.2 *Retomando la curva elíptica binaria, $\mathcal{E}/\mathbb{F}_{2^7}$, usada en el ejemplo 3.1.1, se puede verificar que $|\mathcal{E}(\mathbb{F}_{2^7})| = 146$, $2^7 + 1 - 2\sqrt{2^7} \approx 106$ y $2^7 + 1 + 2\sqrt{2^7} \approx 152$, es decir,*

$$2^7 + 1 - 2\sqrt{2^7} \leq |\mathcal{E}(\mathbb{F}_{2^7})| \leq 2^7 + 1 + 2\sqrt{2^7}.$$

3.2. Isogenias entre curvas elípticas

Las siguientes definiciones, ejemplos y teoremas provienen de la sección 5.4 [Galbraith, 2012] y de [Tate, 1966].

Dados dos números enteros positivos l, $q = 2^l$, un campo binario $\mathrm{k} = \mathbb{F}_q$ y una curva elíptica binaria $\mathcal{E}/\mathrm{k}$. El **campo de funciones racionales** de la curva $\mathcal{E}$ sobre el campo k, denotado por $\mathrm{k}(\mathcal{E})$, está definido como el campo de

fracciones del anillo $\mathrm{k}[\mathcal{E}] = \mathrm{k}[x,y]/\langle y^2 + xy + x^3 + ax^2 + b\rangle$ donde a y b son los paramétros de la curva $\mathcal{E}$, es decir,

$$\mathrm{k}(\mathcal{E}) = \left\{ \frac{\mathrm{pol}_1(x,y)}{\mathrm{pol}_2(x,y)} \middle| \forall \mathrm{pol}_1(x,y), \mathrm{pol}_2(x,y) \in \mathrm{k}[\mathcal{E}] \colon \mathrm{pol}_2(x,y) \neq 0 \right\}. \tag{3.2}$$

Definición 3.2.1 *Sean l un entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$, $\mathcal{E}/\mathrm{k}$ y $\mathcal{E}'/\mathrm{k}$ dos curvas elípticas binarias. Una **proyección racional**, $\psi \colon \mathcal{E} \to \mathcal{E}'$ sobre el campo* k*, es un elemento de la curva elíptica $\mathcal{E}'(\mathrm{k}(\mathcal{E}))$, es decir, $\psi(x,y) = (\psi_1(x,y), \psi_2(x,y))$ donde $\psi_1, \psi_2 \in \mathrm{k}(x,y)$ y para cada punto $(u,v) \in \mathcal{E}(\mathrm{k})$ se tiene que $\psi(u,v) \in \mathcal{E}'(\mathrm{k})$.*

Ejemplo 3.2.1 *Sean l un entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$, $\mathcal{E}/\mathrm{k}$ y $\mathcal{E}'/\mathrm{k}$ dos curvas elípticas binarias y P' un punto* k*-racional de la curva elíptica binaria $\mathcal{E}'$. Entonces,*

$$\begin{aligned} \psi \colon \mathcal{E} &\to \mathcal{E}' \\ (x,y) &\mapsto P' \end{aligned}$$

*determina una proyección racional conocido como **proyección constante**.*

Ejemplo 3.2.2 *Sean l un entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$ y $a,b,c \in \mathrm{k}^\times$. Supongamos que tenemos dos curvas elípticas binarias $\mathcal{E}/\mathrm{k}$ y $\mathcal{E}'/\mathrm{k}$ dadas por las ecuaciones $y^2 + xy = x^3 + ax^2 + b$ e $y^2 + xy = x^3 + (a + c^{-2} + x^{-4})x^2 + b$, respectivamente. Entonces,*

$$\begin{aligned} \psi \colon \mathcal{E} &\to \mathcal{E}' \\ (x,y) &\mapsto (x, y + c^{-1}x) \end{aligned}$$

determina una proyección racional entre las curvas elípticas $\mathcal{E}$ y $\mathcal{E}'$.

Definición 3.2.2 *Sean l un entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$, $\mathcal{E}/\mathrm{k}$ y $\mathcal{E}'/\mathrm{k}$ dos curvas elípticas binarias. Una **isogenia**, $\varphi \colon \mathcal{E} \to \mathcal{E}'$ sobre el campo* k*, es una proyección racional no-constante, el cual es un homomorfismo de grupos entre $\mathcal{E}(k)$ y $\mathcal{E}'(k)$. Se dice que las curvas elípticas $\mathcal{E}$ y $\mathcal{E}'$ son **curvas isógenas** sobre el campo* k *si existe una isogenia entre ellas dos.*

Ejemplo 3.2.3 *Sean l un entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$ y $\mathcal{E}/\mathrm{k}$ una curva elíptica binaria. Entonces, el automorfismo de Frobenius en $\mathrm{k} = \mathbb{F}_q$ induce una isogenia, la cual es descrita mediante el siguiente proyección racional.*

$$\begin{aligned} \psi \colon \mathcal{E} &\to \mathcal{E} \\ (x,y) &\mapsto (x^q, y^q). \end{aligned}$$

Teorema 3.2.1 *Sean l un entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$, $\mathcal{E}/\mathrm{k}$ y $\mathcal{E}'/\mathrm{k}$ dos curvas elípticas binarias. Entonces $\mathcal{E}$ y $\mathcal{E}'$ son curvas isógenas sobre el campo* k *si y sólo si $|\mathcal{E}(\mathrm{k})| = |\mathcal{E}'(\mathrm{k})|$.*

Ejemplo 3.2.4 *Sean* $\mathrm{k} = \mathbb{F}_{2^7} = \mathbb{F}_2[u]/\langle u^7 + u + 1\rangle$, $\mathcal{E}/\mathrm{k}$ *y* $\mathcal{E}'/\mathrm{k}$ *dos curvas elípticas binarias dadas por las ecuaciones* $y^2 + xy = x^3 + (u^4+1)x^2 + (u^6+u)$ *e* $y^2 + xy = x^3 + (u^4+1)x^2 + (u^4+u^3+u)$*, respectivamente. Entonces, se puede verificar que* $|\mathcal{E}(\mathrm{k})| = 146 = |\mathcal{E}'(\mathrm{k})|$ *y por consiguiente* $\mathcal{E}$ *y* $\mathcal{E}$*' son curvas isógenas sobre el campo* k*.*

Definición 3.2.3 *Sean* l *un entero positivo,* $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$ *y* $\mathcal{E}/\mathrm{k}$ *una curva elíptica binaria. La* ***clase de isogenia*** *de* $\mathcal{E}$ *se refiere al conjunto de todas las curvas elípticas que son isógenas a* $\mathcal{E}$*.*

3.3. Problema del Logaritmo Discreto en curvas elípticas

Dados dos números enteros positivos l, $q = 2^l$, un campo binario $\mathrm{k} = \mathbb{F}_q$ y una curva elíptica binaria $\mathcal{E}/\mathrm{k}$. El Problema del Logaritmo Discreto en $\mathcal{E}$ está definido como el Problema del Logaritmo Discreto en el subgrupo maximal del grupo de puntos k-racionales de $\mathcal{E}$, es decir, dados un punto $P \in \mathcal{E}(\mathrm{k})$ de mayor orden primo, r, y un punto $Q \in \langle P\rangle$ hallar $\lambda \in 1, 2, \ldots, r$ tal que $Q = \lambda P$.

En la actualidad, es conocido que el mejor algoritmo para resolver el Problema del Logaritmo Discreto en la curva elíptica $\mathcal{E}$ (el *algoritmo* ρ *de Pollard* [Pollard, 1978]), requiere aproximadamente $O\left(\sqrt{\frac{\pi r}{2}}\right)$ operaciones en la curva elíptica.

Definición 3.3.1 *Sean* l *un entero positivo,* $q = 2^l$, $\mathrm{k} = \mathbb{F}_q$ *y* $\mathcal{E}/\mathrm{k}$ *una curva elíptica binaria y* r *el orden del subgrupo maximal de* $\mathcal{E}(\mathrm{k})$*. El* ***grado de encajamiento*** *de* r *en* $\mathcal{E}(\mathrm{k})$ *está definido como el mínimo entero positivo* $d \geq 1$ *que satisface* $q^d \equiv 1 \mod r$*.*

Ejemplo 3.3.1 *Retomando la curva elíptica binaria,* $\mathcal{E}/\mathbb{F}_{2^7}$*, usada en el ejemplo 3.2.4, tenemos* $r = 73$ *y* $q^9 = (2^7)^9 \equiv 1 \mod 73$*. Por lo tanto, el grado de encajamiento de* $r = 73$ *en* $\mathcal{E}(\mathbb{F}_{2^7})$ *es* 9*.*

El ataque MOV. Este ataque fue creado por Menezes, Okamoto y Vanstone (1993), el cual traslada una instancia del Problema del Logaritmo Discreto en la curva elíptica $\mathcal{E}$ hacia una estancia del Problema del Logaritmo Discreto en $\mathrm{F}^\times$ donde F es una extensión de grado m del campo k y m es el grado de encajamiento de r en $\mathcal{E}(\mathrm{k})$. Este ataque es sumamente efectivo en curvas elípticas con grados de encajamientos "pequeños", debido a que los mejores algoritmo (los algoritmos QPA's [Barbulescu et al., 2014, Granger et al., 2014]) para la resolución del Problema del Logaritmo Discreto requiere aproximadamente $O\left((\log|\mathrm{F}|)^{\log\log|\mathrm{F}|}\right)$ operaciones en el campo finito F.

Observación 3.3.1 *Nótese que si* r *y* m *son "lo suficientemente grandes", entonces el algoritmo* ρ *de Pollard y el ataque MOV son imprácticos.*

Intercambio de claves de Diffie-Hellman en curvas elípticas

Este sistema de intercambio de claves, el cual es descrito mendiante la figura 3.3, se basa en la idea de que dos entidades pueden generar conjuntamente una clave compartida sin que un adversario, que está escuchando las comunicaciones, pueda llegar a obtenerla.

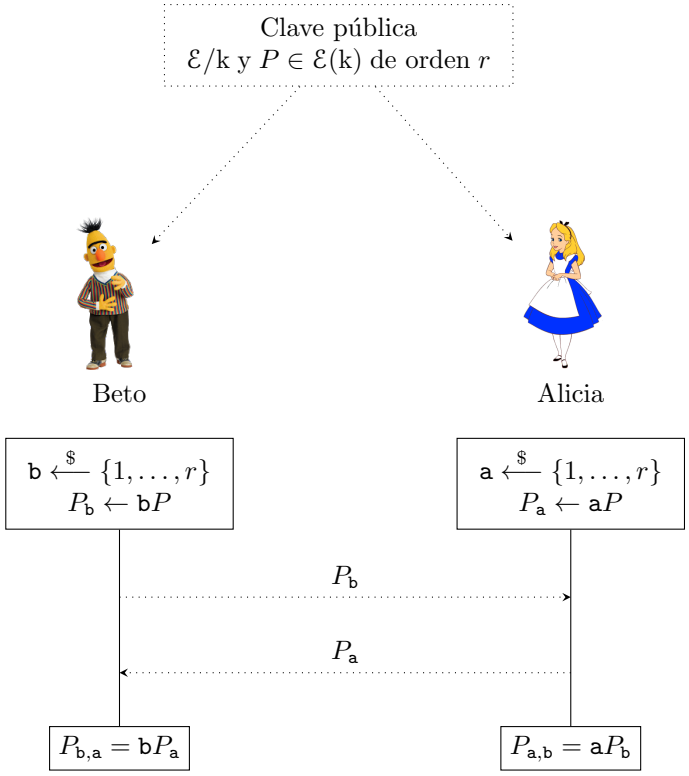

Figura 3.3: Intercambio de claves de Diffie-Hellman en curva elípticas

La seguridad de este sistema de intercambio de claves, radica en la dificultad en resolver el problema de Diffie-Hellman en el subgrupo generado por P, es decir, obtener $(\mathtt{ab})P$ dado que se conoce $\mathtt{a}P$ y $\mathtt{b}P$. En particular, si es posible resolver, con un número polinomial de operaciones en la curva, el Problema del Logaritmo Discreto en la curva elíptica entonces el problema de Diffie-Hellman puede ser resuelto con un número polinomial de operaciones en la curva.

Capítulo 4

Curvas hiperelípticas

En la ciencia estamos consentidos por el éxito de las matemáticas. Las matemáticas son el estudio de problemas tan simples que tienen buenas soluciones.

Whitfield Diffie

En términos generales, una curva hiperelíptica es una "generalización" de una curva elíptica con la limitante de que las reglas de la secante y tangente no definen "de manera natural geométrica" una ley de grupo. Esto es debido a que la unicidad del tercer punto de intersección generado entre la curva hiperelíptica y la respectiva línea recta secante, o tangente, puede no ser satisfecha; por ello mismo se suele trabajar con el *jacobiano* de la curva hiperelíptica el cual tiene asociada una ley de grupo (véase la sección 4.1). Por ejemplo, supongamos que tenemos una curva hiperelíptica dada por la ecuación $y^2 = x^5 - 2x^4 - 7x^3 + 8x^2 + 12x$ definida sobre el campo de los números reales, entonces en la figura 4.1 se puede observar que la línea recta secante generada por dos puntos sobre la curva, no interseca en un único tercer punto a la curva (como en el caso elíptico).

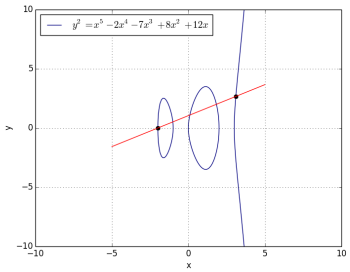

Figura 4.1: Curva hiperelíptica dada por la ecuación $y^2 = x^5 - 2x^4 - 7x^3 + 8x^2 + 12x$ sobre $\mathbb{R}$.

Al igual que en el caso de curvas elípticas, una curva hiperelíptica definida sobre un campo $\mathbb{F}_p$ es conocida como una **curva hiperelíptica binaria** si $p = 2$; mientras que para el caso cuando $p > 2$, es denominada una **curva hiperelíptica prima**.

El contenido de este capítulo ha sido organizado en las siguientes dos secciones: la sección 4.1 detalla, de manera concisa, las definiciones de curva hiperelíptica, el conjunto de divisores y el jacobiano de la curva hiperelíptica; mientras que la sección 4.2 explica, de manera general, tanto el Problema del Logaritmo Discreto en curvas hiperelípticas como los algoritmos que son usados para la resolución de este.

4.1. Curvas hiperelípticas binarias

Las siguientes definiciones, propocisiones y teoremas fueron tomados del capítulo 7 de [Blake et al., 2005].

Dados dos números enteros positivos l, $q = 2^l$ y un campo binario $\mathrm{k} = \mathbb{F}_q$, una curva hiperelíptica $\mathcal{H}$ de género g definida sobre el campo finito k, está dada por la siguiente ecuación:

$$\mathcal{H}/\mathrm{k}\colon y^2 + h(x)y = f(x) \tag{4.1}$$

donde $h, f \in k[x]$, $\mathrm{grad} f = 2g + 1$ y $\mathrm{grad} h \leq g$. Sea K una extensión del campo k, entonces el conjunto $\mathcal{H}(\mathrm{K})$ de puntos K-racionales se define com el conjunto de puntos con coordenadas $x, y \in \mathrm{K}$ que satisfacen la ecuación 4.1 junto con un punto al infinito ∞, es decir,

$$\mathcal{H}(\mathrm{K}) = \{(x, y) \in \mathrm{K}^2\colon y^2 + h(x)y = f(x)\} \cup \{\infty\}.$$

El negativo de un punto $P = (x, y) \in \mathcal{H}(\mathrm{k})$ está definido como el punto $-P = (x, -y - h(x))$.

Ejemplo 4.1.1 *Toda curva elíptica binaria es una curva hiperelíptica binaria de género* 1.

Ejemplo 4.1.2 *Sea* $\mathbb{F}_{2^7} = \mathbb{F}_2[u]/\langle u^7 + u + 1\rangle$ *y sea* $\mathcal{H}/\mathbb{F}_{2^7}$ *una curva dada por la ecuación* $y^2 + (x^4 + x + 1)y = x^9 + x^4 + 1$. *Entonces,* $\mathcal{H}/\mathbb{F}_{2^7}$ *es una curva hiperelíptica de género* 4.

Ejemplo 4.1.3 *Sea* $\mathbb{F}_{2^7} = \mathbb{F}_2[u]/\langle u^7 + u + 1\rangle$ *y sea* $\mathcal{H}/\mathbb{F}_{2^7}$ *una curva dada por la ecuación* $y^2 + (x^{15} + x^2 + 1)y = x^{31} + x^{12} + x + 1$. *Entonces,* $\mathcal{H}/\mathbb{F}_{2^7}$ *es una curva hiperelíptica de género* 15.

De manera analoga al caso de curvas elípticas binarias, el **campo de funciones racionales** de la curva $\mathcal{H}$ sobre el campo k, denotado por $\mathrm{k}(\mathcal{H})$, está

definido como el campo de fracciones del anillo $\mathrm{k}[\mathcal{H}] = \mathrm{k}[x,y]/\langle y^2+h(x)y-f(x)\rangle$, es decir,

$$\mathrm{k}(\mathcal{H}) = \left\{ \frac{\mathrm{pol}_1(x,y)}{\mathrm{pol}_2(x,y)} \middle| \forall \mathrm{pol}_1(x,y), \mathrm{pol}_2(x,y) \in \mathrm{k}[\mathcal{H}] \colon \mathrm{pol}_2(x,y) \neq 0 \right\}. \quad (4.2)$$

El jacobiano de una curva hiperelíptica

El conjunto de divisores, $\mathrm{Div}_{\mathcal{H}}(\mathrm{k})$, de una curva hiperelíptica binaria, $\mathcal{H}/\mathrm{k}$, está definido como el conjunto de sumas formales finitas de puntos k-racionales de $\mathcal{H}$, es decir:

$$\mathrm{Div}_{\mathcal{H}}(\mathrm{k}) = \left\{ \mathrm{D} = \sum_{P_i \in \mathcal{H}(\mathrm{k})} n_i\,(P_i) : n_i \in \mathbb{Z} \text{ y } n_i = 0 \text{ para casi toda } i \right\}. \quad (4.3)$$

El **grado** de un divisor $\mathrm{D} = \sum n_i\,(P_i)$, denotado por $\mathrm{grad}\,(\mathrm{D})$, está definido como la suma de los coeficientes, es decir, $\mathrm{grad}\,(\mathrm{D}) = \sum n_i$; mientras que el conjunto de puntos P_i para los cuales $n_i \neq 0$ es llamado el **soporte** de D y es denotado por $Spt(\mathrm{D})$. Cabe mencionar que es posible asociar "de manera natural" una operación binaria en $\mathrm{Div}_{\mathcal{H}}(\mathrm{k})$ mediante la suma a coeficientes, es decir, dados dos divisores $\mathrm{D}_1 = \sum n_i(P_i)$ y $\mathrm{D}_2 m_i(P_i)$ entonces $\mathrm{D}_1 + \mathrm{D}_2 = \sum (n_i + m_i)(P_i)$.

