XXX COMCA
Congreso de Matemática Capricornio
3, 4 y 5 de Agosto de 2022, Iquique, Chile

Aplicaciones de la Criptografía en Ciberseguridad

Dr. Iván F. Jirón Araya
Departamento de Matemáticas
Núcleo de Investigación en Inteligencia Artificial y Data Science
Universidad Católica del Norte
Antofagasta, Chile
ijiron@ucn.cl

Resumen

Dado el aumento exponencial del uso de la Internet como medio de comunicación y colaboración es necesario desarrollar mecanismos de seguridad para proteger la información privada de cada usuario. Ejemplos de esto son el comercio electrónico, transacciones bancarias y protección de datos. Así, en los últimos años la criptografía ha dado respuestas a estas demandas, tanto civiles como militares.
En este curso se presentan los conceptos criptográficos más relevantes, por ejemplo, confidencialidad, integridad de la información, autentificación y privacidad, firmas digitales e intercambio de llaves. Y algunos tópicos matemáticos como por ejemplo, campos de Galois y curvas algebraicas con el fin de alcanzar un adecuado entendimiento de los fundamentos de la criptografía moderna.

1. Introducción

Sin duda que la Internet ha revolucionado nuestro estilo de vida y la importancia de esta herramienta ha crecido en forma exponencial en los últimos veinte años [1] como lo muestra la Figura 1.

Hay mucha información circulando por Internet en correos electrónicos, Tweets, Facebook, vídeos, fotos, comercio electrónico CyberDay, secretos militares, secretos industriales, etc. Pero la Internet es una red pública e insegura, en la cual hay buenos y malos usuarios. Siempre hay noticias sobre sitios web que han sido hackeados, clonación de tarjetas de crédito, adulteración de imágenes, etc. Esto motiva la necesidad de proteger la información con mecanismo que impidan a los usuarios no autorizados (los intrusos o hackers) acceder a ésta. Esta necesidad de protección de la información es satisfecha por la Criptología, que es la ciencia de la Criptografía y el Criptoanálisis. En la Figura 2 se muestra una taxonomía breve.

La Criptografía es el estudio de técnicas matemáticas para ocultar el significado de la información almacenada en un medio o transmitida por un canal, ambos considerados inseguros, como por ejemplo, un disco duro externo, una red wifi e Internet. Entonces, se habla de criptosistemas para ocultar y proteger la información original en un mensaje encriptado.

Por ejemplo, el número y clave de una tarjeta de crédito es información valiosa por 2 ó 3 años, porque caduca o se cierra la tarjeta. Así, si un criptosistema será efectivo si puede proteger el número y clave de la tarjeta de crédito por más de 3 años. Con este ejemplo, se quiere explicar que un criptosistema es seguro si es capaz de proteger la información privada el tiempo suficiente hasta que pierda su valor. Por otra parte, el Criptoanálisis es el estudio de técnicas matemáticas para extraer el significado de la información oculta en un mensaje encriptado. Así, el Criptoanálisis es importante cuando se propone un nuevo criptosistema, para probar su fortaleza a los ataques. Aquí, se habla de quebrar o romper el criptosistema, con el fin de extraer la información original, y dejarla disponible para un usuario no autorizado (el intruso).

En forma intuitiva la esencia de la Criptografía consiste en crear funciones que sean fáciles de evaluar, pero cuya función inversa sea muy difícil de obtener con los medios computacionales existentes.Aquellas funciones que tienen esta característica son clasificadas como One-Way.

