IIC3253: Criptografía y Seguridad Computacional - Tarea #3

Raimundo Herrera - rjherrera@uc.cl
Colaborador(a): Thomas Reisenegger

6 de junio de 2018

Problema 1

Calcule a mano lo siguiente (puede ocupar el teorema del resto chino).

1. **Los últimos 2 díıgitos de 7^{35688}**.

   Lo pedido equivale a encontrar el resultado de $7^{35688} \mod 100$. Además, por el *hint* se tiene que

   $$\phi(100) = (2^1 \cdot 1)(5^1 \cdot 4) = 40.$$

   Por otro lado, sabemos $\forall a \in \mathbb{Z}_N^*$, $\forall x \in \mathbb{Z}$ se tiene que

   $$[a^x \mod N] = [a^{x \mod \phi(N)} \mod N]$$

   de modo que para el caso particular se tiene

   $$[7^{35688} \mod 100] = [7^{35688 \mod 40} \mod 100].$$

   Ahora, considerando que $[35688 \mod 40] = 8$ se tiene que

   $$\begin{aligned}
   [7^{35688 \mod 40} \mod 100] &= [7^8 \mod 100] \\
   &= [(7^4)^2 \mod 100] \\
   &= [(2401)^2 \mod 100] \\
   &= [(2401 \mod 100)^2 \mod 100] \\
   &= [(1)^2 \mod 100] \\
   &= 1
   \end{aligned}$$

   De modo que finalmente los últimos dos dígitos de 7^{35688} son 01.

2. **[$233^{230000022}$ mód 35]**.

   La descomposición prima de 35 es $7 \cdot 5$ y por teorema del resto chino se tiene que primero hay que representar 233 como elemento en $\mathbb{Z}_7^* \times \mathbb{Z}_5^*$, y esto es $(2, 3)$. De este modo,

$$\begin{aligned}[233^{230000022} \mod 35] &\leftrightarrow ([(2)^{230000022} \mod 7], [(3)^{230000022} \mod 5]) \\ &\leftrightarrow (2^{230000022} \mod 7, 3^{230000022} \mod 5)\end{aligned}$$

   Por otro lado se tiene también que 230000022 es divisible por 3 y par, ya que (1) la suma de sus dígitos es 9 y (2) termina en 2. De este modo, si $a = 230000022/3$ y $b = 230000022/2$ (notar que b es impar), se tiene que

$$\begin{aligned}(2^{230000022} \mod 7, 3^{230000022} \mod 5) &= (2^{3 \cdot a} \mod 7, 3^{2 \cdot b} \mod 5) \\ &= (8^a \mod 7, 9^b \mod 5) \\ &= (1^a \mod 7, -1^b \mod 5) \\ &= (1 \mod 7, -1 \mod 5) \\ &= (1, -1) \\ &= (1, 4).\end{aligned}$$

   Ahora, para encontrar el número cuya representación en $\mathbb{Z}_7^* \times \mathbb{Z}_5^*$ es $(1, 4)$ hay que primero encontrar x e y tales que $7x + 5y = 1$. Para esto, utilizando el algoritmo de Euclides extendido se tiene que $x = -2$ e $y = 3$.

   Con esto podemos obtener 1_7 y 1_5, que corresponden a $1_p = [y \cdot Q \mod N]$ y $1_q = [x \cdot P \mod N]$ respectivamente. Entonces

$$\begin{aligned}1_7 &= [3 \cdot 5 \mod 35] \\ &= 15 \\ 1_5 &= [-2 \cdot 7 \mod 35] \\ &= 21.\end{aligned}$$

   Así, podemos usar el último paso del algoritmo visto en clases que dice que el número buscado es $x_p \cdot 1_p + y_q \cdot 1_q \mod N$, en particular

$$\begin{aligned}x_7 \cdot 1_7 + y_5 \cdot 1_5 \mod 35 &= [1 \cdot 15 + 4 \cdot 21 \mod 35] \\ &= [99 \mod 35] \\ &= 29.\end{aligned}$$

   Por lo que el resultado es $[233^{230000022} \mod 35] = 29$.

3. **[46^{51} mód 55]**.