Teorema 4.1.1 *Sean l un número entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_{2^l}$ y una curva hiperelíptica binaria $\mathcal{H}/\mathrm{k}$. Entonces, $\mathrm{Div}_{\mathcal{H}}(\mathrm{k})$ forma un grupo abeliano y el conjunto de divisores de grado cero, denotado por $\mathrm{Div}^0_{\mathcal{H}}(\mathrm{k})$, es un subgrupo de $\mathrm{Div}_{\mathcal{H}}(\mathrm{k})$.*

Definición 4.1.1 *Sean l un número entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_{2^l}$, $\mathcal{H}/\mathrm{k}$ una curva hiperelíptica, $\mathrm{k}\,(\mathcal{H})$ el campo de funciones racionales de $\mathcal{H}$ y $\varphi \in \mathrm{k}\,(\mathcal{H})^{\times}$. Entonces, el divisor $\mathrm{div}(\varphi) = \sum_{P_i \in \mathcal{H}(\mathrm{k})} \nu_{P_i}(\varphi)(P_i)$ donde $\nu_{P_i}(\varphi)$ es un número entero definido de la siguiente manera:*

$$\nu_{P_i}(\varphi) = \begin{cases} \textit{la multiplicidad de } P_i & \textit{si } \varphi \textit{ tiene un cero en } P_i \\ \textit{el negativo de la multiplicidad de } P_i & \textit{si } \varphi \textit{ tiene un polo en } P_i \\ 0 & \textit{en otro caso,} \end{cases}$$

*es llamado **divisor principal**.*

Teorema 4.1.2 *Sean l un número entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_{2^l}$ y una curva hiperelíptica binaria $\mathcal{H}/\mathrm{k}$. Entonces, el conjunto de divisores principales, denotado por $\mathrm{Prin}_{\mathcal{H}}(\mathrm{k})$ es un subgrupo de $\mathrm{Div}^0_{\mathcal{H}}(\mathrm{k})$.*

Definición 4.1.2 *Sean l un número entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_{2^l}$ y $\mathcal{H}/\mathrm{k}$ una curva hiperelíptica. Entonces, el **jacobiano** de la curva $\mathcal{H}/\mathrm{k}$, el cual es denotado por $\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$, está definido como el grupo cociente*

$$\mathrm{Jac}_{\mathcal{H}}(\mathrm{k}) = \mathrm{Div}^0_{\mathcal{H}}(\mathrm{k})/\mathrm{Prin}_{\mathcal{H}}(\mathrm{k}). \quad (4.4)$$

Una consecuencia del Teorema de Riemann-Roch es el hecho de que cada elemento del jacobiano de una curva hiperelíptica $\mathcal{H}/\mathrm{k}$ de género g puede ser representado por un divisor de la forma

$$\mathrm{D} = (P_1) + (P_2) \cdots + (P_r) - r(\infty) \tag{4.5}$$

donde $P_i \in \mathcal{H}(\mathrm{k})$ para cada $i = 1, \ldots$ y $r \leq g$ es un entero positivo. Si sucede que $P_i \neq -P_j$ para toda $i \neq j$, entonces el divisor D es llamado **divisor reducido** y puede ser representado por un único par de polinomios $u, v \in \mathrm{k}[x]$ con las siguientes propiedades:

$$i) \quad \mathrm{grad}v < \mathrm{grad}u \leq g, \tag{4.6}$$

$$ii) \quad u \text{ es un polinomio mónico}, \tag{4.7}$$

$$iii) \quad u|(v^2 + vh - f). \tag{4.8}$$

A esta representación se le conoce como **representación de Mumford** y el respectivo elemento en el jacobiano correspondiente a los polinomios u y v es denotado por $\mathrm{div}(u, v)$. Cuando el polinomio u es irreducible en $\mathrm{k}[x]$, el divisor $\mathrm{div}(u, v)$ es llamado **divisor primo**. La aritmética en el jacobiano, bajo esta representación, es realizada mediante el algoritmo de Cantor (si se desea profundizar y comprender el porqué el algoritmo de Cantor funciona, véase [Cantor, 1987]).

Input: $\mathrm{D}_1, \mathrm{D}_2 \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ en representación de Mumford, es decir, $\mathrm{D}_i = \mathrm{div}(u_i, v_i)$
Output: $\mathrm{D}_3 = \mathrm{D}_1 + \mathrm{D}_2$ en representación de Mumford

1 $d_1 \leftarrow \mathrm{mcd}(u_1, u_2)$; // $d_1 = e_1u_1 + e_2u_2$
2 $d \leftarrow \mathrm{mcd}(d_1, v_1 + v_2 + h)$; // $d = c_1d_1 + c_2(v_1 + v_2 + h)$
3 $s_1 \leftarrow c_1e_1$, $s_2 \leftarrow c_1e_2$, $s_3 \leftarrow c_2$;
4 $u_3 \leftarrow \frac{u_1u_2}{d^2}$, $v_3 \leftarrow \frac{s_1u_1v_2 + s_2u_2v_1 + s_3(v_1v_2 + f)}{d} \mod u_3$;
5 **repeat**
6 $\quad u_3 \leftarrow \frac{f - v_3h - v_3^2}{u_3}$, $v_3 \leftarrow (-h - v_3) \mod u_3$;
7 **until** $\mathrm{grad}u_3 \leq g$;
8 hacer mónico al polinomio u_3;
9 **return** $\mathrm{D}_3 = \mathrm{div}(u_3, v_3)$

Algoritmo 4.1: Algoritmo de Cantor

Ejemplo 4.1.4 *Retomando la curva hiperelíptica binaria, $\mathcal{H}/\mathbb{F}_{2^7}$ de género 4, usada en el ejemplo 4.1.2, sean $\mathrm{D}_1 = \mathrm{div}(x + (u^5 + u^4 + u + 1), u^2 + 1)$ y $\mathrm{D}_2 = \mathrm{div}(x + (u^6 + u^5 + u^4 + 1), u^6 + u^4 + u^3 + u^2)$ dos divisores en representación de Mumford. Entonces, mediante el algoritmo de Cantor, se puede verificar que el divisor $\mathrm{D}_3 = \mathrm{D}_1 + \mathrm{D}_2$, es igual a:*

$$\mathrm{div}\left(x^2 + (u^6 + u)x + (u^4 + u^2 + u), (u^3 + u + 1)x + (u^6 + u^3 + u^2 + 1)\right).$$

Proposición 4.1.1 *Sean l un número entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_{2^l}$, $\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ el jacobiano de una curva hiperelíptica $\mathcal{H}/\mathrm{k}$ y $\mathrm{D} = \mathrm{div}(u.v) \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ un divisor reducido en represantación de Mumford. Supongamos que $u(x)$ se factoriza como el producto de w polinomios irreducibles en $\mathrm{k}[x]$, es decir, $u = \prod_{i=1}^{w} u_i$ y cada $u_i \in \mathrm{k}[x]$ es irreducible. Entonces, $\mathrm{D}_i = \mathrm{div}(u_i.v_i)$ con $v_i = v \mod u_i$ es un divisor primo y $\mathrm{D} = \sum_{i=1}^{w} \mathrm{D}_i$.*

Ejemplo 4.1.5 *Retomando la curva hiperelíptica binaria, $\mathcal{H}/\mathbb{F}_{2^7}$ de género 4, usada en el ejemplo 4.1.2. Si $\mathrm{D} = \mathrm{div}(\mathtt{u}, \mathtt{v}) \in \mathrm{Jac}_{\mathcal{H}}(\mathbb{F}_{2^7})$ es un divisor en representación de Mumford con*

$$\begin{aligned}
\mathtt{u} &= x^4 + (u^5+u^4+u^3+u^2)x^3 + (u^6+u^5+u^4+u^2+u)x^2 \\
&\quad +(u^5+u^4+u^3+u^2+1)x + (u^5+u^4+u^3+u^2), \\
\mathtt{v} &= (u^5+u^4+u^2+u)x^3 + (u^6+u^4+u^3+u^2+1)x^2 \\
&\quad +(u^6+u^5+u^4+u^3+u^2)x + u^6.
\end{aligned}$$

entonces, el polinomio $\mathtt{u}$ se descompone en $\mathbb{F}_{2^7}[x]$ como el producto de los siguientes cuatro polinomios lineales:

$$\begin{aligned}
\mathtt{u}_1 &= x + (u^5+u^4+u+1), \\
\mathtt{u}_2 &= x + (u^5+u^4+u^3+u^2+u+1), \\
\mathtt{u}_3 &= x + (u^6+u^5+u^4+1), \\
\mathtt{u}_4 &= x + (u^6+1).
\end{aligned}$$

y $\mathrm{D} = \sum_{i=1}^{4} \mathrm{div}(\mathtt{u}_i, \mathtt{v}_i)$ donde

$$\begin{aligned}
\mathtt{v}_1 &= \mathtt{v} \mod \mathtt{u}_1 = u^2+1, \\
\mathtt{v}_2 &= \mathtt{v} \mod \mathtt{u}_2 = u^2+1, \\
\mathtt{v}_3 &= \mathtt{v} \mod \mathtt{u}_3 = u^6+u^4+u^3+u^2, \\
\mathtt{v}_4 &= \mathtt{v} \mod \mathtt{u}_4 = u^5+u^3+u^2+u.
\end{aligned}$$

Proposición 4.1.2 *Sean l un número entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_{2^l}$, $\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ el jacobiano de una curva hiperelíptica $\mathcal{H}/\mathrm{k}$ y $\mathrm{D} = (P_1) + (P_2) \cdots + (P_r) - r(\infty) \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ un divisor reducido. Supongamos que $P_i = (x_i, y_i)$ para cada $i \in \{1, \ldots, r\}$, entonces $\mathrm{D} = \mathrm{div}(u, v)$ donde $u(x) = \prod_{1 \le i \le r}(x - x_i)$ y $v(x)$ es el único polinomio de grado menor a r que satisface $v(x_i) = y_i$.*

Ejemplo 4.1.6 *Sean l un número entero positivo, $q = 2^l$, $\mathrm{k} = \mathbb{F}_{2^l}$, $\mathcal{H}/\mathrm{k}$ una curva hiperelíptica binaria, $\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ su respectivo jacobiano y $P = (P_x, P_y) \in \mathcal{H}(\mathrm{k})$. Entonces, la representación de Mumford del divisor $\mathrm{D} = (P) - (\infty) \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ es $\mathrm{div}(x - P_x, P_y)$.*

Ejemplo 4.1.7 *Retomando la curva hiperelíptica binaria, $\mathcal{H}/\mathbb{F}_{2^7}$ de género 4, usada en el ejemplo 4.1.2. Entonces, el divisor*

$$\mathrm{D}_3 = \left(u^5+u^4+u+1, u^2+1\right) + \left(u^6+u^5+u^4+1, u^6+u^4+u^3+u^2\right) - 2\,(\infty)$$

tiene la siguiente representación de Mumford

$$\operatorname{div}\left(x^2+(u^6+u)x+(u^4+u^2+u),(u^3+u+1)x+(u^6+u^3+u^2+1)\right).$$

Teorema 4.1.3 (**Generalización del intervalo de Hasse**) *Sean l un entero positivo, $q=2^l$ y $\mathrm{Jac}_{\mathcal{H}}(\mathbb{F}_q)$ el jacobiano de una curva hiperelíptica $\mathcal{H}/\mathbb{F}_q$. Entonces:*

$$\begin{aligned}(\sqrt{q}-1)^{2g}\leq|\mathrm{Jac}_{\mathcal{H}}(\mathbb{F}_q)| &\leq (\sqrt{q}+1)^{2g},\\ ||\mathcal{H}(\mathbb{F}_q)|-(q+1)| &\leq 2g\sqrt{q}.\end{aligned}$$

donde g determina el género de la curva hiperelíptica $\mathcal{H}/\mathbb{F}_q$.

Ejemplo 4.1.8 *Retomando la curva hiperelíptica binaria, $\mathcal{H}/\mathbb{F}_{2^7}$ de género 4, usada en el ejemplo 4.1.2, se puede verificar que $|\mathcal{H}(\mathbb{F}_{2^7})|=85$, $2\cdot 4\sqrt{2^7}=90.51$, $|\mathrm{Jac}_{\mathcal{H}}(\mathbb{F}_{2^7})|=188450252$, $\left(\sqrt{2^7}-1\right)^{2\cdot 4}\approx 128032085.77$ y $\left(\sqrt{2^7}+1\right)^{2\cdot 4}\approx 528580268.22$, es decir,*

$$\begin{aligned}\left(\sqrt{2^7}-1\right)^{2\cdot 4}\leq|\mathrm{Jac}_{\mathcal{H}}(\mathbb{F}_{2^7})| &\leq \left(\sqrt{2^7}+1\right)^{2\cdot 4},\\ ||\mathcal{H}(\mathbb{F}_{2^7})|-(2^7+1)| &\leq 2\cdot 4\sqrt{2^7}.\end{aligned}$$

Definición 4.1.3 *Sean l un número entero positivo, $q=2^l$, $\mathrm{k}=\mathbb{F}_{2^l}$, $\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ el jacobiano de una curva hiperelíptica $\mathcal{H}/\mathrm{k}$ y $\mathrm{D}=(u,v)\in\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ un divisor en representación de Mumford. Entonces, el divisor D es llamado m-**suave** si el polinomio $u\in\mathrm{k}[x]$ es m-**suave**, es decir, si u se decompone en $\mathrm{k}[x]$ como el producto de factores de grados menores o iguales a m. Cuando $m=1$, el divisor es llamado **suave**.*

Ejemplo 4.1.9 *El divisor $\mathrm{D}=\mathrm{div}(\mathtt{u},\mathtt{v})$ del ejemplo 4.1.5 es un divisor suave.*

Ejemplo 4.1.10 *Retomando la curva hiperelíptica binaria, $\mathcal{H}/\mathbb{F}_{2^7}$ de género 4, usada en el ejemplo 4.1.2. Entonces, el divisor $\mathrm{D}=\mathrm{div}(\mathtt{u},\mathtt{v})$ con*

$$\begin{aligned}\mathtt{u} &= x^4+(u^6+u^5+u^3+u^2+u)x^3+(u^5+u^3+u+1)x^2\\ &\quad +(u^6+u^5+u^3+u^2+u)x+(u^6+u^5+u^2+1),\\ \mathtt{v} &= (u^5+u^4+u^3+u^2+1)x^3+(u^6+u^5+u^4+u^3+u^2+u+1)x^2\\ &\quad +(u^5+1)x+(u^6+u^3+u+1).\end{aligned}$$

es un divisor 4-suave debido a que el polinomio $\mathtt{u}$ es irreducible en $\mathbb{F}_{2^7}[x]$.

4.2. Problema del Logaritmo Discreto en curvas hiperelípticas

Dados dos números enteros positivos l, $q=2^l$, un campo binario $\mathrm{k}=\mathbb{F}_q$ y el jacobiano $\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ de la curva hiperelíptica $\mathcal{H}/\mathrm{k}$ de género g, entonces

el Problema del Logaritmo Discreto en $\mathcal{H}$ está definido como el Problema del Logaritmo Discreto en el subgrupo maximal del jacobiano $\mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$, es decir, dados un divisor $\mathrm{D} \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ de mayor orden primo, r, y un divisor $\mathrm{D}' \in \langle \mathrm{D} \rangle$ hallar $\lambda \in 1, 2, \ldots, r$ tal que $\mathrm{D}' = \lambda \mathrm{D}$.

Ciertamente, el Problema del Logaritmo Discreto en $\mathcal{H}$ puede ser resuelto mediante el algoritmo ρ de Pollard, el cual requiere aproximadamente $O\left(\sqrt{\frac{\pi q^g}{2}}\right)$ operaciones en el jacobiano de la curva hiperelíptica; aunque por otro lado, los algoritmos propuesto en la literatura criptográfica para la resolución de dicho problema, son algoritmos basados en el **algoritmo de cálculo de índices** en su versión hiperelíptica (el cual es descrito de manera general mediante el algoritmo 4.2).

Input: un divisor $\mathrm{D} \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ de mayor orden primo r y un divisor $\mathrm{D}' \in \langle \mathrm{D} \rangle$
Output: un número entero $\lambda \in \{1, 2, \ldots, r\}$ tal que $\mathrm{D}' = \lambda \mathrm{D}$
1 fijar una cota de suavidad m y construir la base de factores;
2 **while** *no hayan las suficientes relaciones* **do**
3 seleccionar aleatoriamente un elemento $\mathrm{R} = \alpha \mathrm{D} + \beta \mathrm{D}'$;
4 **if** R *es m-suave* **then**
5 guardar R;
6 **end**
7 **end**
8 resolver un sistema lineal sobre $\mathbb{F}_r$ para obtener λ;
9 **return** λ

Algoritmo 4.2: Algoritmo de cálculo de índices hiperelíptico

A continuación se mencionan los algoritmos basados en el algoritmo de cálculo de índices usados para la resolución del Problema del Logaritmo Discreto en $\mathcal{H}$:

1. Gaudry (2000) propuso un algoritmo que requiere aproximadamente $O(g^3 q^2 \log^2 q + g^2 g! q \log^2 q)$ operaciones en el jacobiano de la curva hiperelíptica.

2. Enge y Gaudry (2002) propusieron un algoritmo génerico que requiere aproximadamente un número subexponencial de operaciones en el jacobiano de la curva hiperelíptica, es decir aproximadamente $L_{q^g}\left[\frac{1}{2}, c + o(1)\right]$ operaciones en el jacobiano de la curva hiperelíptica cuando $g/\log q \to \infty$.