El resto de estas notas se organizan en las siguientes secciones:

2. Aritmética Modular
3. Criptografía
4. Criptografía de llave pública o Asimétrica
5. Curvas Elípticas sobre Campos de Galois $\mathbb{Z}_p$ y el Problema del Logaritmo Discreto
6. Conceptos Básicos en Ciberseguridad

Referencias

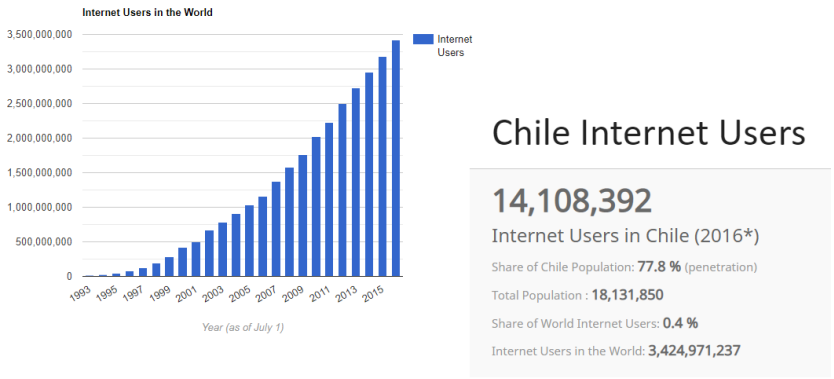

Figura 1: Número de usuarios de Internet 1993-2016 mundial y en Chile [1]

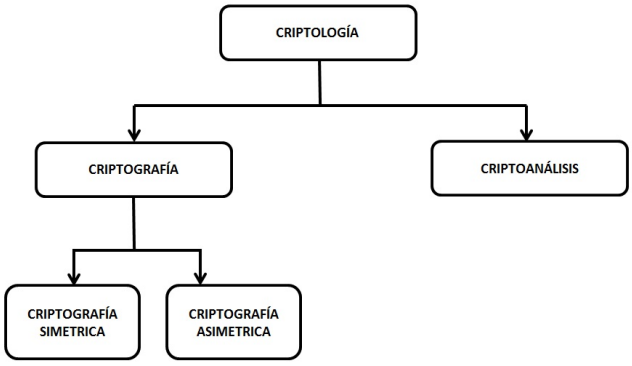

Figura 2: Taxonomía breve de la Criptología

2. Aritmética Modular

En esta sección se explican los conceptos básicos de la aritmética modular, la cual permite desarrollar cálculos aritméticos en conjuntos finitos de números.

Definición: Sean $a, r \in \mathbb{Z}$ y $n \in \mathbb{Z}^+$. Se dice que a es *congruente con* r *modulo* n, $a \equiv r \ mod\,(n)$, si:

$$\exists k, k \in \mathbb{Z} : a = kn + r$$

En esta definición se establece que r es el residuo de la división de a por n.
La relación de congruencia modulo n, $a \equiv r \ mod\,(n)$, es una relación de equivalencia en $\mathbb{Z}$, es decir, establece una partición sobre los números enteros compuesta por n clases de equivalencia (todas son disjuntas entre sí).

Se pueden demostrar las siguientes propiedades:

(a) Si $a \equiv b \ mod\,(n)$ y $c \equiv d \ mod\,(n)$, entonces $a + c \equiv b + d \ mod\,(n)$ y $ac \equiv bd \ mod\,(n)$.

(b) Si $ab \equiv ac \ mod\,(n)$ y $m.c.d.\,(a, n) = 1$, entonces $b \equiv c \ mod\,(n)$.

Las operaciones de suma y multiplicación entre dos clases de equivalencia $[a], [b]$ se calculan como:

$$[a] + [b] = [a + b]$$
$$[a] * [b] = [a * b]$$

Ejemplo: En el grupo $\mathbb{Z}_9$ los cálculos aritméticos se hacen con el modulo $n = 9$.
Para $a = 42$ se tiene que

$$42 = 4 * 9 + 6 \Leftrightarrow 42 \equiv 6 \ mod\,(9)$$

donde $k = 4$ y el resto es $r = 6$.
La clase de equivalencia para $a = 42$, denotada por $[a] = [42]$ está dada por

$$[42] = \{\cdots, -21, -12, -3, 6, 15, 24, 33, 42, \cdots\} = [-21] = [-12] = [-3] = [6] = [15] = [24] = [33]$$

La clase de equivalencia para $a = 17$ está dada por

$$[17] = \{\cdots, -28, -19, -10, -1, 8, 17, 26, 35, \cdots\} = [-28] = [-19] = [-10] = [-1] = [8] = [17] = [26]$$