   Por el mismo *hint* de la parte (1) se tiene que $\phi(55) = (5^0 \cdot 4)(11^0 \cdot 10) = 40$. Así, podemos usar que

$$\begin{aligned}[46^{51} \mod 55] &= [46^{51 \mod 40} \mod 55] \\ &= [46^{11} \mod 55]\end{aligned}$$

   Por teorema del resto chino, usando que la descomposición prima de 55 es $5 \cdot 11$ tenemos que al expresar 55 como elemento de $\mathbb{Z}_5^* \times \mathbb{Z}_{11}^*$ se obtiene $(1, 2)$, así

$$\begin{aligned}[46^{11} \mod 55] &\leftrightarrow ([(1)^{11} \mod 5], [(2)^{11} \mod 11]) \\ &\leftrightarrow (1 \mod 5, 2^{11} \mod 11)\end{aligned}$$

   Ahora, $[2^{11} \mod 11]$ se puede expresar como $[(2^5)(2^5)(2) \mod 11]$ y por propiedades de la aritmética modular, eso es quivalente a $[[2^5 \mod 11][2^5 \mod 11][2 \mod 11] \mod 11]$, lo que a su vez, como $[32 \mod 11] = -1$, termina siendo equivalente a $[2 \mod 11]$. Con esto podemos decir finalmente que en $\mathbb{Z}_5^* \times \mathbb{Z}_{11}^*$ se tiene lo siguiente

$$(1 \mod 5, 2^{11} \mod 11) = (1, 2).$$

   Así procediendo análogamente para encontrar a qué numero equivale el $(1, 2)$, debemos encontrar x e y tales que $5x+11y = 1$. Utilizando el algoritmo extendido de Euclides se tiene que $x = -2$ e $y = 1$.

   Con esto podemos obtener 1_5 y 1_{11}, que corresponden a $1_p = [y \cdot Q \mod N]$ y $1_q = [x \cdot P \mod N]$ respectivamente. Por lo tanto

$$\begin{aligned}1_5 &= [1 \cdot 11 \mod 55] \\ &= 11 \\ 1_{11} &= [-2 \cdot 5 \mod 55] \\ &= 45.\end{aligned}$$

   Ahora, usando último paso del algoritmo hay que retornar $x_p \cdot 1_p + y_q \cdot 1_q \mod N$, en particular

$$\begin{aligned}x_5 \cdot 1_5 + y_1 1 \cdot 1_{11} \mod 55 &= [1 \cdot 11 + 2 \cdot 45 \mod 55] \\ &= [101 \mod 55] \\ &= 46.\end{aligned}$$

   Por lo que el resultado es $[46^{51} \mod 55] = 46$.

Problema 2

Sea $\mathbb{G}$ es un grupo cíclico de orden n y generador g.

1. **Demuestre que $\mathbb{Z}_n$ y $\mathbb{G}$ son isomorfos.**

   Para demostrar lo pedido, basta con encontrar una función biyectiva o permutación, entre $\mathbb{Z}_n$ y $\mathbb{G}$, y que esta prerserve estructura.

   De este modo, consideremos que los elementos del grupo se expresan como g^i con $i \in \mathbb{Z}_n$. Es inmediato ver que la cardinalidad de $\mathbb{G}$ es igual a la de $\mathbb{Z}_n$, por ende una biyección es posible. Sea $f : \mathbb{Z}_n \to \mathbb{G}$, tal que

   $$f(x) = g^x$$

   La inversa de esta función es inmediata, abusando de notación para que el argumento de la función inversa sea un elemento de $\mathbb{G}$:

   $$f^{-1}(g^x) = x$$

   Por construcción dicha función es biyectiva. Ahora para fijarse en si preserva o no estructura, se debe cumplir que

   $$\forall g_i, g_j \in \mathbb{G},\ f(g_i \circ_{\mathbb{G}} g_j) = f(g_i) \circ_{\mathbb{Z}_n} f(g_j)$$

   En este caso, se tiene que $\circ_{\mathbb{Z}_n}$, esto es, la operación de $\mathbb{Z}_n$ es la suma módulo n, y la operación del grupo $\mathbb{G}$ se puede considerar en notación multiplicativa. Como al hablar de índices en el grupo, se tiene que g_i corresponde al i-ésimo elemento y este corresponde a su vez al elemento g^i, se debe cumplir que

   $$f(g^i * g^j) = f(g^i) + f(g^j) \mod n$$

   Es inmediato ver que se cumple ya que

   $$\begin{aligned} f(g^i * g^j) &= f(g^{i+j}) \\ &= f(g^{i+j \mod n}) \\ &= i + j \mod n \end{aligned}$$

   Así, f es un isomorfismo entre $\mathbb{Z}_n$ y $\mathbb{G}$, por lo que son isomorfos.