3. Gaudry et al. (2007) propusieron una variación doble de primos grandes ("*Double large prime variation*") de su algoritmo, la cual requiere aproximadamente $\widetilde{O}(q^{2-\frac{2}{g}})$ operaciones en el jacobiano de la curva hiperelíptica cuando el género de la curva, $g \geq 3$, es relativamente pequeño.

4. Sarkar y Singh (2014) propusieron un método basado en el trabajo de Nagao (2010), el cual evita la necesidad de resolver un sistema multivariable

y combina un método de criba propuesto por Joux y Vitse (2012). Dicho trabajo fue descrito para el caso de curvas hiperelípticas primas de género pequeño. En 2015, se aplicó esta idea para el caso de curvas hiperelípticas binarias [Sarkar y Singh, 2015] y mostraron que es posible obtener una sola relación en $(2g+3)!$ intentos.

Observación 4.2.1 *El mejor algoritmo "asintóticamente hablando" entre estas cuatro aproximaciones, es el algoritmo de Enge-Gaudry debido a que para curvas hiperelípticas de género lo suficientemente grande, el método 3) no aplica y los métodos 1) y 4) se vuelven imprácticos* [Jacobson et al., 2001].

El algoritmo de Enge-Gaudry (véase el algoritmo 4.3), es una "extensión" del algoritmo de Gaudry, la cual maneja una base de factores no-lineales que balancea el número de operaciones requeridas para la resolución del sistema lineal asociado y la generación de relaciones. Formalmente, la base de factores está constituida por todos lo polinomios irreducibles en $\mathbb{F}_q[x]$ de grado menor o igual a $m = \lceil \log_q L_{q^g}\left[\frac{1}{2}, \varrho\right] \rceil$ donde $\varrho = \sqrt{\frac{1}{2} + \frac{1}{4\vartheta}} - \sqrt{\frac{1}{4\vartheta}}$ para algún entero positivo ϑ que satisface (i) $g \geq \vartheta \log q$ y (ii) $q \leq L_{q^g}\left[\frac{1}{2}, \frac{1}{\sqrt{\vartheta}}\right]$.

Observación 4.2.2 *El porqué el algoritmo de Gaudry calcula el logaritmo discreto de* D' *en base* D*, es decir,* $\lambda \in \{1, \ldots, r\}$ *tal que* $\mathrm{D}' = \lambda \mathrm{D}$*, es por el siguiente hecho: Supongamos que* γ *es un vector del núcleo de* $\mathrm{M}^\top$ *diferente del vector cero con la propiedad* $\sum \beta_k \gamma_k \neq 0$*. Entonces,*

$$0 = \sum \gamma_i \mathrm{R}_i = \sum \gamma_k \left(\alpha_k \mathrm{D} + \beta_k \mathrm{D}'\right) = \left(\sum \gamma_k \alpha_k\right) \mathrm{D} + \left(\sum \gamma_k \beta_k\right) \mathrm{D}'$$

y por consiguiente $\lambda = -\frac{\sum \alpha_k \gamma_k}{\sum \beta_k \gamma_k}$ [Enge y Gaudry, 2002].

Input: un divisor $\mathrm{D} \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$ de mayor orden primo r, un divisor $\mathrm{D}' \in \langle \mathrm{D} \rangle$ y un parámetro `cnts`
Output: un número entero $\lambda \in \{1, 2, \ldots, r\}$ tal que $\mathrm{D}' = \lambda \mathrm{D}$
`/* construcción de la base de factores G */`
para cada polinomio lineal $u_i \in \mathrm{k}[x]$, tratar de hallar un polinomio $v_i \mathrm{k}[x]$ tal que $\mathrm{div}(u_i, v_i) \in \mathrm{Jac}_{\mathcal{H}}(\mathrm{k})$. Si existe tal divisor, guardar $\mathtt{g}_i = \mathrm{div}(u_i, v_i)$ en `G` (sólo se agrega uno de los dos divisores opuestos);
repeat
  `/* inicialiación de la caminata aleatoria */`
  for $j = 1$ *hasta* `cnts` **do**
    $\alpha^{(j)}, \beta^{(j)} \xleftarrow{\$} \{1, \ldots, r\}$;
    $\mathrm{T}^{(j)} \leftarrow \alpha^{(j)}\mathrm{D} + \beta^{(j)}\mathrm{D}'$;
  end
  $\alpha_0, \beta_0 \xleftarrow{\$} \{1, \ldots, r\}$;
  $\mathrm{R}_0 \leftarrow \alpha_0 \mathrm{D} + \beta_0 \mathrm{D}'$;
  $k \leftarrow 1$;
  `/* ciclo principal */`
  repeat
    `// buscando divisores suaves;`
    repeat
      $j \xleftarrow{\$} \{1, \ldots, \mathtt{cnts}\}$;
      $\mathrm{R}_0 \leftarrow \mathrm{R}_0 + \mathrm{T}^{(j)}$;
      $\alpha_0 \leftarrow \alpha + \alpha^{(j)} \mod r$;
      $\beta_0 \leftarrow \beta + \beta^{(j)} \mod r$;
    until R_0 *sea suave*;
    `// expresando` R_0 `en términos de la base de factores G;`
    escribir $\mathrm{R}_0 = \sum \mathrm{m}_{k,i}\mathtt{g}_i$ y guardar la fila resultante m_k de la matriz M que será usada en el álgebra lineal;
    $\alpha_k \leftarrow \alpha_0$, $\beta_k \leftarrow \beta_0$, $k \leftarrow k + 1$;
  until $k = |\mathtt{G}| + 1$;
  `/* álgebra lineal */`
  hallar un vector γ del núcleo de $\mathrm{M}^{\top}$ diferente del vector cero;
until $\sum \beta_k \gamma_k \neq 0$;
`/* solución */`
return $\lambda \leftarrow -\frac{\sum \alpha_k \gamma_k}{\sum \beta_k \gamma_k}$

Algoritmo 4.3: Algoritmo de Gaudry

Capítulo 5

El ataque GHS

No porque no puedas hallar una solución quiere decir que no exista.

Andrew Wiles

Ciertamente, la seguridad de un sistema criptográfico basado en curvas elípticas, depende de la complejidad del Problema del Logaritmo Discreto en el grupo de puntos racionales de la curva elíptica. Por consiguiente, es de interés determinar cuando dicho problema es intratable "computacionalmente hablando". En la actualidad, existen tres estrategias para la resolución de este problema:

1. Resolver el Problema del Logaritmo Discreto en la curva elíptica mediante el uso del algoritmo ρ de Pollard [Pollard, 1978].

2. Reducir el Problema del Logaritmo Discreto hacia un campo finito mediante la aplicación del ataque MOV [Menezes et al., 1993] y el uso de los algoritmos QPA's [Granger et al., 2014, Barbulescu et al., 2014].

3. Reducir el Problema del Logaritmo Discreto hacia una curva hiperelíptica mediante la aplicación del ataque GHS [Gaudry et al., 2002, Galbraith et al., 2002, Hess, 2004] y el uso del algoritmo de Enge-Gaudry [Enge y Gaudry, 2002].

Por lo tanto, es de interés poder determinar cuando las estrategias 2 y 3 reducen el Problema del Logaritmo Discreto en la curva elíptica hacia un problema mucho más sencillo. Por parte de la segunda estrategia, existe una manera de determinar si esto último sucede; mientras que para la tercera estrategia no. Por consiguiente, es importante determinar cuando ésto sucede.

A grandes rasgos, el ataque GHS relaciona puntos de una curva elíptica con divisores del jacobiano de una curva hiperelíptica definida sobre un subcampo del campo en el cual la curva elíptica está definida.

En la literatura criptográfica, se dice que el ataque GHS es factible (ó que la curva elíptica es vulnerable ante el ataque GHS), si el Problema del Logaritmo Discreto en la curva hiperelíptica es mucho más sencillo de resolver que en la curva elíptica; de manera análoga se habla sobre factibilidad y vulnerabilidad ante la generalización del ataque GHS (gGHS).

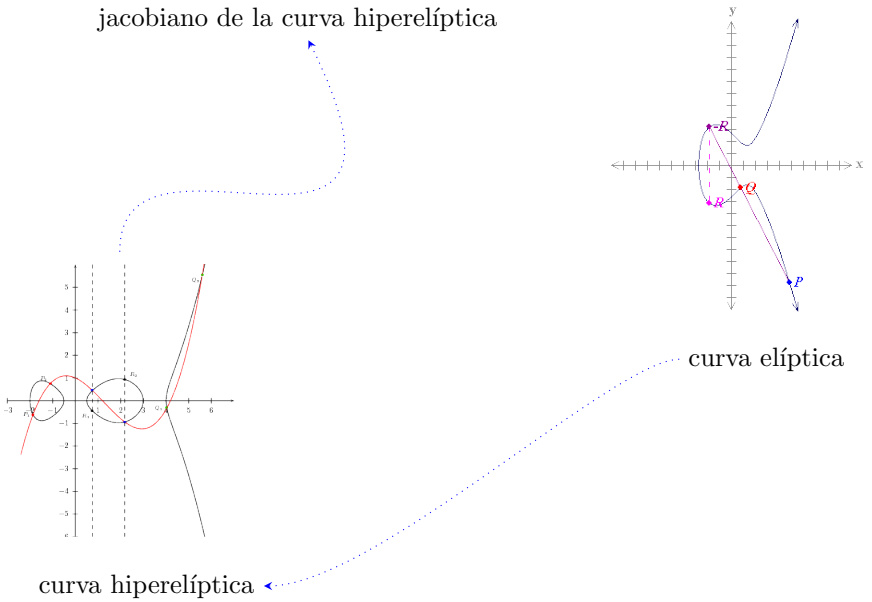

Figura 5.1: Ataque GHS

El contenido de este capítulo ha sido organizado en las siguientes dos secciones: la sección 5.1 describe, de manera general, como está conformado el ataque GHS; mientras que la sección 5.2 explica, de manera breve y concisa, la generalización y extensión de dicho ataque.

5.1. Descripción general del ataque GHS

Las siguientes definiciones y resultados fueron tomados de [Gaudry et al., 2002].

Dados tres números enteros positivos l, n, $q = 2^l$, un campo binario $\mathrm{k} = \mathbb{F}_q$, una extensión $\mathrm{K} = \mathbb{F}_{q^n}$ de grado n del campo k y una curva elíptica binaria $\mathcal{E}/\mathrm{K}$. Entonces, el ataque GHS está constituido mediante los siguientes dos pasos:

1. **Descenso de Weil:** construir la restricción de Weil $\mathcal{W}_{\mathcal{E}/\mathrm{k}}$ de escalares de $\mathcal{E}/\mathrm{K}$, la cual es una variedad abeliana de dimensión n definida sobre el campo k. Esta construcción puede ser realizada de la siguiente manera:

supongamos que $\beta = \{\phi_1, \ldots, \phi_n\}$ determina una base de K (visto K como un k-espacio vectorial). Primero, se deben expresar los parámetros a y b de la curva elíptica $\mathcal{E}$ "al igual que las variables x e y" en términos de la base β, es decir,

$$a = \sum_{i=1}^{n} a_i\phi_i, \quad b = \sum_{i=1}^{n} b_i\phi_i, \quad x = \sum_{i=1}^{n} x_i\phi_i \text{ e } y = \sum_{i=1}^{n} y_i\phi_i \tag{5.1}$$

donde $a_i, b_i, x_i, y_i \in$ k para cada $i = 1, \ldots, n$. Luego, susituir estas representaciones en la ecuación que describe a la curva elíptica $\mathcal{E}$, obteniendo así una variedad abeliana $\mathcal{A}$ de dimensión n definida sobre el campo k, cuya ley de grupo asociada a $\mathcal{A}$ está dada mediante la ley de grupo de la curva elíptica $\mathcal{E}$.

Finalmente, intersecando la variedad abeliana $\mathcal{A}$ con hiperplanos de dimensión $n-1$ (por ejemplo: $x_1 = x_2 = \cdots = x_n = x$) y haciendo uso de la propiedad de independencia de la base β, se debe obtener una curva $\mathcal{C}$ definida sobre el campo k.

2. **Reducción del Problema del Logaritmo Discreto:** reducir el Problema del Logaritmo Discreto en $\mathcal{E}$(K) hacia el Problema del Logaritmo Discreto en $\mathrm{Jac}_{\mathcal{C}}$(k) y resolver este nuevo problema.

No obstante, Gaudry, Hess y Smart mostraron que si $\mathcal{C}(x, y) = 0$ denota la ecuación de la curva obtenida por el descenso de Weil y $0 = \mathcal{C}(x, y) = \mathcal{C}_1 \cdots \mathcal{C}_t$ donde cada $\mathcal{C}_i$ es un polinomio irreducible en K$[x, y]$, es decir, son factores irreducibles de la curva $\mathcal{C}$. Entonces es posible reducir el Problema del Logaritmo Discreto en $\mathcal{E}$(K) hacia $\mathrm{Jac}_{\mathcal{H}}$(k) siendo $\mathcal{H} = \mathcal{C}_j$ una curva hiperelíptica para algún j (para mayor profundidad véanse los lemas 2, 3 y 4 y el teorema 5 de [Gaudry et al., 2002].

Formalmente, supongamos que se satisface al menos una de las siguientes tres propiedades: i) n es un número entero impar, ii) m = n o iii) $\mathrm{Tr}_{\mathrm{K}/\mathbb{F}_2}(a) = 0$ donde

$$m = \dim\left(\mathcal{L}_{\mathbb{F}_2}\left\{\left(1, {b_0}^{1/2}\right), \ldots, \left(1, {b_{n-1}}^{1/2}\right)\right\}\right) \tag{5.2}$$

y b_i denota la i-ésima aplicación del automorfismo de Frobenius de K con respecto a k, es decir, $b_i = \sigma^i(b) = b^{q^i}$ para cada $i = 0, \ldots, n-1$. Entonces, el ataque GHS construye explicitamente un homomorfismo entre los grupos $\mathcal{E}$(K) y $\mathrm{Jac}_{\mathcal{H}}$(k) donde $\mathcal{H}$ es una curva hiperelíptica de género $g = 2^{m-1}$ ó $g = 2^{m-1} - 1$ definida sobre el campo k, la cual es un factor irreducible de la curva $\mathcal{C}$ obtenida por el descenso de Weil.

Observación 5.1.1 *La parte más importante del ataque GHS es el descenso de Weil. Por ello mismo, el siguiente ejemplo muestra una manera de obtener la curva hiperelíptica que resulta del descenso de Weil.*

Ejemplo 5.1.1 ***(Descenso de Weil)*** *Sean l un número entero positivo, $q = 2^l$, $\mathrm{K} = \mathbb{F}_{q^2}$ una extensión de grado 2 del campo $\mathrm{k} = \mathbb{F}_q$ y $\mathcal{E}/\mathrm{K}$ una curva elíptica binaria dada por la ecuación 3.1.*

Objetivo: *obtener el factor irreducible de la curva $\mathcal{C}$ generada por el descenso de Weil.*

Gaudry, Hess y Smart en [Gaudry et al., 2002] afirman que es posible reducir, mediante una proyección racional, cada punto de $\mathcal{C}$ a un punto de una curva $\mathcal{D}$ (y viceversa), donde la curva $\mathcal{D}$ está definida sobre el campo K y está dada por las siguientes ecuaciones:

$$\mathcal{D}\colon \begin{cases} w_0^2 + xw_0 = x^3 + a_0x^2 + b_0 \\ w_1^2 + xw_1 = x^3 + a_1x^2 + b_1 \end{cases}$$

donde $a_i = \sigma^i(a)$, $b_i = \sigma^i(b)$, $w_i = \sigma^i(y)$, $x \in \mathrm{k}$ e $y \in \mathrm{K}$ con $i = 0, 1$. Por lo tanto, trabajaremos con la curva $\mathcal{D}$ debido a que se conoce "explícitamente" su estructura.

Nuevo objetivo: *obtener una factor irreducible de la curva $\mathcal{D}$, el cual estará descrito por la ecuación de una curva hiperelíptica.*

Para obtener dicho factor irreducible, hallaremos la ecuación que describe a dicho factor. Esto se realizará con la ayuda de los siguientes cambios de variables:

Primer cambio de variables. *Sea $s_i = w_i/x + b_i^{1/2}/x$, entonces sustituyendo cada s_i en las ecuaciones que describen a $\mathcal{D}$, se obtiene:*

$$s_0^2 + s_0 = x + a_0 + b_0^{1/2}/x, \tag{5.3}$$

$$s_1^2 + s_1 = x + a_1 + b_1^{1/2}/x. \tag{5.4}$$

Segundo cambio de variables. *Sea $t = s_1 + s_0$, entonces sustituyendo t en la ecuación 5.3, y usando la ecuación 5.4, se obtiene el siguiente cambio de variables para x:*

$$x = \left(\mathrm{Tr}_{\mathrm{K/k}}(a) + t + t^2\right)^{-1} \mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right). \tag{5.5}$$

Observación 5.1.2 *En vista de que $\sigma(w_0) = w_1$, $\sigma(w_1) = w_0$, $\sigma\left(b_0^{1/2}\right) = b_1^{1/2}$, $\sigma\left(b_1^{1/2}\right) = b_0^{1/2}$, se obtiene:*

$$\begin{aligned} \sigma(s_0) &= \sigma(w_0/x + b_0^{1/2}/x) = \sigma(w_0)/x + \sigma\left(b_0^{1/2}\right)/x = w_1/x + b_1^{1/2}/x = s_1, \\ \sigma(s_1) &= \sigma(w_1/x + b_1^{1/2}/x) = \sigma(w_1)/x + \sigma\left(b_1^{1/2}\right)/x = w_0/x + b_0^{1/2}/x = s_0, \\ \sigma(t) &= \sigma(s_1) + \sigma(s_0) = s_0 + s_1 = t, \end{aligned}$$

lo cual implica que $t \in \mathrm{k}$.

Tercer cambio de variables. *Sean* $\mu \in \mathrm{K}$ *con* $\mathrm{Tr}_{\mathrm{K/k}}(\mu) = 1$ *e* $y' = \mathrm{Tr}_{\mathrm{K/k}}(\mu w_0) = \mathrm{Tr}_{\mathrm{K/k}}\left(\mu x s_0 + \mu b_0^{1/2}\right) = \mathrm{Tr}_{\mathrm{K/k}}\left(\mu x s_0 + \mu b^{1/2}\right) = x\mathrm{Tr}_{\mathrm{K/k}}(\mu s_0) + \mathrm{Tr}_{\mathrm{K/k}}\left(\mu b^{1/2}\right)$.