Así, las $n = 9$ clases de equivalencias para este módulo son:

$$[0], [1], [2], [3], [4], [5], [6], [7], [8]$$

La suma y la multiplicación de clases de equivalencia se calculan como:

$$[8] + [7] = [8 + 7] = [15] = [6] \Leftrightarrow 8 + 7 \equiv 6 \ mod\,(9)$$

Ya que $15 = 9 * 1 + 6 \Leftrightarrow 8 + 7 \equiv 6 \ mod\,(9)$

$$[8] * [7] = [8 * 7] = [56] = [2] \Leftrightarrow 8 * 7 \equiv 2 \ mod\,(9)$$

Ya que $56 = 9 * 6 + 2 \Leftrightarrow 8 * 7 \equiv 2 \ mod\,(9)$

3. Criptografía

La Figura 3 muestra el esquema básico de un criptosistema para un canal inseguro (llamado CANAL, que puede ser un canal de transmisión o un dispositivo de almacenamiento), que está constituido por dos usuarios autorizados (Alice y Bob) y un usuario no autorizado (Eve) que puede escuchar todo lo que se transmite por el CANAL. Para una comunicación segura Alice y Bob deben acordar el uso de un criptosistema, es decir, un método de encriptación ($\boldsymbol{E}_k(\boldsymbol{m}) = \boldsymbol{c}$) para ocultar el mensaje original $\boldsymbol{m}$ en un mensaje encriptado $\boldsymbol{c}$, y un método de desencriptación ($\boldsymbol{D}_k(\boldsymbol{c}) = \boldsymbol{m}$) para recuperar desde $\boldsymbol{c}$ el mensaje original $\boldsymbol{m}$. Estos métodos son llamados algoritmos de encriptación y de desencriptación, respectivamente. Ambos algoritmos usan una llave (o clave) secreta k, que sólo conocen Alice y Bob.
¿Qué pasa con Eve? Bueno ella conoce el criptosistema acordado por Alice y Bob, pero no sabe cuál es la llave secreta k, que ellos comparten. Entonces, los ataques que hace Eve tienen como objetivo descubrir la llave secreta k. Lo más importante es que el algoritmo de encriptación sea fácil de utilizar con la llave secreta k, y por el contario el algoritmo de desencriptación sea muy díficil y casi imposible de utilizar si no se conoce la llave secreta k.
El esquema mostrado en la Figura 3 corresponde a un criptosistema simétrico o de llave privada, esto es, Alice y Bob deben usar la misma llave secreta k en los algoritmos de encriptación y de desencriptación. Lo que plantea el gran problema del intercambio de la llave secreta k entre ellos, por ejemplo, que Alice le entregue la llave k en forma segura a Bob, de manera que Eve no la pueda descubrir. Debido a que todos los canales de transmisión son inseguros, un cartero puede ser sobornado o torturado para que entregue la llave secreta k, una conversación a través de smartphones por puede ser escuchada, un servidor de correo electrónico puede ser atacado, etc.
Algunos objetivos de la criptografía son:

a) La confidencialidad de la información, esto es, que la información está disponible sólo para aquellos usuarios autorizados para conocerla. Por ejemplo, el número de una tarjeta de crédito y su contraseña.

b) La integridad de la información, esto es, prohibir hacer cambios en la información a los usuarios no autorizados, es decir, a los intrusos. Por ejemplo, cambiar la contraseña de la cuenta corriente de un cliente en un banco.

c) La autentificación de un usuario, que consiste en verificar la identidad del usuario que trata de acceder a la información.

d) La autentificación de un mensaje se consigue verificando la identidad de la fuente del mensaje. Por ejemplo, verificar la legitimidad de un correo electrónico, para evitar ser víctima de un Phishing.

e) La firma digital es un medio para identificar y diferenciar a un usuario en particular del resto de los usuarios.

f) La no-repudiación es un mecanismo para que los usuarios no puedan negar o desistir de sus acciones. Por ejemplo, negar la compra de un bien a través de un sitio web, después de haberlo recibido en su hogar.