   ■

2. **Asumiendo que $n = p \cdot q$, en donde p y q son primos distintos ¿Cuantos generadores tiene $\mathbb{G}$? Demuestre su resultado.**

   Los grupos cíclicos tienen un generador g_* tal que $\mathbb{G} = \{g_*^0, g_*^1, ..., g_*^{q-1}\}$ donde q es el mínimo entero positivo tal que $g_*^q = 1$.

   Primero demostraré que todos los elementos del grupo g^k donde $\texttt{mcd}(k, n) = 1$, o equivalentemente k y n son primos relativos, generan el grupo. Para esto mostraré que un elemento g^x genera el grupo si y solo si x es coprimo con n (orden del grupo).

   Por propiedad de los coprimos existen a y b tal que $a \cdot x + b \cdot n = 1$, entonces se puede hacer lo siguiente, para todo $y \in \mathbb{Z}$

$$\begin{aligned} g^y &= g^{y \cdot 1} \\ &= g^{y(ax+bn)} \\ &= g^{yax+ybn)} \\ &= g^{yax} \cdot g^{ybn} \\ &= g^{yax}, \text{ ya que } g^{ybn} = 1 \\ &= (g^x)^{ya} \end{aligned}$$

   Por lo que como era para todo y, g^x genera el grupo. Ahora para el si y solo si falta demostrar que es suficiente, es decir la implicancia para el otro lado, esto es, si g^x es generador, genera el grupo.

   Tomemos g^x como el generador del grupo, entonces existe un l tal que $(g^x)^l = g$, ya que g es un elemento del grupo y se puede generar con el generador. Si esto ocurre, entonces en particular se tiene que dar que $x \cdot l = 1 \mod n$, por lo visto en clases, pero es evidente, ya que para que genere a $g = g^1$ se necesita que el producto del exponente en módulo n sea 1. De este modo, por propiedades de la congruencia $\mod n$, se tiene que también existen a y b tal que $a \cdot x + b \cdot n = 1$, que es justamente la definición de coprimos para x y n.

   Una vez probado lo anterior, esto es, g^x genera el grupo si y solo si x es coprimo con n, se tiene que todos los generadores del grupo son de orden coprimo. Esto implica que la cantidad de generadores es igual a la cantidad de elementos coprimos con n en $\{0, ..., n-1\}$ que es justamente $\mathbb{Z}_n^*$, y como sabemos, para saber la cantidad de elementos existe la función $\phi(n)$, que nos dice la cantidad de elementos coprimos con n en $\mathbb{Z}_n$ o bien, la cantidad de elementos en $\mathbb{Z}_n^*$.

   De este modo, para $n = p \cdot q$ se tiene que $\phi(n) = (p-1)(q-1)$.

3. **Generalice el resultado anterior para n entero positivo cualquiera.**

   A parrtir de lo visto en la parte anterior, el problema se reduce a encontrar la cantidad de elementos de $\mathbb{Z}_n^*$. Para esto existe la función ϕ que realiza justamente

eso, sin embargo, explícitamente, se tiene que, para cualquier entero positivo n, siendo este el orden del grupo, la cantidad de elementos corresponde a

$$\phi(n) = \prod_{i=i}^{k} (p_i^{e_i - 1}(p_i - 1)).$$

Siendo cada p un primo de la descomposición prima de n y cada e la cantidad de veces que está presente en la misma.

Problema 3

Construya PPT $\mathcal{A}'$ tal que la probabilidad de adivinar sea 0.99 para todo x.

Primero genero el siguiente algoritmo $\mathcal{A}^*(e, N, z)$ para conseguir lo deseado:

- generar un $y \xleftarrow{\$} \mathbb{Z}_N^*$
- computar $x^* = z \cdot y^e \mod N$.
- obtener $candidate = \mathcal{A}(e, N, x^*)$, de modo que si estoy en ese 0,01 de probabilidad, lo que va a retornar el adversario es $x \cdot y$ ya que con un input de $(xy)^e$, retorna xy en esa fracción de los casos.
- computar $trick = y^{-1} \cdot candidate$, ya que si $candidate$ es xy entonces $trick = x$ ya que y con su inversa se cancelan (y tiene inversa ya que es parte de $\mathbb{Z}_N^*$ y es una propiedad de dicho conjunto).
- si $[trick^e \mod N] = z$ retornar $trick$, en otro caso retornar $null$.