Observación 5.1.3 *Debido a que* $\mathrm{Tr}_{\mathrm{K/k}}(\mu s_0) = \mu^q s_0^q + \mu s_0 = \mu^q \sigma(s_0) + \mu s_0 = \mu^q s_1 + \mu s_0 = \mu^q\left(s_0 + t\right) + \mu s_0 = \mu^q t + (\mu^q + \mu)s_0 = \mu^q t + \mathrm{Tr}_{\mathrm{K/k}}(\mu)s_0 = \mu^q t + s_0$, *se obtiene:*

$$y' = x\left(\mu^q t + s_0\right) + \mathrm{Tr}_{\mathrm{K/k}}\left(\mu b^{1/2}\right) \Rightarrow s_0 = \left(y' + \mathrm{Tr}_{\mathrm{K/k}}(\mu b^{1/2})\right) x^{-1} + \mu^q t.$$

Luego, al multiplicar la ecuación 5.3 por x^{-2} *y al sustituir el valor obtenido de* s_0, *se obtiene:*

$$\mathrm{Y}^2 + \mathrm{Y}x^{-1} = x^{-1} + (a + \mu^q t + \mu^{2q}t^2)x^{-2} + b^{1/2}x^{-3} \tag{5.6}$$

donde

$$\mathrm{Y} = y'x^{-2} + \mathrm{Tr}_{\mathrm{K/k}}(\mu b^{1/2}) \cdot \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}\left(t^4 + t^2 + \left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\right). \tag{5.7}$$

Observación 5.1.4 *Nótese que como* $\mathrm{Tr}_{\mathrm{K/k}}(\gamma) \in \mathrm{k}$ *para todo* $\gamma \in \mathrm{K}$. *Entonces,* $\mathrm{Y} \in \mathrm{k}$.

Obtención de la curva hiperelíptica. *Al sustituir la ecuación 5.5 en la ecuación 5.6, se obtiene la siguiente ecuación en las variables* Y *y* t, *la cual define una curva* $\mathcal{H}$ *con coeficientes en* K *pero con variables en* k *(es decir,* $\mathcal{H}$ *está definida en* k*):*

$$\mathrm{Y}^2 + \mathrm{Y}\left[(\mathrm{Tr}_{\mathrm{K/k}}(a) + t + t^2)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1}\right] = c_6t^6 + c_5t^5 + c_4t^4 + c_3t^3 + c_2t^2 + c_1t + c_0$$

donde

$$\begin{aligned}
c_6 &= \mu^{2q}\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2} + \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3} b^{1/2},\\
c_5 &= c_3 = \mu^q\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2} + \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3} b^{1/2},\\
c_4 &= \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}(a + \mu^{2q}) + b^{1/2}\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3}\left(1 + \mathrm{Tr}_{\mathrm{K/k}}(a)\right)\\
c_2 &= \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1} + \left(a + \mu^{2q}\left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\right)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}\\
&\quad + \left(\mathrm{Tr}_{\mathrm{K/k}}(a) + \left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\right)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3} b^{1/2},\\
c_1 &= \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1} + \mu^q\left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}\\
&\quad + \left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3} b^{1/2},\\
c_0 &= \mathrm{Tr}_{\mathrm{K/k}}(a)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1} + a\left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}\\
&\quad + b^{1/2}\left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^3\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3}.
\end{aligned}$$

Pero, debido a que la curva $\mathcal{H}$ debe ser una curva hiperelíptica, vemos que se tiene que cumplir que $c_6 = 0 \in \mathrm{K}$; implicando que el género de $\mathcal{H}$ sea igual a $2 = 2^{m-1}$, lo cual nos deja en el caso $m = 2 = n$.

Observación 5.1.5 *En vista de que $\mathrm{Tr}_{\mathrm{K/k}}(\mu) = 1 = 1^2 = \left(\mathrm{Tr}_{\mathrm{K/k}}(\mu)\right)^2$ y $0 = c_6 = \mu^{2q}\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2} + \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3} b^{1/2}$, se obtiene:*

$$\begin{aligned} \mu^{2q} &= \mu^q + \mu^2 + \mu, \\ \mu^{2q} &= b^{1/2}\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right), \\ \mu^q &= b^{1/2}\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right) + \mu^2 + \mu. \end{aligned}$$

Entonces, la curva hiperelíptica $\mathcal{H}$ de género 2 está definida por la siguiente ecuación:

$$\mathrm{Y}^2 + \mathrm{Y}\left[(\mathrm{Tr}_{\mathrm{K/k}}(a) + t + t^2)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1}\right] = c_5 t^5 + c_4 t^4 + c_3 t^3 + c_2 t^2 + c_1 t + c_0$$

donde

$$\begin{aligned} c_5 &= c_3 = \left(\mu^2 + \mu\right)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}, \\ c_4 &= b^{1/2}\mathrm{Tr}_{\mathrm{K/k}}(a)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3} + a\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}, \\ c_2 &= \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1} + a\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2} + b^{1/2}\mathrm{Tr}_{\mathrm{K/k}}(a)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-3}, \\ c_1 &= \left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1} + \left(\mu^2 + \mu\right)\left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}, \\ c_0 &= \mathrm{Tr}_{\mathrm{K/k}}(a)\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-1} + a\left[\mathrm{Tr}_{\mathrm{K/k}}(a)\right]^2\left[\mathrm{Tr}_{\mathrm{K/k}}\left(b^{1/2}\right)\right]^{-2}. \end{aligned}$$

5.2. Extensión y generalización del ataque GHS

Las siguientes definiciones y resultados, junto con el único teorema enunciado en esta sección, aparecen en [Galbraith et al., 2002] y [Hess, 2004].

Definición 5.2.1 *Sean l un número entero primo, $q = 2^l$, $\mathrm{K} = \mathbb{F}_{q^n}$ una extensión de grado n del campo $\mathrm{k} = \mathbb{F}_q$, $p = \sum_{i=0}^{d} p_i x^i \in \mathbb{F}_2[x]$ un polinomio de grado d y $\gamma_1, \gamma_2, \gamma_3 \in \mathrm{K}$. Entonces $p^\sigma(x) = \sum_{i=0}^{d} p_i x^{q^i}$ y para cada elemento $\gamma \in \mathrm{K}$ se define a $\mathrm{Ord}_\gamma(x)$ como el único polinomio mónico irreducible $p \in \mathbb{F}_2[x]$ de menor grado tal que $p^\sigma(\gamma) = 0$. De igual manera, se define a $\mathrm{Ord}_{\gamma_1,\gamma_2,\gamma_3}$ como el siguiente polinomio de grado m*

$$\mathrm{Ord}_{\gamma_1,\gamma_2,\gamma_3}(x) = \begin{cases} \mathrm{mcm}\left(\mathrm{Ord}_{\gamma_1}(x), \mathrm{Ord}_{\gamma_2}(x)\right) & \text{si } \mathrm{Tr}_{\mathrm{K/F_2}}(\gamma_3) = 0 \\ \mathrm{mcm}\left(\mathrm{Ord}_{\gamma_1}(x), \mathrm{Ord}_{\gamma_2}(x), x+1\right) & \text{en otro caso.} \end{cases} \tag{5.8}$$

El siguiente teorema es resultado de los teoremas 11 y 12 de [Hess, 2004].

Teorema 5.2.1 *(**ataque gGHS**) Sean l un número entero primo, $q = 2^l$, $\mathrm{K} = \mathbb{F}_{q^n}$ una extensión de grado n del campo $\mathrm{k} = \mathbb{F}_q$, $m = \mathrm{grad}\,(\mathrm{Ord}_{\gamma_1,\gamma_2,\gamma_3}(x))$ y $\gamma_1, \gamma_2, \gamma_3 \in \mathrm{K}$. Entonces, existe una curva hiperelíptica $\mathcal{H}/\mathrm{k}$ de género*

$$g_{\mathcal{H}} = 2^m - 2^{m-\mathrm{grad}\left(\mathrm{Ord}_{\gamma_1}(x)\right)} - 2^{m-\mathrm{grad}\left(\mathrm{Ord}_{\gamma_2}(x)\right)} + 1 \tag{5.9}$$

que puede ser relacionada con una curva elíptica $\mathcal{E}/\mathrm{K} : y^2 + xy = x^3 + ax^2 + b$ con $a = \gamma_3$ y $b = (\gamma_1\gamma_2)^2$ si y sólo si uno "y sólo uno" de los siguientes tres es cierto.

$$\mathrm{Tr}_{\mathrm{K}/\mathbb{F}_2}(\gamma_3) = 0,\ \mathrm{Tr}_{\mathrm{K}/\mathrm{k}}(\gamma_1) \neq 0 \ \text{ó}\ \mathrm{Tr}_{\mathrm{K}/\mathrm{k}}(\gamma_2) \neq 0. \tag{5.10}$$

Observación 5.2.1 *Nótese que si sucede que $\mathrm{grad}\,(\mathrm{Ord}_{\gamma_1}(x)) = 1$, entonces $g_{\mathcal{H}} = 2^{m-1} - 2^{m-\mathrm{grad}\left(\mathrm{Ord}_{\gamma_2}(x)\right)} + 1$. Por consiguiente, si $m - \mathrm{grad}\,(\mathrm{Ord}_{\gamma_2}(x))$ es igual a 0 ó 1, se obtiene que $g_{\mathcal{H}}$ es igual a 2^{m-1} ó $2^{m-1} - 1$, es decir, la curva hiperelíptica $\mathcal{H}$ obtenida por el ataque gGHS coincide con la del ataque GHS.*

Corolario 5.2.1 *El género $g_{\mathcal{H}}$ de la curva hiperelíptica obtenida por el ataque gGHS es igual a $2^{m-1} - 1$ si y sólo si $\mathrm{Tr}_{\mathrm{K}/\mathbb{F}_{q^n}}(\beta) = 0$ donde $n = 2^u \cdot n'$ y n' es un número entero impar.*

Definición 5.2.2 *Sea $\mathcal{E}$ una curva elíptica binaria. Entonces, el ataque GHS extendido se refiere a la aplicación del ataque GHS sobre cada curva elíptica $\mathcal{E}'$ isógena a $\mathcal{E}$. Se dice que $\mathcal{E}$ es vulnerable ante el ataque GHS extendido si existe al menos una curva elíptica isógena a $\mathcal{E}$ vulnerable ante el ataque GHS; de manera análoga se define el ataque gGHS extendido y la vulnerabilidad ante dicha extensión.*

Capítulo 6

Análisis y desarrollo

La verdadera muestra de inteligencia no es el conocimiento sino la imaginación.

Albert Einstein

En las últimas dos décadas, los sistemas criptográficos basados en curvas elípticas se han empleado cada vez más para crear estándares de clave pública y protocolos. La principal razón de este incremento es debido al uso de claves de longitud, en bits, "pequeña", en comparación a otros sistemas criptográficos de clave pública, implicando así una ligera y rápida implementación.

Galbraith, Lin y Scott (2009) introdujeron endomorfismos eficientes "computacionalmente hablando", para una extensa clase de curvas elípticas definidas sobre $\mathbb{F}_{p^2}$ donde p es un número primo. Desde entonces, muchos autores han combinado los endomorfismos de Galbraith-Lin-Scott con el método de descomposición de Gallant-Lambert-Vanstone [Gallant et al., 2001] para obtener aceleraciones en las implementaciones de la multiplicación escalar en curvas elípticas binarias [Hankerson et al., 2011, Oliveira et al., 2014] y primas [Hu et al., 2012, Bos et al., 2013, Longa et al., 2014, Faz-Hernández et al., 2014].

Debido al interés que se tiene en determinar la seguridad que un sistema criptográfico basado en una curva elíptica GLS brinda ante el ataque gGHS extendido; este capítulo se compone de cuatro secciones requeridas para la comprensión y justificación del mecanismo de vulnerabilidad "ante el ataque gGHS extendido" propuesto en esta tesis.

La sección 6.1 describe de manera general la definición y construcción de una curva elíptica GLS, la sección 6.2 explica el mecanismo de vulnerabilidad propuesto, justificando y ejemplificando cada resultado. La sección 6.3 detalla dicho mecanismo de manera algorítmica junto con la implementación realizada y, para la finalizar, la sección 6.4 comprueba los resultados teóricos obtenidos mediante ejemplos y comparaciones.

6.1. Analizando las curvas GLS

Las siguientes definiciones y aproximación fueron tomados de [Hankerson et al., 2011].

Dados un número entero primo ℓ, una extensión $\mathbb{F}_{2^{2\ell}}$ de grado dos del campo $\mathbb{F}_{2^\ell}$ y una curva elíptica binaria $\mathcal{E}'/\mathbb{F}_{2^\ell}$ dada por la ecuación 6.1.

$$\mathcal{E}'/\mathbb{F}_{2^\ell}: y^2 + xy = x^3 + a'x^2 + b \qquad a' \in \mathbb{F}_{2^\ell}, b \in \mathbb{F}_{2^\ell}^{\times} \ . \tag{6.1}$$

Nótese que $\#\mathcal{E}'(\mathbb{F}_{2^\ell}) = 2^\ell + 1 - t$ y $\#\mathcal{E}'(\mathbb{F}_{2^{2\ell}}) = (2^\ell + 1)^2 - t^2$ donde t denota la traza de Frobenius de la curva $\mathcal{E}'/\mathbb{F}_{2^\ell}$. Sea $a \in \mathbb{F}_{2^{2\ell}}$ con $\mathrm{Tr}_{\mathbb{F}_{2^{2\ell}}/\mathbb{F}_{2^\ell}}(a) = 1$, entonces es posible construir una curva elíptica GLS, $\mathcal{E}/\mathbb{F}_{2^{2\ell}}$, que está dada por la ecuación 6.2

$$\mathcal{E}/\mathbb{F}_{2^{2\ell}}: y^2 + xy = x^3 + ax^2 + b, \tag{6.2}$$

la cual es isomorfa a la curva elíptica $\mathcal{E}'$ sobre $\mathbb{F}_{2^{4\ell}}$ bajo el isomorfismo involutivo $\tau: \mathcal{E}' \to \mathcal{E}$ definido por $(x, y) \mapsto (x, y + sx)$ donde $s \in \mathbb{F}_{2^{4\ell}} \setminus \mathbb{F}_{2^{2\ell}}$ satisface $s^2 + s = a + a'$. Como consecuencia, el endomorfismo de Galbraith-Lin-Scott, ψ, puede ser construido aplicando el automorfismo de Frobenius σ de la siguiente manera: $\psi = \tau\sigma\tau^{-1}$.

En vista de que la seguridad de un sistema criptográfico basado en una curva elíptica GLS depende de la complejidad de resolver el Problema del Logaritmo Discreto en $\mathcal{E}(\mathbb{F}_{2^{2\ell}})$, la aproximación usual es aplicar el algoritmo ρ de Pollard para curvas elípticas, el cual requiere aproximadamente $O\left(\sqrt{\frac{\pi 2^{2\ell}}{2}}\right)$ operaciones. Por consiguiente, al aplicar el ataque gGHS, es necesario determinar si la complejidad en resolver el Problema del Logaritmo Discreto en $\mathrm{J}_{\mathcal{H}}(\mathbb{F}_2)$, $\mathrm{J}_{\mathcal{H}}(\mathbb{F}_{2^2})$ ó $\mathrm{J}_{\mathcal{H}}(\mathbb{F}_{2^\ell})$ es menor a $O\left(\sqrt{\frac{\pi 2^{2\ell}}{2}}\right)$. Dicha comparación se realiza con el algoritmo de Enge-Gaudry, el cual requiere aproximadamente $L_{q^g}\left[\frac{1}{2}, c + o(1)\right]$ operaciones, donde g denota el género de la curva hiperelíptica definida sobre el campo finito $\mathbb{F}_q$ para $q \in \{2, 2^2, 2^\ell\}$. Cabe mencionar que para el caso cuando $q = 2^\ell$, el ataque gGHS genera una curva hiperelíptica de género 2 ó 3 y para estos casos, el ataque GHS no es factible [Hankerson et al., 2011].

En la actualidad existen dos métodos usados para determinar si dada una curva elíptica GLS, ésta es "o no es" vulnerable ante el ataque gGHS extendido:

- Suponiendo que el número de curvas elípticas isógenas a la curva eliptica GLS, $\mathcal{E}$, es menor que el número de clases de isogenias de curvas elípticas vulnerables ante el ataque gGHS, entonces los siguientes pasos describen un método para determinar si dada una curva elíptica GLS, $\mathcal{E}$, es "ó no es" vulnerable ante el ataque gGHS extendido [Galbraith et al., 2002].

1. **Configuración del entorno.** Sea $\mathcal{E}_{a,b}$ $\mathbb{F}_{2^{2\ell}}$ una curva elíptica GLS con parámetros a y b, que está determinada por la ecuación 6.2. En el contexto del ataque gGHS, la extensión de campo $\mathbb{F}_{2^{2\ell}}$ puede ser vista como una extensión de grado ℓ del campo $\mathbb{F}_{2^2}$ o como una extensión de grado 2ℓ del campo $\mathbb{F}_2$.
2. **Verificando el parámetro b.** Para ambos casos, cuando $q = 2$ y $q = 2^2$, hallar "si es posible" dos elementos γ_1 y γ_2 elementos de $\mathbb{F}_{2^{2\ell}}$ que minimicen el género, g, resultante del ataque gGHS con la propiedad $b = (\gamma_1\gamma_2)^2$.
3. **Determinando vulnerabilidad.** i) Si el par (γ_1, γ_2) existe para $q = 2$ o $q = 2^2$, entonces ii) verificar si la complejidad del algoritmo de Enge-Gaudry, aplicado a la curva resultante del ataque gGHS, es menor que la complejidad del algoritmo ρ de Pollard, aplicado a la curva $\mathcal{E}_{a,b}$. Si este es el caso, entonces la curva $\mathcal{E}_{a,b}$ es vulnerable ante el ataque gGHS. Si i) o ii) es falso, ir al paso 4.
4. **El ataque gGHS extendido.** Para cada curva isógena a $\mathcal{E}_{a,b}$, realizar los pasos 2 y 3. Si no existen curvas elípticas isógenas a $\mathcal{E}_{a,b}$ vulnerables ante el ataque gGHS, entonces la curva elíptica $\mathcal{E}_{a,b}$ no es vulnerable ante el ataque gGHS extendido.

