

Clase 13: preliminares matemáticos II

Fernando Virdia, versión: 0.1, mayo 2025

8 Grupos, y Suposiciones de Dificultad

A la base de gran parte de la PKC hay objetos de naturaleza algebraica: grupos, anillos, cuerpos. Nosotros vamos a intentar cubrir la menor cantidad posible, quedando cerca de lo utilizado en la practica.

Definición 25 (grupo abeliano). Sea G un conjunto, $\times$ una operación binaria $\times: G \times G \rightarrow G$. Decimos que $\mathbb{G} = (G, \times)$ es un grupo abeliano si

1. $\times$ es asociativa: $(a \times b) \times c = a \times (b \times c)$, $\forall a, b, c \in G$.
2. Existe un elemento neutro $e \in G$ tal que $a \times e = e \times a = a$, $\forall a \in G$.
3. Cada elemento a tiene un inverso a^{-1} , tal que $a \times a^{-1} = a^{-1} \times a = e$, $\forall a \in G$.
4. $\times$ es conmutativa: $a \times b = b \times a$, $\forall a, b \in G$.

Si $|G| < \infty$, decimos que $\mathbb{G}$ es finito, y que $|G|$ es el orden de $\mathbb{G}$. A menudo abusamos notación y escribimos $a \in \mathbb{G}$ en lugar de $a \in G$.

Ejemplo 4. • $(\mathbb{Z}, +)$ es un grupo abeliano

- $(\mathbb{Z}, \cdot)$ **no es** un grupo abeliano. $(\mathbb{Q}, \cdot)$ tampoco. $(\mathbb{Q} \setminus \{0\}, \cdot)$ lo es. $(\mathbb{R} \setminus \{0\}, \cdot)$ también.
- Sea $\mathbb{Z}_n$ el conjunto de los enteros “modulo n ”. $(\mathbb{Z}_n, +)$ es un grupo abeliano.

Comentario 46. En un grupo podemos “exponenciar”: si $a \in \mathbb{G}$, $a^n := a \times a \times \cdots \times a$ (n veces). Las comunes reglas valen:

- $a^n \times a^m = a^{n+m}$,
- $(a^n)^m = a^{nm} = a^{mn} = (a^m)^n$,
- $a^1 \cdot a^{-1} = a^0 = e$.

En el caso de un grupo donde usamos “+”, como $(\mathbb{Z}, +)$, el exponente es un factor: $a + \cdots + a$ (n veces) $= n \times a$.

Definición 26 (subgrupo). Sea $\mathbb{G} = (G, \times)$ un grupo, y sea $S \subset G$ un subconjunto de G . Si $\mathbb{S} = (S, \times)$ forma un grupo, decimos que $\mathbb{S}$ es un subgrupo de $\mathbb{G}$.

Ejemplo 5. Sea $\mathbb{G} = (\mathbb{Z}, +)$. $\mathbb{S} = (2\mathbb{Z}, +) = (\{2n \mid n \in \mathbb{Z}\}, +)$ es un subgrupo de $\mathbb{G}$.

Definición 27 (grupo cíclico). Un grupo $\mathbb{G} = (G, \times)$ se dice cíclico si existe $g \in G$ tal que el grupo $\langle g \rangle$ “generado” g ,

$$\langle g \rangle := (\{g^n \mid n \in \mathbb{Z}\}, \times),$$

es $\mathbb{G}$, o sea $\mathbb{G} = \langle g \rangle$.

Ejemplo 6. $(\mathbb{Z}, +)$ es cíclico, y generado por ± 1 .

Lemma 14. Dado un grupo $\mathbb{G} = (G, \times)$ de orden primo $|G| = p$, $\mathbb{G}$ es cíclico y generado por cualquier elemento no neutro.

Ejemplo 7. $(\mathbb{Z}_p, +)$ es cíclico. Cualquier $1 \leq g < p$ lo genera. Por ejemplo, sea $p = 3$.

1 es un generador		2 es un generador	
1	mod 3 = 1	2	mod 3 = 2
1 + 1	mod 3 = 2	2 + 2	mod 3 = 4 mod 3 = 1
1 + 1 + 1	mod 3 = 3	2 + 2 + 2	mod 3 = 6 mod 3 = 0
1 + 1 + 1 + 1	mod 3 = 4 mod 3 = 1	2 + 2 + 2 + 2	mod 3 = 8 mod 3 = 2

Aquí podríamos adentrarnos en mas detalles sobre los grupos usados en criptografía, pero se vuelve muy matemático. En su lugar, vamos a definir tres problemas matemáticos relacionados, y un postulado. En criptografía, usamos familias de grupos que satisfacen el postulado.