El *truco* está en utilizar la inversa de y para que luego se cancele en el caso en el que ocurra lo deseado.

Ahora, utilizamos el algoritmo $\mathcal{A}^*$ para generar el algoritmo pedido, esto porque necesitamos que tenga éxito en los casos en los que el otro algoritmo falla, esto es, con una probabilidad de 0,99.

Este algoritmo $\mathcal{A}'(e, N, z)$ itera una cantidad fija de veces M que discutiré más adelante, y es así

- para i desde 0 a M
- si $\mathcal{A}^*(e, N, z)! = \texttt{null}$ retornarlo

El análisis probabilístico es el siguiente, para encontrar x al correr el algoritmo original, la probabilidad es 0,01. Ahora, la probabilidad de que se encuentre x en el i-ésimo paso del algoritmo corrresponde a $(0{,}01)(0{,}99)^{i-1}$ ya que tiene que haber caído en el caso contrario i veces. Así, lo que se necesita obtener es un i de modo que la probabilidad sea mayor o igual a 0,99, esto es que se haya encontrado x en i pasos o menos, es decir

$$\sum_{k=1} i(0{,}01)(0{,}99)^{k-1} \geq 0{,}99$$

Esa sumatoria es equivalente a

$$1 - (0{,}99)^i \geq 0{,}99$$

Y despejando i se tiene que $i = 458{,}2$, por lo que para que la probabilidad sea 0,99 para todo x, se requiere que M sea 459, y con esto el algoritmo es PPT, ya que todos los computos son polinomiales ya que aparte de cálculos triviales, se basa en el otro algoritmo que lo es y en computar la inversa de y que viene dado por $\mathbb{Z}_n^*$.

Problema 4

Como un adversario puede escuchar la conversación bajo protocolo Diffie-Hellman y qué más puede hacer.

El protocolo Diffie-Hellman se basa en generar un *secreto común*, esto se logra compartiendo por un lado las características del grupo, que son g, q, $\mathbb{G}$ y además g^α donde α es la llave de Alice. Y por el otro devolviendo g^β donde es análogo a α pero de Bob.

Así, si ambos computan $k = (g^\alpha)^\beta$ y $k = (g^\beta)^\alpha$ respectivamente, tendrán entonces un *secreto común*. El problema, y lo que puede hacer el adversario, es ubicarse en el medio del canal de conversación y entregar a cada miembro de la misma un secreto distinto, de modo que el adversario establece una comunicación bajo Diffie-Hellman con cada uno de los participantes.

Más específicamente, lo que puede hacer es lo siguiente:

- Alice envia hacia Bob $(g, q, \mathbb{G}, g^\alpha)$
- El adversario intercepta, guarda todo, genera una llave nueva γ y se queda con $g^{\alpha\gamma}$.
- El adversario envia a Bob $(g, q, \mathbb{G}, g^\gamma)$
- Bob envía de vuelta hacia Alice (g^β) y guarda $g^{\beta\gamma}$ pensando que es el *secreto común* con Alice
- El adversario intercepta nuevamente, guarda $g^{\beta\gamma}$
- El adversarrio envia a Alice (g^γ)
- Alice guarda $g^{\alpha\gamma}$ pensando que es el secreto común con Bob

Así el adversario tiene un secreto común con Alice: $g^{\alpha\gamma}$, y a su vez uno con Bob: $g^{\beta\gamma}$, sin embargo, Alice y Bob no tienen un secreto común. Esto permite al adversario controlar absolutamente toda la conversación, ya que puede 1) escuchar lo que se mandan y transmitirlo intacto, 2) escuchar lo que se mandan y modificarlo, 3) enviar cosas sin que ninguno de los dos haya mandado algo (posterior al intercambio de llaves). Hago la última distinción porque el caso 2) corresponde a cambiar mensajes, pero la intención de enviarlos existía, en el caso 3) puede enviarlos sin que haya intención de ninguno de los 2 participantes de mandar algo. En el fondo, el adversario impersona a Alice y a Bob cuando estime conveniente.