- Si el número de curvas elípticas isógenas a $\mathcal{E}$ es mayor que el número de clases de isogenias de curvas elípticas vulnerables ante el ataque gGHS, un método más eficiente es listar todos los parámetros $\tilde{b}$ tales que $\mathcal{E}_{a,\tilde{b}}$ es vulnerable ante el ataque gGHS y guardar $\#\mathcal{E}_{a,\tilde{b}}(\mathbb{F}_{2^{2\ell}})$ en el conjunto L. La verificación consiste en determinar si $\#\mathcal{E}(\mathbb{F}_{2^{2\ell}}) \in L$ [Hankerson et al., 2011].

Como consecuencia de lo mencionado antes, la complejidad en resolver el Problema del Logaritmo Discreto en $\mathrm{J}_{\mathcal{H}}(\mathbb{F}_2)$ ó $\mathrm{J}_{\mathcal{H}}(\mathbb{F}_{2^2})$ es determinado por el género de la curva hiperelíptica obtenida; dicho género depende directamente de los polinomios Ord_{γ_1} y Ord_{γ_2}, los cuales son factores del polinomio $(x^\ell + 1)^2$ ó $x^\ell + 1$, respectivamente. Por esta razón, en característica dos, existen muchas extensiones de campos donde el género de la curva obtenida es enorme y consecuentemente el ataque gGHS se vuelve impráctico para cualquier curva GLS definida sobre dichos campos.

Para ilustrar estos casos, se presenta en el cuadro 6.1 el número de bits necesarios para resolver el Problema del Logaritmo Discreto usando el algoritmo de Enge-Gaudry aplicado a la curva hiperelíptica obtenida, $\mathcal{H}$ con género $g < 10^6$, y el algoritmo ρ de Pollard aplicado a la curva elíptica GLS, $\mathcal{E}$ definida sobre el campo $\mathbb{F}_{2^{2\ell}}$ con $\ell \in [5, 257]$.

6.2. Mecanismo de vulnerabilidad

En esta sección se propone un mecanismo que realiza el paso 3 antes mencionado. El mecanismo es factible cuando el número de curvas elípticas isógenas

				Logaritmo base dos de la complejidad de cada algoritmo	
Campo base de $\mathcal{E}$	Campo base de $\mathcal{H}$	Género de $\mathcal{H}$	$\log_2(\#\mathcal{E}(\mathbb{F}_{2^{2\ell}}))$	Algoritmo de ρ de Pollard sobre $\mathcal{E}$	Algoritmo de Enge-Gaudry sobre $\mathcal{H}$
$\mathbb{F}_{2^{2\cdot 5}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	32 15	9.08	**4.87**	16.91 16.20
$\mathbb{F}_{2^{2\cdot 7}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	16 7	13.02	**6.83**	10.53 9.58
$\mathbb{F}_{2^{2\cdot 11}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	2048 1023	21.00	**10.82**	207.10 206.98
$\mathbb{F}_{2^{2\cdot 13}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	8192 4095	25.00	**12.82**	452.03 451.96
$\mathbb{F}_{2^{2\cdot 17}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	512 255	33.00	**16.82**	93.13 92.92
$\mathbb{F}_{2^{2\cdot 19}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	524288 262143	37.00	**18.82**	4400.90 4400.90
$\mathbb{F}_{2^{2\cdot 23}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	4096 2047	45.00	**22.82**	306.55 306.46
$\mathbb{F}_{2^{2\cdot 31}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	64 31	61.00	30.82	26.46 **25.93**
$\mathbb{F}_{2^{2\cdot 43}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	32768 16383	85.00	**42.82**	973.85 973.82
$\mathbb{F}_{2^{2\cdot 73}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	1024 511	145.00	**72.82**	139.27 139.12
$\mathbb{F}_{2^{2\cdot 89}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	4096 2047	177.00	**88.82**	306.55 306.46
$\mathbb{F}_{2^{2\cdot 127}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	256 127	253.00	126.83	61.84 **61.56**
$\mathbb{F}_{2^{2\cdot 151}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	65536 32767	301.00	**150.83**	1424.00 1424.00
$\mathbb{F}_{2^{2\cdot 257}}$	$\mathbb{F}_2$ $\mathbb{F}_{2^2}$	131072 65535	513.00	**256.82**	2077.90 2077.90

Cuadro 6.1: Curvas GLS vulnerables ante el ataque gGHS

a la curva eliptica GLS es menor que el número de clases de isogenias de curvas elípticas vulnerables ante el ataque gGHS.

Entrando en formalidad matemática, sean $n \in \{\ell, 2\ell\}$, $q = 2^{\frac{2\ell}{n}}$, $i_n = \frac{n}{\ell}$ (i_n es 2 ó 1) y $x^n + 1 = (x+1)^{i_n} f_1^{i_n} f_2^{i_n} \cdots f_s^{i_n}$ donde cada $f_i \in \mathbb{F}_2[x]$ es un polinomio irreducible de grado d para cada $i \in \{1, \ldots, s\}$. De igual manera, para cada $\gamma \in \mathbb{F}_{2^{2\ell}}$ sea Ord_γ definido como en la sección 5.2, es decir, Ord_γ define el polinomio, $f(x) = \sum_{i=0}^{d_f} c_i x^i \in \mathbb{F}_2[x]$, de menor grado, d_f, que satisface $f^\sigma(\gamma) = 0$ donde $f^\sigma(x) = \sum_{i=0}^{d_f} c_i x^{q^i}$. Entonces, podemos enunciar los siguentes corolarios y teoremas:

Teorema 6.2.1 $\forall f, g \in \mathbb{F}_2[x]$, $(f \cdot g)^\sigma(x) = (f^\sigma \circ g^\sigma)(x) = (g^\sigma \circ f^\sigma)(x)$.

Demostración. En vista de que

$$(f \cdot g)(x) = (\sum_{i=0}^{d} f_i x^i)(\sum_{j=0}^{d} g_i x^j) = \sum_{i=0}^{d}\sum_{j=0}^{d}(f_i g_i) x^{i+j},$$

entonces

$$\begin{aligned}
(f\cdot g)^{\sigma}(x) &= \sum_{i=0}^{d}\sum_{j=0}^{d} f_i g_j x^{q^{i+j}} \\
&= \sum_{i=0}^{d}\sum_{j=0}^{d} f_i \left(g_i x^{q^j}\right)^{q^i} \\
&= \sum_{i=0}^{d} f_i \sum_{j=0}^{d} \left(g_j x^{q^j}\right)^{q^i} \\
&= \sum_{i=0}^{d} f_i \left(\sum_{j=0}^{d} g_j x^{q^j}\right)^{q^i} \\
&= \left(f^{\sigma}\circ g^{\sigma}\right)(x)
\end{aligned}$$

De manera análoga se obtiene que $(g\cdot f)^{\sigma}(x) = \left(g^{\sigma}\circ f^{\sigma}\right)(x)$. Por lo tanto, debido a que $(f\cdot g)(x) = (g\cdot f)(x)$, se concluye que $(f\cdot g)^{\sigma}(x) = (g\cdot f)^{\sigma}(x) = (g^{\sigma}\circ f^{\sigma})(x)$, es decir, el teorema se satisface.

Teorema 6.2.2 *$\forall f \in \mathbb{F}_2[x]$, $f^{\sigma}(x) \mid (f\cdot g)^{\sigma}(x)$ en $\mathcal{F}[x]$ donde $\mathcal{F}$ denota el campo de descomposición de f^{σ}, es decir, $\mathcal{F} = \mathbb{F}_{2^{2\ell}}(\alpha_1,\ldots,\alpha_{q^d})$ y cada $\alpha_i \in \overline{\mathbb{F}}_2$ es una raíz del polinomio f^{σ}.*

Demostración. Dado que el polinomio $p^{\sigma}(x)$ tiene como raíz $x=0$ para todo polinomio $p\in\mathbb{F}_2[x]$, sea $\alpha\in\overline{\mathbb{F}}_2$ una raíz de $f^{\sigma}(x)$. Entonces, $g^{\sigma}(f^{\sigma}(\alpha)) = g^{\sigma}(0) = 0$, es decir, α también es una raíz del polinomio $(f\cdot g)^{\sigma}(x)$. Por consiguiente, $f^{\sigma}(x) = \prod(x-\alpha_i)$ divide a $(f\cdot g)^{\sigma}(x)$ en $\mathcal{F}[x]$ donde $\mathcal{F} = \mathbb{F}_{2^{2\ell}}(\alpha_1,\ldots,\alpha_{q^d})$ y cada $\alpha_i\in\overline{\mathbb{F}}_2$ es una raíz del polinomio f^{σ}.

Corolario 6.2.1 *$\forall\gamma\in\mathbb{F}_{2^{2\ell}}$, $\mathrm{Ord}_{\gamma^{\sigma}}(x)$ se descompone en factores lineales en $\mathbb{F}_{2^{2\ell}}[x]$.*

Demostración. Como resultado del teorema 6.2.2, dados dos polinomios $p(x)$, $r(x)\in\mathbb{F}_2[x]$ con $r(x)|p(x)$ satisfacen $r^{\sigma}(x)|p^{\sigma}(x)$ en $\mathcal{F}[x]$ donde $\mathcal{F} = \mathbb{F}_{2^{2\ell}}(\alpha_1,\ldots,\alpha_{q^{\mathrm{grad}(r)}})$ y cada $\alpha_i\in\overline{\mathbb{F}}_2$ es una raíz de $r^{\sigma}(x)$. Por lo tanto, $\mathrm{Ord}_{\gamma}(x)|x^n+1$ implica $\mathrm{Ord}_{\gamma}^{\sigma}(x)|(x^{q^n}+x)$ en $\mathcal{F}[x]$ siendo $\mathcal{F} = \mathbb{F}_{2^{2\ell}}(\alpha_1,\ldots,\alpha_{q^{\mathrm{grad}(\mathrm{Ord}_{\gamma})}})$ y debido a que $\lambda^{q^n} = \lambda^{2^{2\ell}} = \lambda$ para todo $\lambda\in\mathbb{F}_{2^{2\ell}}$ obtenemos que $x^{q^n}+x$ tiene todas su raíces en $\mathbb{F}_{2^{2\ell}}$. Como consecuencia, $\mathbb{F}_{2^{2\ell}} = \mathbb{F}_{2^{2\ell}}(\alpha_1,\ldots,\alpha_{q^{\mathrm{grad}(\mathrm{Ord}_{\gamma})}})$ y $\mathrm{Ord}_{\gamma}^{\sigma}(x)$ se descompone en factores lineales en $\mathbb{F}_{2^{2\ell}}[x]$.

Corolario 6.2.2 *Sean $b\in\mathbb{F}_{2^{2\ell}}$ y $\mathcal{S} = \left\{(s_1,s_2)\mid \exists\gamma_1,\gamma_2\in\mathbb{F}_{2^{2\ell}}\colon s_i = \mathrm{Ord}_{\gamma_i}, \sqrt{\frac{\pi 2^{2\ell}}{2}} > L_{q^g}\left[\frac{1}{2}, c+o(1)\right]\right.$ donde $g = 2^m - 2^{m-\mathrm{grad}(s_1)} - 2^{m-\mathrm{grad}(s_2)} + 1 > n$ y m es el grado del polinomio $p_{s_1,s_2}(x) = \mathrm{mcm}(s_1(x), s_2(x), x+1)$ y $\mathrm{grad}(s_1)\geq \left.\mathrm{grad}(s_2)\right\}$. Definamos para cada $(s_1,s_2)\in\mathcal{S}$ los siguientes polinomios $b_i(x) = s_i^{\sigma}(b^{1/2}x)$ y $\overline{s}_i(x) = x^{q^{\mathrm{grad}(s_i)}} s_i^{\sigma}(\frac{1}{x})$. Entonces, $b_i(x)$ y $\overline{s}_i(x)$ se descomponen en factores lineales en $\mathbb{F}_{2^{2\ell}}[x]$.*

Demostración. Para cada par $(s_1, s_2) \in \mathcal{S}$ tenemos $s_1(x), s_2(x) | x^n + 1$, entonces aplicando el corolario 6.2.1 obtenemos que $s_1(x)$ y $s_2(x)$ se descomponen en factores lineales en $\mathbb{F}_{2^{2\ell}}[x]$. Por lo tanto, por construcción $b_i(x)$ y $\overline{s}_i(x)$ satisfacen el enunciado.

Teorema 6.2.3 *Sean* $b \in \mathbb{F}_{2^{2\ell}}$ *y* $\mathcal{B} = \{\tilde{b} = (\gamma_1\gamma_2)^2 \in \mathbb{F}_{2^{2\ell}} \colon (\mathrm{Ord}_{\gamma_1}, \mathrm{Ord}_{\gamma_2}) \in \mathcal{S}\}$*, entonces*

$$\exists (s_1, s_2) \in \mathcal{S} \colon \mathrm{mcd}\,(b_1(x), \overline{s}_2(x)) \neq 1 \Leftrightarrow b \in \mathcal{B}. \tag{6.3}$$

Demostración. Supongamos que $\exists (s_1, s_2) \in \mathcal{S}$ tal que $\mathrm{mcd}\,(b_1(x), \overline{s}_2(x)) \neq 1$, entonces aplicando el corolario 6.2.2 obtenemos que esta afirmación es equivalente al siguiente enunciado: $\exists \gamma \in \mathbb{F}^{\times}_{2^{2\ell}}$ tal que $s_1^{\sigma}(b^{1/2}\gamma) = 0$ y $\lambda^{q^{\mathrm{grad}(s_2)}} s_2^{\sigma}(\frac{1}{\lambda}) = 0$, luego

$$\begin{array}{lll} \exists (s_1, s_2) \in \mathcal{S} \colon & & \exists \gamma \in \mathbb{F}^{\times}_{2^{2\ell}} \colon s_i^{\sigma}(\gamma_i) = 0 \\ \mathrm{mcd}\,(b_1(x), \overline{s}_2(x)) \neq 1 & \Leftrightarrow & \text{donde } \gamma_1 = b^{1/2}\gamma \wedge \gamma_2 = \frac{1}{\gamma} \\ & \Leftrightarrow & \exists \gamma_1, \gamma_2 \in \mathbb{F}^{\times}_{2^{2\ell}} \colon \\ & & b = (\gamma_1\gamma_2)^2 \wedge s_i^{\sigma}(\gamma_i) = 0 \\ & \Leftrightarrow & b \in \mathcal{B}. \end{array}$$

Como consecuencia del teorema 6.2.3, se obtiene que para determinar la pertenencia de un elemento en el conjunto $\mathcal{B}$, no es necesario verificar cada posible combinación $(\gamma_1\gamma_2)^2$ donde $\gamma_1, \gamma_2 \in \mathbb{F}_{2^{2\ell}}$, $s_1^{\sigma}(\gamma_1) = 0$ y $s_2^{\sigma}(\gamma_2) = 0$ para cada par $(s_1, s_2) \in \mathcal{S}$. De igual manera, por la estructura del conjunto $\mathcal{B}$ se obtiene un manera eficiente de determinar si una curva elíptica GLS es o no vulnerable ante el ataque gGHS.

Corolario 6.2.3 *Sea* $\mathcal{E}_{a,b}$ *una curva elíptica GLS con parámetros a y b, entonces* $\mathcal{E}_{a,b}$ *es vulnerable ante el ataque gGHS si y sólo si* $\exists (s_1, s_2) \in \mathcal{S}$ *tal que* $\mathrm{mcd}\,(b_1(x), \overline{s}_2(x)) \neq 1$*.*

Demostración. Debido a que $\mathrm{Tr}_{\mathbb{F}_{2^{2\ell}}/\mathbb{F}_2}(a) = 1 \neq 0$, entonces el género de la curva hiperelíptica resultante del ataque gGHS coincide con el número natural, g, usado en la definición del conjunto $\mathcal{S}$ definido en en el corolario 6.2.2. Además, las complejidades de los algoritmos de Enge-Gaudry y ρ de Pollard satisfacen las propiedades necesarias y suficientes para afirmar que $\mathcal{E}_{a,b}$ es vulnerable ante el ataque gGHS, es decir, basta determinar si $b \in \mathcal{B}$. Por lo tanto, por el teorema 6.2.3 se concluye la veracidad de este corolario.

6.3. Algoritmo e implementación

El algoritmo 6.1 describe de manera general el resultado principal obtenido en la sección anterior, es decir, el corolario 6.2.3; mientras que el algoritmo 6.2 aplica el algoritmo 1 a cada curva elíptica isógena, para así determinar si acaso es vulnerable ante el ataque gGHS extendido.

Input: el elemento $b \in \mathbb{F}_{2^{2\ell}}^{\times}$, las listas de polinomios $b_1, \overline{s}_2$ obtenidos del conjunto $\mathcal{S}$ para $q \in \{2, 2^2\}$
Output: *verdadero* si el parámetro b construye una curva elíptica GLS vulnerable ante el ataque gGHS y *falso* en caso contrario

```
1  aux ← 1, j ← 0;
2  while aux = 1 and j < #S do
3  |  j ← j + 1;
4  |  aux ← gcd(b_1[j], s̄_2[j]);
5  end
6  if aux ≠ 1 then
7  |  return verdadero
8  else
9  |  return falso
10 end
```

Algoritmo 6.1: Mecanismo de verificación del parámetro b.

Input: una curva elíptica GLS $\mathcal{E}_{a,b}$ con paŕámetros a yb
Output: *verdadero* si $\mathcal{E}_{a,b}$ es vulnerable ante el ataque gGHS extendido y *falso* en caso contrario

```
1 for Para cada curva elíptica E_i isógena a E_{a,b} do
2 |  determinar el parámetro b_i ∈ F_{2^{2ℓ}} de la curva E_i;
3 |  construir las listas de polinomios b_{1,i}, s̄_{2,i} obtenidos del conjunto S_i
  |   para q = 2 y q = 2^2;
4 |  flag ← Algoritmo_1(b_i, b_{1,i}, s̄_{2,i});
5 |  if flag then
6 |  |  return verdadero
7 |  end
8 end
9 return falso
```

Algoritmo 6.2: Mecanismo de vulnerabilidad ante el ataque gGHS extendido

Análisis de complejidad. Sea $s_{\text{máx}}$ el polinomio de mayor grado, $d_{\text{máx}}$, de todos los pares de polinomios (s_1, s_2) en $\mathcal{S}$ para $q = 2$ y $q = 2^2$, supongamos que $q_{\text{máx}}$ es el valor que corresponde a $d_{\text{máx}}$. Entonces, en vista de que la complejidad en computar el máximo cómun divisor sobre todos los pares de polinomios $(b_1, \overline{s}_2)$ es $O((q_{\text{máx}}{}^{d_{\text{máx}}})^2)$, obtenemos que la complejidad del algoritmo 6.1 es $O(\#\mathcal{S} \cdot (q_{\text{máx}}{}^{d_{\text{máx}}})^2)$. Por otro lado, si suponemos que el número de curvas elípticas isógenas a la curva original es I, entonces la complejidad del algoritmo 6.2 es $O(\mathrm{I} \cdot \#\mathcal{S} \cdot (q_{\text{máx}}{}^{d_{\text{máx}}})^2)$.