Definición 28 (DLOG, informal). Sea $\mathbb{G}$ un grupo cíclico. Dados $g, h \in \mathbb{G}$, el problema del logaritmo discreto (DLOG, “discrete logarithm”) requiere recuperar $x \in \mathbb{Z}$ tal que $g^x = h$, o sea “ $x = \log_g h$ ”.

Comentario 47. DLOG es fácil de resolver en $(\mathbb{Z}, +)$ y $(\mathbb{R} \setminus \{0\}, \times)$:

- $(\mathbb{Z}, +)$: recordando que “ g^x ” es $x \cdot g$, $x \cdot g = h \iff x = h/g$.
- $(\mathbb{R} \setminus \{0\}, \times)$: $g^x = h \iff x = \log_g h$, donde $\log$ es el logaritmo “común”.

Definición 29 (CDH, informal). Sea $\mathbb{G}$ un grupo cíclico finito con generador g . Sean $x \xleftarrow{\$} [|\mathbb{G}|]$, $y \xleftarrow{\$} [|\mathbb{G}|]$ muestreados independientemente. El problema computacional de Diffie-Hellman (CDH, “computational Diffie-Hellman”) requiere, dados (g, g^x, g^y) , calcular $g^{xy} = (g^x)^y = (g^y)^x$.

Comentario 48. CDH no puede ser mas difícil de DLOG (o sea, resolver DLOG $\Rightarrow$ resolver CDH). Supongamos que $\mathcal{O}$ es un oráculo que calcula $(g, g^x) \mapsto x$. Dados (g, g^x, g^y) , calculamos $(g^y)^{\mathcal{O}(g, g^x)} = (g^y)^x = g^{xy}$.

Lemma 15. Sea $\mathbb{G} = \langle g \rangle$ un grupo finito. Dada la variable aleatoria $z \sim U([|\mathbb{G}|])$, g^z es una variable aleatoria $g^z \sim U(\mathbb{G})$.

Proof.

$$\forall h \in \mathbb{G}, \quad \Pr_{z \sim U([|\mathbb{G}|])} [g^z = h] = \Pr[z = \log_g h] = \frac{1}{|\mathbb{G}|}.$$

□

Definición 30 (DDH). Sea $\mathbb{G} = \langle g \rangle$ un grupo cíclico finito. Para simplificar notación, supongamos que g es un “parámetro público”, o sea un elemento fijo y disponible públicamente. Definimos el juego Exp^{DDH} .

$\text{Exp}^{\text{DDH}}(\mathcal{A}, b)$	
1:	$x \xleftarrow{\$} [\mathbb{G}]$
2:	$y \xleftarrow{\$} [\mathbb{G}]$
3:	if $b = 0$
4:	$z \leftarrow x \cdot y$
5:	else
6:	$z \xleftarrow{\$} [\mathbb{G}]$
7:	$b' \leftarrow \mathcal{A}(g^x, g^y, g^z)$
8:	return b'

Decimos que $\mathbb{G}$ otorga seguridad (ε, t) -DDH si cualquier adversario $\mathcal{A}$ que corre en tiempo $\leq t$ tiene ventaja

$$\text{Adv}(\text{Exp}^{\text{DDH}}, \mathcal{A}) := \left| \Pr[\text{Exp}^{\text{DDH}}(\mathcal{A}, 0) \Rightarrow 1] - \Pr[\text{Exp}^{\text{DDH}}(\mathcal{A}, 1) \Rightarrow 1] \right| \leq \varepsilon.$$

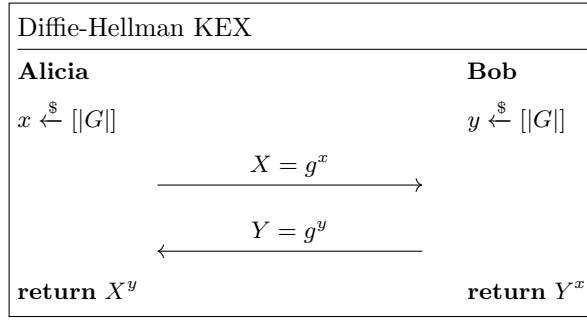
Finalmente introducimos un nuevo postulado:

Postulado 3. Dado $\lambda > 0$, existe un grupo cíclico finito $\mathbb{G}$ que otorga λ -bits de seguridad DDH.