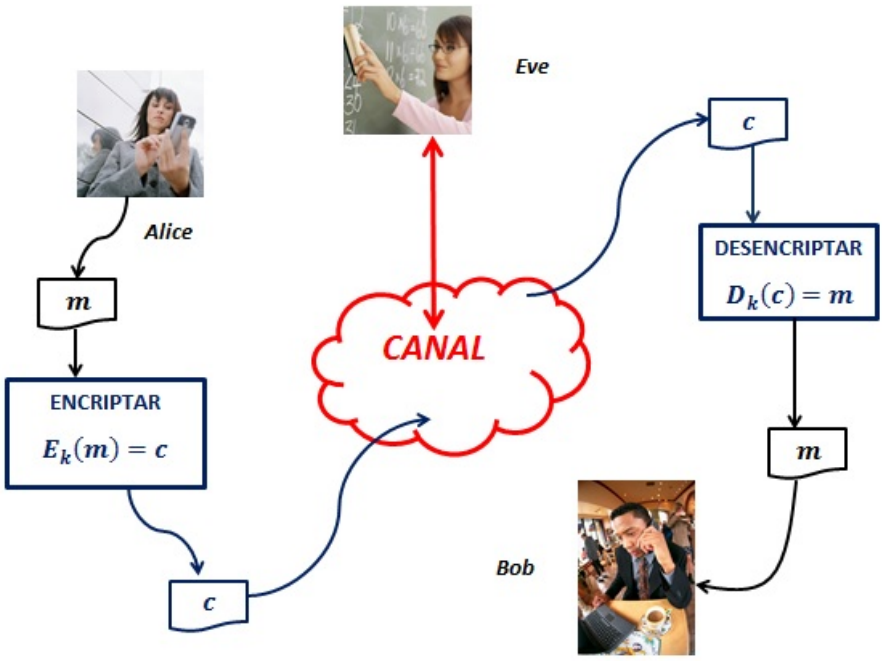

Figura 3: Esquema de un Criptosistema

4. Criptografía de llave pública o Asimétrica

La Criptografía de llave pública o asimétrica nace con el trabajo de Diffie y Hellman en 1976 [6]. Y a diferencia de la criptografía de llave privada (o simétrica) para los usuarios autorizados, Alice y Bob, cada uno tiene un par de llaves. De estas llaves una es secreta y la otra es pública, como se muestra en la figura 4. Alice y Bob usan sus llaves públicas y privadas para calcular una tercera llave, que es común para ambos y que no se transmite por ningún canal.
En esta sección se presentan dos soluciones al gran problema del intercambio de la llave secreta entre los usuarios autorizados, debido a que todos los canales de transmisión son inseguros. Estos se clasifican en dos tipos: el problema del logaritmo discreto sobre un campo finito y la factorización de números gigantes en sus factores primos.