Implementación. Se realizaron las implementaciones del algoritmo 6.1 y del ataque GHS mediante la ayuda del software **Magma Computational Alge-**

bra System (**Magma**) desarrollado por el Grupo de Álgebra Computacional de la Escuela de Matemáticas y Estadística de la Universidad de Sydney; se optó por usar **Magma** debido a que brinda una gran variedad de herramientas matemáticas necesarias para dichas implementaciones. A continuación se explica de manera general las implementaciones realizadas[1]:

- Implementación del ataque GHS
  - Funciones que **Magma** trae por omisión:
    - **WeilDescent()**. Usada para la obtención de la curva hiperelíptica y del homomorfismo generado por el ataque GHS.
    - **Jacobian()**. Necesaria para la obtención del Jacobiano de la curva hiperelíptica obtenida por **WeilDescent()**.
  - Se realizó la implementación de una adaptación del algoritmo Enge-Gaudry sobre el jacobiano de una curva hiperelíptica descrito en la sección 4.2. Dada una cota de suavidad $\mathtt{d_G}$, dicha adaptación consiste en los siguentes tres pasos:
    1. La construcción de la base de factores fue hecha dinámicamente; iniciando la base de factores $\mathtt{G} = \emptyset$ y limitando que $\mathtt{G}$ solo puede contener polinomios de grado menor o igual a $\mathtt{d_G}$.
    2. En seguida, para cada relación válida en el algoritmo de Enge-Gaudry, es decir, cuando el polinomio $\mathtt{u}$ de un divisor $\mathrm{div}(\mathtt{u}, \mathtt{v})$ que es $\mathtt{d_G}$-suave, se añaden los factores irreducibles de $\mathtt{u}$ que no pertenecen a $\mathtt{G}$.
    3. Finalmente, cuando el número de relaciones es igual a $|\mathtt{G}|$, se finaliza la etapa de recolección de relaciones.
- La implentación del algoritmo 6.1 se realizó únicamente usando la biblioteca de campos finitos y de anillos polinimos que **Magma** trae por omisión.

Cabe mencionar que estás implementaciones hechas en **Magma** pueden ser trasladadas a lenguaje **C**. Teniendo en cuenta que se deberían realizar las siguientes implementaciones: i) aritmética en campos binarios, en curvas binaras y en el jacobiano de curvas hiperelípticas binarias, ii) el descenso de Weil: para la obtención de la curva hiperelíptica binaria, iii) el algoritmo de Enge-Gaudry, iv) el homomorfismo resultante del ataque GHS y v) el algoritmo 6.1.

6.4. Comparaciones y ejemplos

Con el propósito de comprender las implicaciones prácticas del ataque gGHS sobre una curva elíptica GLS, se ha implementado un ataque completo sobre una curva definida sobre el campo $\mathbb{F}_{2^{31\cdot 2}}$, se escogió dicho campo por lo observado

[1] El código de las implementaciones se adjunta en los apéndices A y B

en el cuadro 6.1. De igual manera, se muestra que la curva elíptica GLS usada en [Oliveira et al., 2014] no es vulnerable ante el ataque gGHS mediante una ejecución del algoritmo 6.1.

6.4.1. Construyendo una curva vulnerable ante el ataque gGHS

Sean $n \in \{31, 62\}$, $q = 2^{\frac{2 \cdot 31}{n}}$ y $\mathbb{F}_{2^{2 \cdot 31}}$ una extensión de grado n del campo $\mathbb{F}_q$. Entonces, es posible representar el campo $\mathbb{F}_{2^{62}}$ de las siguientes dos maneras:

- $n = 62, q = 2$, $\mathbb{F}_{2^{62}} \cong \mathbb{F}_2[v]/f(v)$, con $f(v) = v^{62} + v^{29} + 1$.
- $n = 31, q = 2^2$, $\mathbb{F}_{2^2} \cong \mathbb{F}_2[z]/g(z)$, con $g(z) = z^2 + z + 1$.
  $\mathbb{F}_{2^{62}} \cong \mathbb{F}_{2^2}[u]/h(u)$, con $h(u) = u^{31} + u^3 + 1$.

Sea $\mathcal{E}_{a,b}$ una curva elíptica GLS, con parámetros a y b, dada por la ecuación 6.2 con la propiedad $\#\mathcal{E}_{a,b}(\mathbb{F}_{2^{62}}) = c \cdot r$, donde c un número natural pequeño y r es un número primo.

Objetivo: hallar un par de parámetros que hagan a $\mathcal{E}_{a,b}$ vulnerable ante el ataque gGHS ó demostrar que estos no existen.

Dado que el parámetro a puede ser seleccionado aleatoriamente sujeto a la condición $\mathrm{Tr}_{\mathbb{F}_{2^{62}}/\mathbb{F}_2}(a) = 1$, seleccionemos $a = z^2$. Sea $x^{31} + 1 = (x+1)f_1 \cdots f_6$ con $\mathrm{grad}(f_i) = 5$, entonces los cuadros 6.2 y 6.3 listan los pares de polinomios que generan parámetros $b = (\gamma_1\gamma_2)^2$ de nuestro interés.

Cuadro 6.2: Pares de polinomios $(\mathrm{Ord}_{\gamma_1}, \mathrm{Ord}_{\gamma_2}) \in \mathcal{S}$ para el caso $n = 31$, $q = 2^2$.

Ord_{γ_1}	Ord_{γ_2}	$\mathrm{grad}(\mathrm{Ord}_{\gamma_1})$	$\mathrm{grad}(\mathrm{Ord}_{\gamma_2})$	m	género	Seguridad en bits del algoritmo de Enge-Gaudry
$(x+1)f_i$	$x+1$	6	1	6	32	26.46
f_i	$x+1$	5	1	6	31	25.93

Cuadro 6.3: Pares de polinomios $(\mathrm{Ord}_{\gamma_1}, \mathrm{Ord}_{\gamma_2}) \in \mathcal{S}$ para el caso $n = 62$, $q = 2$.

Ord_{γ_1}	Ord_{γ_2}	$\mathrm{grad}(\mathrm{Ord}_{\gamma_1})$	$\mathrm{grad}(\mathrm{Ord}_{\gamma_2})$	m	género	Seguridad en bits del algoritmo de Enge-Gaudry
$(x+1)^2 f_i$	$x+1$	7	1	7	64	26.46

Bajo la configuración ($n = 31$, $q = 2^2$), seleccionamos el parámetro $b = u^{24} + u^{17} + u^{16} + u^{12} + u^5 + u^4 + u^3 + u + 1$, el cual permite construir una curva elíptica GLS de orden $\#\mathcal{E}_{a,b}(\mathbb{F}_{2^{62}}) = 4611686014201959530$, con un subgrupo orden aproximadamente igual a $2^{51.38}$.

Observación 6.4.1 *En teoría, resolver el Problema del Logaritmo Discreto sobre este subgrupo mediante el algoritmo ρ de Pollard tomaría alrededor de 2^{26} operaciones en el campo $\mathbb{F}_{2^{2\ell}}$, el cual coincide con el del algoritmo de Enge-Gaudry.*

Por consiguiente, el ataque gGHS genera una curva hiperlíptica, $\mathcal{H}$, de género 32 la cual está dada por la siguiente ecuación:

$$\begin{aligned}\mathcal{H}/\mathbb{F}_{2^2} \quad : \quad & y^2 + (z^2x^{32} + x^{16} + z^2x^8 + z^2x^2 + x)y = x^{65} + x^{64} + z^2x^{33} \\ & + zx^{32} + x^{17} + z^2x^{16} + x^8 + x^5 + x^4 + z^2x^3 + zx^2 + zx.\end{aligned}$$

6.4.1.1. Trasladando el Problema del Logaritmo Discreto

Finalmente, mediante la ayuda de **Magma** generamos aleatoriamente el punto $P \in \mathcal{E}_{a,b}(\mathbb{F}_{2^{62}})$, el cual es un generador de orden r y al punto $Q \in \langle P \rangle$.

$$\begin{aligned}P(x,y) \quad = \quad & (u^{29} + u^{28} + zu^{26} + zu^{25} + z^2u^{24} + zu^{23} + z^2u^{22} + z^2u^{21} + z^2u^{20} + zu^{19} + \\ & z^2u^{18} + z^2u^{17} + u^{16} + z^2u^{13} + z^2u^{12} + u^{11} + zu^8 + z^2u^5 + u^4 + zu^3 + z^2u^2 \\ & + z^2u + z, zu^{30} + z^2u^{29} + z^2u^{27} + z^2u^{25} + zu^{24} + z^2u^{22} + zu^{20} + u^{19} + u^{18} \\ & + zu^{17} + u^{15} + zu^{13} + zu^{12} + z^2u^{10} + z^2u^9 + u^7 + u^5 + zu^4 + zu^3 + zu^2 \\ & + zu + 1),\end{aligned}$$

$$\begin{aligned}Q(x,y) \quad = \quad & (u^{27} + zu^{26} + zu^{25} + z^2u^{24} + zu^{23} + zu^{22} + z^2u^{21} + zu^{19} + z^2u^{18} + zu^{17} + \\ & z^2u^{16} + z^2u^{15} + u^{14} + u^{13} + zu^{12} + zu^10 + zu^8 + z^2u^6 + z^2u^5 + zu^4 + zu^3 \\ & + z^2u + z, u^{30} + z^2u^{29} + u^{28} + z^2u^{27} + z^2u^{25} + u^{24} + zu^{22} + z^2u^{21} + z^2u^{20} \\ & + zu^{19} + u^{18} + u^{17} + u^{15} + zu^{13} + u^{12} + z^2u^{11} + u^{10} + z^2u^9 + z^2u^8 + u^7 + \\ & z^2u^6 + zu^5 + u^4 + zu^2 + zu + 1).\end{aligned}$$

Los puntos P, Q son proyectados al jacobiano $J_H(\mathbb{F}_{2^2})$ hacia los divisores D_P y D_Q, respectivamente.

$$\begin{aligned}\mathrm{D}_P \quad = \quad & (x^{31} + zx^{30} + z^2x^{27} + x^{25} + zx^{24} + x^{23} + z^2x^{22} + x^{21} + z^2x^{20} + z^2x^{19} + zx^{18} \\ & + x^{17} + zx^{16} + zx^{15} + z^2x^{14} + x^{13} + zx^{12} + zx^9 + z^2x^8 + zx^7 + zx^6 + x^5 + \\ & zx^4 + x^3 + zx^2 + z^2x, z^2x^{30} + x^{28} + zx^{27} + z^2x^{24} + z^2x^{23} + zx^{22} + zx^{20} + x^{19} \\ & + z^2x^{18} + zx^{16} + z^2x^{15} + zx^{13} + z^2x^{12} + z^2x^{10} + zx^9 + zx^8 + zx^7 + zx^6 + \\ & z^2x^5 + x^4 + zx^3 + x),\end{aligned}$$

$$\begin{aligned}\mathrm{D}_Q \quad = \quad & (x^{32} + z^2x^{31} + x^{30} + zx^{29} + z^2x^{27} + x^{26} + z^2x^{25} + z^2x^{24} + z^2x^{23} + zx^{22} + \\ & x^{20} + x^{19} + z^2x^{18} + x^{17} + zx^{16} + z^2x^{15} + z^2x^{14} + z^2x^{13} + z^2x^{12} + x^{11} + \\ & z^2x^10 + zx^9 + zx^8 + z^2x^7 + zx^5 + zx^4 + zx^3 + x^2 + x, z^2x^{31} + x^{30} + zx^{29} + \\ & zx^{27} + z^2x^{26} + z^2x^{25} + z^2x^{24} + x^{23} + x^{22} + x^{21} + x^{20} + x^{19} + zx^{18} + zx^{17} \\ & + x^{16} + x^{15} + z^2x^{14} + zx^{12} + x^{11} + zx^{10} + zx^7 + zx^6 + zx^4 + zx).\end{aligned}$$

6.4.1.2. Aplicación del algoritmo de Enge-Gaudry

Para la obtención de la cota de suavidad $\mathtt{d_G}$, es decir, el máximo grado de los polinomios irreducibles permitidos en la base de factores. Se tiene que $\mathtt{d_G} = \left\lceil \log_{2^2} L_{2^{2^{32}}}\left[\frac{1}{2}, \varrho\right]\right\rceil$ donde $\varrho = \sqrt{\frac{1}{2} + \frac{1}{4\vartheta}} - \sqrt{\frac{1}{4\vartheta}}$ para algún entero positivo ϑ el cual satisface (i) $32 \geq \vartheta \log 2^2$ y (ii) $2^2 \leq L_{2^{2^{32}}}\left[\frac{1}{2}, \frac{1}{\sqrt{\vartheta}}\right]$. Luego, $\mathtt{d_G} \in \{4, 5, 6\}$.

Observación 6.4.2 *Experimentalmente, al construir la base de factores dinámicamente se obtiene que al finalizar la etapa de recolección de relaciones, ésta está compuesta sólo de una parte del total de los polinomios irreducibles de grado menor o igual a* $\mathtt{d_G}$*. Por dicha razón, con el propósito de aproximar el número de polinomios irreducibles de grado menor o igual a* $\mathtt{d_G}$*, se optó por usar como cota de suavidad* $\mathtt{d_G} \leftarrow \mathtt{d_G} + 1$*.*

Por lo tanto, sea $\mathtt{d_G} = 7$. Entonces, mediante la ayuda de la implemetación realizada en **Magma**, se obtiene que $P = \lambda Q$ donde $\lambda = 2288059015772263$. El cuadro 6.4 muestra los tiempos de ejecución de las dos etapas principales del algoritmo de Enge-Gaudry [1]. La figura 6.1 muestra los tiempos para diferentes cotas de suavidad en el intervalo $[5, 12]$, de los cuales se puede observar que $\mathtt{d_G} = 11$ es el que mejor"balancea" dichos tiempos [2].

Promedio por cada relación	0.212 s
Recolección de relaciones	310.160 s
Álgebra Lineal (Lanczos)	0.100 s

Cuadro 6.4: Tiempos de ejecución de la adaptación del algoritmo de Enge-Gaudry

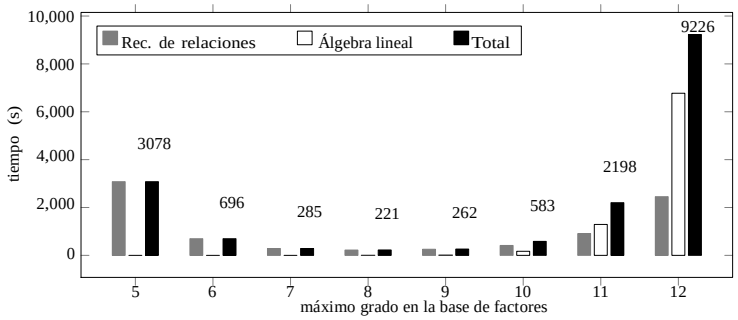

Figura 6.1: Tiempos de ejecución para $\mathtt{d_G} \in [5, 12]$

El problema del balanceo de esta adaptación del algoritmo de Enge-Gaudry es levemente diferente al algoritmo tradicional debido a que el costo de hallar una relación válida y la proporición α/β decrece conforme incrementa el número de elementos en la base de factores (véase el cuadro 6.5 [2]. Nótese que la probabilidad de tener relaciones con factores que no están incluidos en nuestras bases de factores aumenta acorde a un mayor número de polinomios irreducibles. Como consecuencia, al obtener una relación válida nos vemos en la problemática de

[1] Los tiempos de ejecución de cada etapa fueron realizados en un máquina cuyo procesador es un Intel(R) Core(TM) i5-3230M 2.60 GHz

[2] Los tiempos de ejecución de cada etapa fueron realizados en una máquina cuyo procesador es un Intel(R) Core(TM) i7-4700MQ 2.40 GHz .

que se añaden más factores y el costo para conseguir una matriz con el mismo número de columnas y filas también aumenta. Este efecto se puede observar en la figura 6.2 [2] (la gráfica del centro muestra el mejor desempeño para la fase de generación de relaciones).

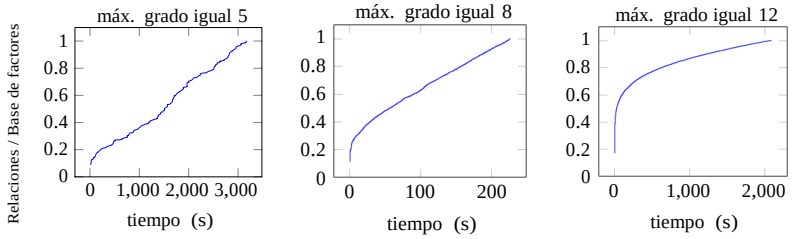

Figura 6.2: Proporción entre el número de relaciones válidas obtenidas y el número polinomios en la base de factores

	Máximo grado de la base de factores (d_G)							
	5	6	7	8	9	10	**11**	12
Etapa de recolección de relaciones								
Número de polinomios irreducibles de grado menor o igual a d_G (α)	294	964	3304	11464	40584	145338	**526638**	1924378
Tamaño de la base de factores (β)	152	474	1458	4352	12980	34883	**91793**	214116
Proporción β/α	0.517	0.492	0.441	0.380	0.320	0.240	**0.174**	0.111
Número de bits de las operaciones requeridas (teoricamente hablando)	23.24	20.88	19.50	19.08	19.16	19.49	**20.03**	20.57
Tiempo promedio de ejecución por relación	20.250	1.469	0.195	0.050	0.019	0.012	**0.009**	0.011
Tiempo de ejecución del algoritmo de Enge-Gaudry	3078.14	646.43	284.52	220.12	252.05	413.15	**909.12**	2451.80
Estimación del tiempo de ejecución del algoritmo de Enge-Gaudry (teoricamente hablando)	5953.50	1416.12	644.28	573.20	771.10	1744.06	**4739.74**	21168.16
Etapa del álgebra lineal								
Número de bits de las operaciones requeridas (teoricamente hablando)	17.49	20.58	23.71	26.75	29.78	32.55	**35.29**	37.65
Tiempo de ejecución	0.01	0.03	0.11	0.87	9.62	169.78	**1288.65**	6774.26

Cuadro 6.5: Diferentes configuraciones de la adaptación del algoritmo de Enge-Gaudry

6.4.2. Determinando vulnerabilidad ante el ataque gGHS

Sean $n \in \{127, 254\}$, $q = 2^{\frac{2 \cdot 127}{n}}$ y $\mathbb{F}_{2^{2 \cdot 127}}$ una extensión de grado n del campo $\mathbb{F}_q$. Entonces, $x^{127} + 1 = (x+1)f_1 \cdots f_{18}$ donde $f_i \in \mathbb{F}_2[x]$ es irreducible y $\text{grad}(f_i) = 7$ para $i = 1, \ldots, 18$. Los cuadros 6.6 y 6.7 enumeran los pares de polinmios en $\mathbb{F}_q[x]$ que construyen curvas vulnerables, es decir, los elementos del conjunto $\mathcal{S}$ para $q = 2$ $q = 2^2$ [Hankerson et al., 2011].