Comentario 49. Por motivos criptoanalíticos, es generalmente deseable que $\mathbb{G}$ tenga orden primo.

Vamos ahora a introducir brevemente el intercambio de claves (KEX, key exchange) de Diffie y Hellman. Lo hacemos por motivos informacionales e históricos, al ser el primer protocolo de PKC considerado seguro. No vamos a intentar demostrar su seguridad en detalle, porque los modelos de seguridad de KEX pueden ser muy complicados.

Supongamos que dos partes Alicia y Bob se ponen de acuerdo publicamente sobre un grupo cíclico finito $\mathbb{G}$ y sobre un generador g , tal que $\mathbb{G} = \langle g \rangle$.

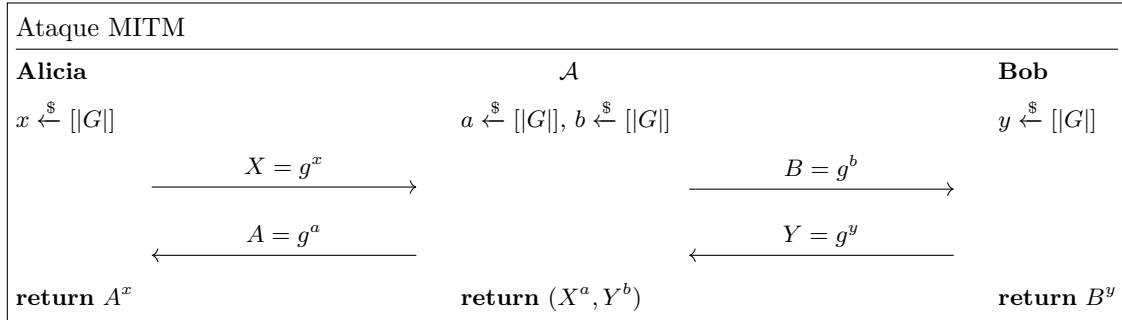


Notamos que $X^y = (g^x)^y = g^{xy} = g^{yx} = (g^y)^x = Y^x$, por lo cual Alicia y Bob retornan el mismo elemento $g^{xy} \in \mathbb{G}$. Este puede ser utilizado para generar una clave simétrica k , de modo de continuar comunicando a través de un SKES seguro, por ejemplo un AEAD. Una opción común es definir una función de hash $H: \mathbb{G} \rightarrow \{0, 1\}^\ell$, y utilizar $H(g^{xy})$ como clave.

Definir la seguridad del KEX puede ser complicado. Supongamos un adversario que se limita a observar pasivamente la comunicación (el “transcript”) (g^x, g^y) . Si CDH es difícil en $\mathbb{G}$, sabemos que calcular g^{xy} es difícil. Es esto suficiente para garantizar que DH es seguro? CDH no excluye que “mitad de los bits” de g^{xy} sean aprendibles. Esto podría teóricamente resultar en vulnerar la siguiente fase de comunicación.

De aquí la idea de usar DDH: si (g^x, g^y, g^{xy}) es indistinguible de (g^x, g^y, g^z) con z aleatorio, $\mathcal{A}$ no podría averiguar si... De otro lado, si $\mathcal{A}$ tuviera una suposición $h \in \mathbb{G}$ para el valor de g^{xy} , aunque DDH sea difícil, podría utilizar la siguiente comunicación con clave $H(g^{xy})$ para determinar si adivino h correctamente.

Notamos también que $\mathcal{A}$ podría no limitarse simplemente a observar tráfico, y también interceptarlo si tiene control sobre la red. Este tipo de ataque se llama machine-in-the-middle (MITM).



En este caso $\mathcal{A}$ podría fingir de ser Bob a Alice, y Alicia a Bob. Por ejemplo, podría interceptar $\text{Enc}(H(g^{ax}), m)$ de Alicia a Bob, descifrarlo, volverlo a cifrar como $\text{Enc}(H(g^{by}), m)$, y finalmente mandarlo a Bob. Alicia y Bob no podrían detectar el ataque.

Este tipo de táctica no funciona si Alice y Bob intercambian claves secretas de persona, y utilizan exclusivamente criptografía simétrica. Defendernos de este tipo de táctica requiere nuevos instrumentos, para construir una “infraestructura de claves públicas” (PKI, public-key infrastructure).