Cuadro 6.6: Pares de polinomios $(\mathrm{Ord}_{\gamma_1}, \mathrm{Ord}_{\gamma_2}) \in \mathcal{S}$ para el caso $n = 127$, $q = 2^2$.

Ord_{γ_1}	Ord_{γ_2}	$\mathrm{grad}(\mathrm{Ord}_{\gamma_1})$	$\mathrm{grad}(\mathrm{Ord}_{\gamma_2})$	m	género	Seguridad en bits del algoritmo de Enge-Gaudry
$(x+1)f_i$	$(x+1)f_i$	8	8	8	255	92.92
$(x+1)f_i$	f_i	8	7	8	254	92.71
$(x+1)f_i$	$x+1$	8	1	8	128	61.85
f_i	f_i	7	7	8	253	92.50
f_i	$x+1$	7	1	8	127	61.56

Cuadro 6.7: Pares de polinomios $(\mathrm{Ord}_{\gamma_1}, \mathrm{Ord}_{\gamma_2}) \in \mathcal{S}$ para el caso $n = 254,\ q = 2$.

Ord_{γ_1}	Ord_{γ_2}	$\mathrm{grad}(\mathrm{Ord}_{\gamma_1})$	$\mathrm{grad}(\mathrm{Ord}_{\gamma_2})$	m	género	Seguridad en bits del algoritmo de Enge-Gaudry
$(x+1)^2 f_i$	$(x+1)^2 f_i$	9	9	9	511	93.03
$(x+1)^2 f_i$	$(x+1)f_i$	9	8	9	510	92.92
$(x+1)^2 f_i$	f_i	9	7	9	508	92.71
$(x+1)^2 f_i$	$(x+1)^2$	9	2	9	384	78.66
$(x+1)^2 f_i$	$(x+1)$	9	1	9	256	61.85
$(x+1)f_i$	$(x+1)^2$	8	2	9	383	78.54
f_i	$(x+1)^2$	7	2	9	381	78.30

Como consecuencia, es posible verificar que la curva elíptica GLS $\mathcal{E}/\mathbb{F}_{2^{2\cdot 127}}$ usada en [Oliveira et al., 2014], la cual está dada por la ecuación 6.1 con parámetros $a = u \in \mathbb{F}_{2^{2\cdot 127}}$ y $b \in \mathbb{F}_{2^{127}}$ es un polinomio de grado 126 cuya representación en hexadecimal es $0x59C8202CB9E6E0AE2E6D944FA54DE7E5$, no es vulnerable ante el ataque gGHS debido a que $\nexists (s_1, s_2) \in \mathcal{S}$ tal que $\mathrm{mcd}\,(b_1(x), \bar{s}_2(x)) \neq 1$ tanto para $q = 2$ como para $q = 2^2$. A continuación se muestra una salida de una ejecución de la implementación del algoritmo 6.1:

```
Loading file "gls_test.magma"
Loading "setup.magma"

Building the polynomials used in the vulnerability test
Loading "AlgVul.magma"
time:= 0.000
E is an Elliptic Curve defined by y^2 + x*y = x^3 + u*x^2 + (v^126 +
                                    v^124 + v^123 + v^120 + v^119 + v^118
                                  + v^115 + v^109 + v^101 + v^99 + v^98
                                  + v^95 + v^93 + v^92 + v^91 + v^88 +
                                  v^87 + v^86 + v^85 + v^82 + v^81 + v^79
                                  + v^78 + v^77 + v^71 + v^69 + v^67 +
                                  v^66 + v^65 + v^61 + v^59 + v^58 + v^57
                                  + v^54 + v^53 + v^51 + v^50 + v^48 +
                                  v^47 + v^44 + v^42 + v^38 + v^35 + v^34
                                  + v^33 + v^32 + v^31 + v^29 + v^26 +
                                  v^24 + v^22 + v^19 + v^18 + v^16 + v^15
                                  + v^14 + v^13 + v^10 + v^9 + v^8 + v^7
                                  + v^6 + v^5 + v^2 + 1) over GF(2^254)
```

```
Is E vulnerable over F_{2^{254}}? -> false
The total time for answering the vulnerability for F_{2^{254}} was 0.630
Is E vulnerable over F_{{2^{2}}^{127}}? -> false
The total time for answering the vulnerability for F_{{2^{2}}^{127}} was
268.930
```

Observación 6.4.3 *El tiempo total de ejecución del algoritmo 6.1 fue de* 269.56 *segundos.*

Observación 6.4.4 *En* [Hankerson et al., 2011] *se brinda una cota superior para el tamaño del conjunto $\mathcal{B}$, la cual es 2^{36} para el caso $q = 2^2$ y $n = 127$. Con base a esta cota superior, y con ayuda de* ***Magma****, se puede verificar que al realizar una búsqueda "ingenua" (es decir, verificar si alguno de estos 2^{36} elementos coincide con el parámetro b), se requieren aproximadamente* 14417.92 *segundos de ejecución (para el caso $q = 2$ y $n = 2 \cdot 127$, la cota superior es de 2^{21} elementos y el tiempo de ejecución es aproximadamente* 0.440 *segundos)* [1]*. Como consecuencia, la aplicación del algoritmo 6.1 resulta más eficiente que realizar la búsqueda "ingenua".*

Observación 6.4.5 *Debido a que el número de curvas isógenas a una curva elíptica dada, para este ejemplo, es aproximadamente 2^{127} y el número de clases de isogenias de curvas elípticas vulnerables ante el ataque gGHS es aproximadamente $2^{36} + 2^{20}$* [Hankerson et al., 2011]*, entonces el algoritmo 6.2 se vuelve impráctico y resulta más eficientes realizar la aproximación de Hankerson, Karabina y Menezes (2011).*

Capítulo 7

Conclusiones

La ciencia es una ecuación diferencial. La religión es una condición de frontera.

Alan Turing

En este trabajo de tesis se propuso un mecanismo para determinar una posible vulnerabilidad ante el ataque gGHS [Hess, 2004], el cual puede ser extendido para abordar vulnerabilidad ante el ataque eGHS [Galbraith et al., 2002]. De igual manera, se realizaron las implementaciones en **Magma** del ataque GHS [Gaudry et al., 2002] y de una adaptación del algoritmo de Enge-Gaudry [Enge y Gaudry, 2002], la cual maneja una base de factores dinámica.

Para concluir con este trabajo de tesis, a continuación se hace un resumen de los principales resultados obtenidos, así como el trabajo a futuro que puede llegar a ser desarrollado.

Resultados obtenidos:

- Al manejar una construcción dinámica de la base de factores usada en el algoritmo de Enge-Gaudry, se obtuvo que al final de la fase de generación de relaciones, sólo un 44.12 % del total de la base de factores fue usada. Por consiguiente, en la práctica es mucho más eficiente manejar una base de factores dinámica.
- Los teoremas y corolarios de la sección 6.2, de los cuales el más importante es el corolario 6.2.3 debido a que el mecanismo de vulnerabilidad es resultado de dicho corolario.
- Los algoritmos 6.1 y 6.2, los cuales describen el mecanismo de vulnerabilidad propuesto en este trabajo de tesis.
- Implementación del ataque GHS y del mecanismo de vulnerabilidad en **Magma**.

- Implementación de un ataque exitoso (mediante el ataque GHS) contra una curva elíptica GLS definida sobre el campo $\mathbb{F}_{2^{62}}$ con la ayuda de **Magma**.
- No existen curvas elípticas GLS vulnerables ante el ataque gGHS y eGHS definidas sobre el campo $\mathbb{F}_{2^{2\cdot 257}}$ (véase el cuadro 6.1).
- La curva usada en [Oliveira et al., 2014] no es vulnerable ante el ataque gGHS.
- Publicación de un artículo, el cual engloba todo lo anterior [Chi y Oliveira, 2015].

Trabajo a futuro:

- Extensión del algoritmo 6.1 al caso del ataque eGHS mediante una decripción explícita de cómo debe ser una curva isógena. Por ejemplo, si $\mathcal{E}_{a,b}/\mathrm{k}$ es una curva elíptica binaria, determinar la estructura de un homorfismo $\pi\colon \mathrm{k} \to \mathrm{k}$ que traslade el parámetro b hacia su parámetro isógeno, donde parámetro isógeno se refiere al parámetro $\pi(b)$ que satisface $\#\mathcal{E}_{a,b}(\mathrm{k}) = \#\mathcal{E}_{a,\pi(b)}(\mathrm{k})$; de tal manera que el algoritmo 6.2 pueda mejorarse.
- Realizar un análisis, similar al estudiado en este trabajo de tesis, ante el ataque GHS, gGHS y eGHS sobre curvas hiperelípticas de género 2. Para así determinar si es posible construir sistemas criptográficos sobre curvas hiperelípticas de género 2 que sean seguras ante el ataque GHS, gGHS y eGHS. Por ejemplo, Galbraith (2003) realiza una aplicación del ataque GHS a una familia de curvas hiperelípticas.
- Comprender las implicaciones práticas de diferentes aproximaciones usadas para la resolución del Problema del Logaritmo Discreto en curva elípticas; por ejemplo: los métodos de Gaudry [Gaudry, 2009] y Diem [Diem, 2011], los cuales están basados del trabajo de Semaev [Semaev, 2004].

Apéndice A

En este apéndice se muestra la implementación de la adaptación del algoritmo de Enge-Gaudry [Enge y Gaudry, 2002] y la implementación del ataque GHS [Gaudry et al., 2002]

Para la implementación de la adaptación del algoritmo de Enge-Gaudry se implementaron dos funciones auxiliares, las cuales construyen la matriz asociada a la fase del álgebra lineal de dicho algoritmo. De igual manera, se realizó una implementación "ingenua" de la prueba de suavidad para divisores del jacobiano de una curva hiperelíptica dada (donde ingenua se refiere a verificar si, dado un divisor $\mathrm{div}(\mathtt{u}, \mathtt{v})$ del jacobiano de la curva hiperelíptica, el correspondiente polinomio $\mathtt{u}$ se factoriza como el producto de polinomios de grado menor o igual a la cota de suavidad).

Por la parte de la implementación del ataque GHS, ésta se realizó para una curva en particular, la curva elíptica binaria del ejemplo de la subsección 6.4.1 definida sobre el campo finito $\mathbb{F}_{2^{2\cdot 31}}$, la cual es vulnerable ante el ataque gGHS.

Código 1: Implementación ingenua para la prueba de suavidad

```
smooth_test:= function(D, var_deg)
    var_aux:= Factorisation(D[1]);
    for var_i in [1 .. #var_aux] do
        if( Degree(var_aux[var_i][1]) gt var_deg) then
            return false;
                end if;
    end for;
    return true;
end function;
```

Código 2: Primera función auxiliar usada en el algoritmo de Enge-Gaudry

```
fun00:= function(Mtrx)
        nx:= #Mtrx; mx:= 1;
        for i in [2 .. nx] do
                if(#Mtrx[mx] lt #Mtrx[i])
                then mx:= i; end if;
```

```
        end for;
        return nx, #Mtrx[mx];
end function;
```

Código 3: Segunda función auxiliar usada en el algoritmo de Enge-Gaudry

```
fun01:= function(Mtrx)
        Mr, Mc:= fun00(Mtrx);
        M2:= SparseMatrix(IntegerRing(), Mr, Mc);
        for i in [1 .. Mr] do
                for j in [1 .. Mc] do
                        if(IsDefined(Mtrx,[i,j])) then
                        SetEntry(~M2,i,j,Mtrx[i,j]);
                        end if;
                end for;
        end for;
        return M2;
end function;
```

Código 4: Implementación de la adaptación del algoritmo de Enge-Gaudry

```
Alg_EngeGaudry:= function(D1, D2, var_r, var_parameter, var_deg)
    Sp:= [];
    alpha:= [];
    beta:= [];
    T:= [];

    for i in [1 .. var_parameter] do
        alpha[i] := Random(var_r);
        beta[i] := Random(var_r);
        T[i] := alpha[i]*D1 + beta[i]*D2;
    end for;

    Alp:= [];
    Bta:= [];
    Alp0:= Random(var_r);
    Bta0:= Random(var_r);
    R0:= Alp0*D1 + Bta0*D2;
    cnt0:= 0;
    print "\nThe main loop will start";
    Mp:= [];

    RtionsITime:= Cputime();
    while(cnt0 lt (#Sp + 1) ) do
        cnt0 +:= 1;
        Mp[cnt0]:=[];
        var_j:= Random(1,var_parameter);
        R0:= R0 + T[var_j];
        Alp0:= (Alp0 + alpha[var_j]) mod var_r;
        Bta0:= (Bta0 + beta[var_j]) mod var_r;

        while(smooth_test(R0, var_deg) eq false) do
            var_j:= Random(1, var_parameter);
            R0:= R0 + T[var_j];
            Alp0:= (Alp0 + alpha[var_j]) mod var_r;
            Bta0:= (Bta0 + beta[var_j]) mod var_r;
        end while;

        fcts:= Factorisation(R0[1]);
```

```
        for var_j in [1..#fcts] do
            var_aux1:= J![fcts[var_j][1], R0[2] mod fcts[var_j][1]];
            if(( (var_aux1 in Sp) or (-var_aux1 in Sp)) eq false)
                then Append(~Sp, var_aux1);
            end if;
            indxs:= Index(Sp, var_aux1) + Index(Sp,-var_aux1);
            sgn:= (Index(Sp, var_aux1) - Index(Sp,-var_aux1)) div indxs;
            Mp[cnt0,indxs]:= sgn * fcts[var_j][2];
        end for;

        Alp[cnt0]:= Alp0;
        Bta[cnt0]:= Bta0;
    end while;

    printf "Factor base size -> %o", #Sp;
    RtionsFTime:= Cputime();
    print "\nSolving the Linear Algebra";
    M:= fun01(Mp);
    MtxITime:= Cputime();
    gamma:= ModularSolution(Transpose(M), var_r: Lanczos:=true);
    MtxFTime:= Cputime();

    den:= 0;
    enu:= 0;
    for i in [1 .. #Bta] do
        den:= den + Bta[i]*gamma[i];
        enu:= enu + Alp[i]*gamma[i];
    end for;

    Rtime:= RtionsFTime - RtionsITime;
    Mtime:= MtxFTime - MtxITime;
    den:= R!den;
    enu:= R!enu;
    if(den ne 0) then
        stion:= -enu/den;
        return Rtime, Mtime, IntegerRing()!stion;
    end if;

    print "Failed! Build another associated matrix";
    return Rtime,Mtime,R!0;
end function;
```

Código 5: Implementación del ataque GHS

```
clear;
/*********************************
 *             Setup             *
 *********************************/
k<v>:= ExtensionField<GF(2), v | v^17 + v^6 + v^5 + v^3 + 1 >;
K<u>:= ExtensionField<k, u | u^2 + u + 1>;
P<x>:= PolynomialRing(k);

/*********************************
 *   Building the Elliptic Curve  *
 *********************************/
a  := K!u;
b  := v^66055*u + v^54867;
E  := EllipticCurve([K| 1, a, 0, 0, b]);
E_size:= #E;
Factores:= Factorization(E_size);
r := Factores[#Factores,1];
c := E_size div r;

/*********************************
```

```
 *          Weil descent          *
 *********************************/
time H, Weil_map := WeilDescent(E,k, K!1);
J := Jacobian(H);

/*********************************
 *   Reducing into the Jacobian   *
 *********************************/
Pt:= c*Random(E);
key:= Random(r);
Ptp:= key*Pt;

D := Weil_map(Pt);
Dp:= Weil_map(Ptp);
R := ResidueClassRing(r);

/*********************************
 *  Solving de DLP on the Jaobian *
 *********************************/
load "Alg_EngeGaudry.magma";
Rt,Mt,lg:= Alg_EngeGaudry(D,Dp,r,101,1);
print Rt, Mt, lg eq key;
exit;
```

Apéndice B

En este apéndice se muestra la implementación del algoritmo 6.1. Para ello, se realizó un script el cual genera la configuración requerida para dicho algoritmo aplicado a los dos posibles casos, es decir, el algoritmo aplicado sobre $\mathbb{F}_{q^n}$ para $(q, n) = (2, 2\ell)$ y $(q, n) = (2^2, \ell)$, donde $\ell = 127$. De igual manera, se realizó un script que aplica el algoritmo 6.1 a la curva elíptica GLS usada en [Oliveira et al., 2014], la cual está definida sobre el campo $\mathbb{F}_{2^{2 \cdot 127}}$.

Código 6: Implementación del algoritmo 6.1

```
AlgVul:= function(var_b, var_S, var, var_frob, var_PolyRing)
    beta:= Sqrt(var_b);
    au0x01:= var_PolyRing!1;
    jj:= 0;

    while( (au0x01 eq var_PolyRing!1) and (jj lt #var_S) ) do
        jj:= jj + 1;
        var_s1 := MultiplyFrobenius(var, var_S[jj,1], var_frob);
        b_jj := Evaluate(var_s1, beta*var) div (beta^(Degree(var_s1))*var);
        var_s2 := MultiplyFrobenius(var, var_S[jj,2], var_frob);
        sp_jj:= Polynomial(Reverse(Coefficients(var_s2)));
        au0x01:= GCD(b_jj, sp_jj);
    end while;

    if(au0x01 ne var_PolyRing!1) then return jj, true; else return 0, false;
         end if;
end function;
```

Código 7: Configuración del algoritmo 6.1 para curvas elípticas GLS

```
P<x>:= PolynomialRing(GF(2));

F1<v>:= ext<GF(2)| x^127 + x^63 + 1>;
F2<U>:= ext<GF(2)| x^2 + x + 1>;
F3:= GF(2);

K1<u>:= ext<F1| x^2 + x + 1>;
K2<V>:= ext<F2| x^127 + x^63 + 1>;
K3<W>:= ext<GF(2)| x^254 + x^7 + x^2 + x + 1>;

Embed(K1,K2);
Embed(K2,K3);
Embed(K3,K1);

```

```
n:= 127;
d:= 7;

P<x>:= PolynomialRing(F3);

P1<y01>:= PolynomialRing(K1);
P2<y02>:= PolynomialRing(K2);
P3<y03>:= PolynomialRing(K3);

fact0x00:= Factorisation(x^n + 1);
fact:= [];
for jj in [1 .. #fact0x00] do
        fact[jj]:= fact0x00[jj,1];
end for;

frob01 := map< P2 -> P2 | z :-> z^#F2 >;
frob02 := map< P3 -> P3 | z :-> z^#F3 >;

f01:= MultiplyFrobenius(y02, fact[2], frob01);
f02:= MultiplyFrobenius(y03, fact[2], frob02);

print "\nBuilding the polynomials used in the vulnerability test";
Ti:= Cputime();
load "AlgVul.magma";
Tf:= Cputime();
print "time:=", Tf - Ti;
```

Código 8: Aplicación del algoritmo 6.1 a una curva elíptica GLS

```
clear;
load "setup.magma";
/*************************************************************************
 * GLS curve of Two is the fastest prime: lambda coordinates for binary *
 * elliptic curves.                                                      *
 *************************************************************************/

bK1:= K1!(F1!IntegerToSequence(0x59C8202CB9E6E0AE2E6D944FA54DE7E5,2));
bK2:= K2!bK1;
bK3:= K3!bK2;

aK1:= K1!u;
aK2:= F2!aK1;
aK3:= K3!aK2;

EK1:= EllipticCurve([K1| K1!1, aK1, 0, 0, bK1]);
EK2:= EllipticCurve([K2| K2!1, aK2, 0, 0, bK2]);
EK3:= EllipticCurve([K3| K3!1, aK3, 0, 0, bK3]);

r:= 0x1FFFFFFFFFFFFFFFFFFFFFFFFFFFFFFFDAC40D1195270779877DABA2A44750A5;

r1:= FactoredOrder(EK1);
r2:= FactoredOrder(EK2);
r3:= FactoredOrder(EK2);

print "E is an", EK1;
/*
print "\nb in representation F_{{2^127}^{2}} :=", bK1;
print "\nb in representation F_{{2^2}^{127}} :=", bK2;
print "\nb in representation F_{2^254} :=", bK3;
*/
S_F4:=[];
S_F2:=[];

for ii in [2 .. #fact] do
```

```
36          Append(~S_F4,[fact[ii],x+1]);
37          Append(~S_F4,[fact[ii],fact[ii]]);
38          Append(~S_F4,[(x+1)*fact[ii],x+1]);
39          Append(~S_F4,[(x+1)*fact[ii],fact[ii]]);
40          Append(~S_F4,[(x+1)*fact[ii],(x+1)*fact[ii]]);
41
42          Append(~S_F2,[fact[ii],x^2+1]);
43          Append(~S_F2,[(x+1)*fact[ii],x^2+1]);
44          Append(~S_F2,[(x^2+1)*fact[ii],x+1]);
45          Append(~S_F2,[(x^2+1)*fact[ii],x^2+1]);
46          Append(~S_F2,[(x^2+1)*fact[ii],fact[ii]]);
47          Append(~S_F2,[(x^2+1)*fact[ii],(x+1)*fact[ii]]);
48          Append(~S_F2,[(x^2+1)*fact[ii],(x^2+1)*fact[ii]]);
49
50  end for;
51
52  print "\n";
53
54  Ti0:= Cputime();
55  i01, it01:= AlgVul(bK3, S_F2, y03, frob02, P3);
56  Tf0:= Cputime();
57  print "Is E vulnerable over F_{2^{254}}? ->", it01;
58  print "The total time for answering the vulnerability over F_{2^{254}} was",
        Tf0 - Ti0;
59  Ti1:= Cputime();
60  i02, it02:= AlgVul(bK2, S_F4, y02, frob01, P2);
61  Tf1:= Cputime();
62  print "Is E vulnerable over F_{{2^{2}}^{127}}? ->", it02;
63  print "The total time for answering the vulnerability over F_{{2^{2}}^{127}}
        was", Tf1 - Ti1;
```

Bibliografía

[Katz y Lindell, 2007] J. Katz e Y. Lindell, *Introduction to Modern Cryptography*, Chapman & Hall/CRC, 2007.

[Diffie y Hellman, 1976] W. Diffie y M. E. Hellman, *New directions in cryptography*, Institute of Electrical and Electronics Engineers (IEEE) Transactions on Information Theory, Vol. 22, No. 6, pp. 644-654, IEEE Information Theory Society, 1976.

[Koblitz, 1987] N. Koblitz, *Elliptic curve cryptosystems*, Mathematics of Computation, Vol. 48, No. 177, pp. 203-209, American Mathematical Society (AMS), 1987.

[Miller, 1986] V. S. Miller, *Use of Elliptic Curves in Cryptography*, Advances in Cryptology - CRYPTO '85 Proceedings, Hugh C. Williams (Ed.), Springer Berlin Heidelberg, Vol. 218, Lecture Notes in Computer Science, pp. 417-426, 1986.

[Adleman et al., 1978] L. Adleman, R. Rivest y A. Shamir, *A Method for Obtaining Digital Signatures and Public-key Cryptosystems* , Communications of the ACM, Vol. 21, No. 2, pp. 120-126, 1978.

[Pollard, 1978] J. Pollard, *Monte Carlo methods for Index Computation (mod p)*, Mathematics of Computation, Vol. 32, No. 143, pp. 918-924, American Mathematical Society (AMS), 1978.

[Elgamal, 1985] T. Elgamal, *A Public Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms*, Advances in Cryptology: Proceedings of CRYPT0 84, George Robert Blakley y David Chaum (Eds.), Springer Berlin Heidelberg, Vol. 196, Lecture Notes in Computer Science, pp. 10-18, 1985.

[Frey, 1998] Gerhard Frey, *How to disguise an elliptic curve*, Talk at ECC'98 (Workshop on Elliptic Curve Cryptography), Waterloo, 1998. `http://www.cacr.math.uwaterloo.ca/conferences/1998/ecc98/frey.ps`

[Gaudry et al., 2002] P. Gaudry, F. Hess y N. P. Smart, *Constructive and destructive facets of Weil descent on elliptic curves*, Journal of Cryptology, Vol. 15, No. 1, pp. 19-46, 2002.

[Galbraith et al., 2002] S. D. Galbraith, F. Hess y N. P. Smart, *Extending the GHS Weil Descent Attack*, Advances in Cryptology — EUROCRYPT 2002, Lars Knudsen (Ed.), Springer Berlin Heidelberg, Vol. 2332, Lecture Notes in Computer Science, pp. 29-44, 2002.

[Hess, 2004] F. Hess, *Generalising the GHS Attack on the Elliptic Curve Discrete Logarithm Problem*, London Mathematical Society (LMS) Journal of Computation and Mathematics, Vol. 7, pp. 167-192, 2004.

[Menezes y Qu, 2001] Alfred Menezes y Minghua Qu, *Analysis of the Weil Descent Attack of Gaudry, Hess and Smart*, Topics in Cryptology — CT-RSA 2001, David Naccache (Ed.), Springer Berlin Heidelberg, Vol. 2020, Lecture Notes in Computer Science, pp. 308-318, 2001.

[Maurer et al., 2001] M. Maurer, A. Menezes y E. Teske, *Analysis of the GHS Weil Descent Attack on the ECDLP over Characteristic Two Finite Fields of Composite Degree*, Progress in Cryptology — INDOCRYPT 2001, C. Pandu Rangan y Cunsheng Ding (Eds.), Springer Berlin Heidelberg, Vol. 2247, Lecture Notes in Computer Science, pp. 195-213, 2001.

[Galbraith et al., 2009] S. D. Galbraith, X. Lin y M. Scott, *Endomorphisms for Faster Elliptic Curve Cryptography on a Large Class of Curves*, Advances in Cryptology — EUROCRYPT 2009, Antoine Joux (Ed.), Springer Berlin Heidelberg, Vol. 5479, Lecture Notes in Computer Science, pp. 518-535, 2009.

[Hankerson et al., 2011] D. Hankerson, K. Karabina y A. Menezes, *Analyzing the Galbraith-Lin-Scott Point Multiplication Method for Elliptic Curves over Binary Fields*, Institute of Electrical and Electronics Engineers (IEEE) Transactions on Computers, Vol. 58, No. 10, pp. 1411-1420, IEEE Computer Society, 2009.

[Judson, 2014] T. W. Judson, *Abstract Algebra: Theory and Applications, 2014th Edition*, Orthogonal Publishing L3C, 2014.

[Cormen et al., 2009] T. H. Cormen, C. Stein, R. L. Rivest y C. E. Leiserson, *Introduction to Algorithms, Third Edition*, The MIT Press, 2009.

[Guerrero y Pérez, 2010] E. A. Guerrero Lara y J. E. Pérez Terrazas, *Álgebra Abstracta. De grupos a preliminares de la Teoría de Galois*, Ediciones de la Universidad Autónoma de Yucatán, México, 2010.

[Cohen et al., 2012] H. Cohen, G. Frey, R. Avanzi, C. Doche, T. Lange, K. Nguyen y F. Vercauteren, *Handbook of Elliptic and Hyperelliptic Curve Cryptography, Second Edition*, Chapman & Hall/CRC, 2012.

[Hoffstein et al., 2008] J. Hoffstein, J. Pipher y J. H. Silverman, *An Introduction to Mathematical Cryptography*, Springer Publishing Company, Incorporated, 2008.

[Lidl y Niederreiter, 1997] R. Lidl y H. Niederreiter, *Finite Fields, Second edition*, Cambridge University Press, Vol. 20, Encyclopedia of Mathematics and its Applications, 1997.

[Friedberg et al., 2002] S. H. Friedberg, A. J. Insel y L. E. Spence, *Linear Algebra, Fourth Edition*, Pearson, 2002.

[Granger et al., 2014] R. Granger, T. Kleinjung y J. Zumbrägel, *On the Powers of 2*, Cryptology ePrint Archive, Report 2014/300, 2014. `http://eprint.iacr.org/2014/300`

[Hankerson et al., 2003] D. Hankerson, A. Menezes y S. Vanstone, *Guide to Elliptic Curve Cryptography*, Springer Berlin Heidelberg, New York, USA, 2003.

[Menezes et al., 1993] A. Menezes, S. Vanstone y T. Okamoto, *Reducing Elliptic Curve Logarithms to Logarithms in a Finite Field*, Institute of Electrical and Electronics Engineers (IEEE) Transactions on Information Theory, Vol. 39, No. 5, pp. 1639-1646 , 1993.

[Galbraith, 2012] S. D. Galbraith, *Mathematics of Public Key Cryptography*, Cambridge University Press, New York, USA, 2012.

[Tate, 1966] J. Tate, *Endomorphisms of abelian varieties over finite fields*, Inventiones mathematicae, Springer Berlin Heidelberg, Vol. 2, No. 2, pp. 134-144, 1966.

[Barbulescu et al., 2014] R. Barbulescu, P. Gaudry, A. Joux, y E. Thomé, *A Heuristic Quasi-Polynomial Algorithm for Discrete Logarithm in Finite Fields of Small Characteristic*, Advances in Cryptology - EUROCRYPT 2014, Phong Q. Nguyen y Elisabeth Oswald (Eds.), Springer Berlin Heidelberg, Vol. 8441, Lecture Notes in Computer Science, pp. 1-16, 2014.

[Blake et al., 2005] I. F. Blake, G. Seroussi y N. P. Smart, *Advances in Elliptic Curve Cryptography*, Cambridge University Press, New York, USA, 2005.

[Cantor, 1987] D. G. Cantor, *Computing in the Jacobian of a Hyperelliptic Curve*, Mathematics of Computation, Vol. 48, No. 177, pp. 95-95, 1987.

[Gaudry, 2000] P. Gaudry, *An Algorithm for Solving the Discrete Log Problem on Hyperelliptic Curves*, Advances in Cryptology — EUROCRYPT 2000, Bart Preneel (Ed), Springer Berlin Heidelberg, Vol. 1807, Lecture Notes in Computer Science, pp. 19-34, 2000.

[Enge y Gaudry, 2002] A. Enge y P. Gaudry, *A general framework for subexponential discrete logarithm algorithms*, Acta Arithmetica, Vol. 102, No. 1, pp. 83-103, 2002.

[Gaudry et al., 2007] P. Gaudry, E. Thomé, N. Thériault y C. Diem *A double large prime variation for small genus hyperelliptic index calculus*, Mathematics of Computation, Vol. 76, pp. 475-492, 2007.

[Sarkar y Singh, 2014] P. Sarkar y S. Singh, *A New Method for Decomposition in the Jacobian of Small Genus Hyperelliptic Curves*, Cryptology ePrint Archive, Report 2014/815, 2014. `http://eprint.iacr.org/2014/815`

[Sarkar y Singh, 2015] P. Sarkar y S. Singh, *A Simple Method for Obtaining Relations Among Factor Basis Elements for Special Hyperelliptic Curves*, Cryptology ePrint Archive, Report 2015/179, 2015. `http://eprint.iacr.org/2015/179`

[Nagao, 2010] K.-i. Nagao, *Decomposition Attack for the Jacobian of a Hyperelliptic Curve over an Extension Field*, Algorithmic Number Theory, Guillaume Hanrot, François Morain y Emmanuel Thomé (Eds.), Springer Berlin Heidelberg, Vol. 6197, Lecture Notes in Computer Science, pp.285-300, 2010.

[Joux y Vitse, 2012] A. Joux y V. Vitse, *Cover and Decomposition Index Calculus on Elliptic Curves Made Practical*, Advances in Cryptology — EUROCRYPT 2012, David Pointcheval y Thomas Johansson (Eds.), Springer Berlin Heidelberg, Vol. 7237, Lecture Notes in Computer Science, pp. 9-26, 2012

[Jacobson et al., 2001] M. Jacobson , A. Menezes y A. Stein, *Solving Elliptic Curve Discrete Logarithm Problems Using Weil Descent*, Journal of the Ramanujan Mathematical Society, Vol. 16, pp. 231-260, 2001.

[Gallant et al., 2001] R. P. Gallant, R. J. Lambert, y S. A. Vanstone, *Faster Point Multiplication on Elliptic Curves with Efficient Endomorphisms*, Advances in Cryptology — CRYPTO 2001, Joe Kilian (Ed.), Springer Berlin Heidelberg, Vol. 2139, Lecture Notes in Computer Science, pp. 190-200, 2001.

[Hu et al., 2012] Z. Hu, P. Longa, and M. Xu, *Implementing the 4-dimensional GLV method on GLS elliptic curves with j-invariant 0*, Designs, Codes and Cryptography, Springer Berlin Heidelberg, Vol. 63, No. 3, pp. 331-343, 2012.

[Longa et al., 2014] P. Longa and F. Sica, *Four-Dimensional Gallant-Lambert-Vanstone Scalar Multiplication*, Journal of Cryptography, Springer Berlin Heidelberg, Vol. 27, No. 2, pp. 248-283, 2014.

[Oliveira et al., 2014] T. Oliveira, J. López, D. F. Aranha, y F. Rodríguez-Henríquez, *Two is the fastest prime: lambda coordinates for binary elliptic curves*, Journal of Cryptography Engineering, Springer Berlin Heidelberg, Vol. 4, No. 1, pp. 3-17, 2014.

[Bos et al., 2013] J. W. Bos, C. Costello, H. Hisil, y K. Lauter, *High-Performance Scalar Multiplication Using 8-Dimensional GLV/GLS Decomposition*, CHES 2013, Springer Berlin Heidelberg, Vol. 8086, Lecture Notes in Computer Science, pp. 331-348, 2013.

[Faz-Hernández et al., 2014] A. Faz-Hernández, P. Longa, y A. H. Sánchez, *Efficient and Secure Algorithms for GLV-Based Scalar Multiplication and Their Implementation on GLV-GLS Curves*, Topics in Cryptology — CT-RSA 2014, Springer Berlin Heidelberg, Vol. 8366, Lecture Notes in Computer Science, pp. 1-27, 2014.

[Chi y Oliveira, 2015] J.-J. Chi y T. Oliveira, *Attacking a Binary GLS Elliptic Curve with Magma*, Progress in Cryptology - LATINCRYPT 2015, Kristin E. Lauter y Francisco Rodríguez-Henríquez (Eds.), Springer Berlin Heidelberg, Vol. 9230, Lecture Notes in Computer Science, pp. 308-326, 2015.

[Galbraith, 2003] S. D. Galbraith *Weil descent of Jacobians*, Discrete Applied Mathematics, Vol. 128, No. 1, pp. 165-180, 2003

[Diem, 2011] C. Diem, *On the discrete logarithm problem in elliptic curves*, Compositio Mathematica, Vol. 127, pp. 75-104, 2011.

[Gaudry, 2009] P. Gaudry, *Index calculus for abelian varieties of small dimension and the elliptic curve discrete logarithm problem*, Journal of Symbolic Computation, Vol. 44, Notes on Gröbner Bases in Cryptography, Coding Theory, and Algebraic Combinatorics, pp. 1690-1702, 2009.

[Semaev, 2004] I. Semaev, *Summation polynomials and the discrete logarithm problem on elliptic curves*, Cryptology ePrint Archive, Report 2004/031, 2004. `http://eprint.iacr.org/2004/031`