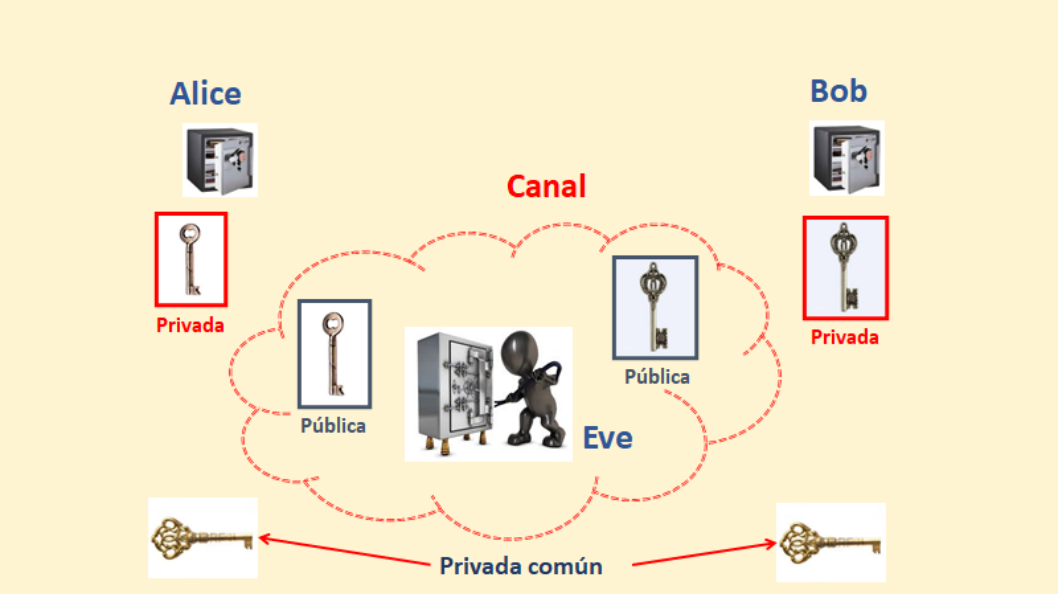

Figura 4: Esquema de la Criptografía Asimétrica

4.1. Algoritmo de Diffie-Hellman

La solución a este problema propuesta por Diffie y Hellman en 1976 [6] consiste en un esquema de intercambio de llaves, en el cual Alice y Bob siguen los siguientes pasos:

1. Elijen un número primo p muy grande, y a continuación utilizan los elementos del campo $\mathbb{Z}_p^\star$.
2. Elijen un entero $\alpha \in \{2, 3, \ldots, p-2\}$
3. Publican p y α.
4. Alice selecciona su llave privada como un entero aleatorio y secreto $k_a < p$, y calcula su llave pública como
$$A = \alpha^{k_a} mod\,(p)$$
y la transmite a Bob.
Por su parte, Bob selecciona su llave privada como un entero aleatorio y secreto $k_b < p$, y cálcula su llave pública como
$$B = \alpha^{k_b} mod\,(p)$$
y la transmite a Alice.
5. Alice calcula
$$(B)^{k_a} = \left(\alpha^{k_b}\right)^{k_a} mod\,(p) = \alpha^{k_b k_a} mod\,(p)$$
mientras que Bob calcula
$$(A)^{k_b} = \left(\alpha^{k_a}\right)^{k_b} mod\,(p) = \alpha^{k_a k_b} mod\,(p)$$
6. La llave secreta que comparten ambos es $\boldsymbol{k} = (B)^{k_a} = (A)^{k_b} = \alpha^{k_a k_b} mod\,(p)$, y no fue transmitida por ningún canal.

La seguridad de este esquema se basa en la dificultad de resolver el problema del logaritmo discreto (discrete logarithm problem, DLP), en campos finitos, de las llaves públicas $A = \alpha^{k_a} mod\,(p)$ y $B = \alpha^{k_b} mod\,(p)$. Esto es, determinar el valor exacto de las llaves secretas k_a y k_b.

Ejemplo: Alice y Bob eligen $p = 13$, $\alpha = 6 \in \{2, 3, \ldots, 11\}$.
Alice selecciona su llave privada $k_a = 5 < 13$, y calcula su llave pública como

$$A = 6^5 \ mod\,(13) = 7{,}776 \ mod\,(13) = 2$$

Por su parte, Bob selecciona su llave privada $k_b = 8 < 13$, y cálcula su llave pública como

$$B = 6^8 \ mod\,(13) = 1{,}679{,}616 \ mod\,(13) = 3$$

Alice calcula

$$(3)^5 = \left(6^8\right)^5 mod\,(13) = 9$$

mientras que Bob calcula

$$(2)^8 = \left(6^5\right)^8 mod\,(13) = 9$$

Entonces, Alice y Bob comparten la llave secreta común $\boldsymbol{k} = 9$.

4.2. Algoritmo RSA

Otra solución para el problema del intercambio de llaves es algoritmo criptográfico asimétrico RSA (por Rivest, Shamir y Adleman) publicado en 1978 [7], que en la actualidad es ampliamente usado. La fortaleza del algoritmo RSA está en la dificultad de factorizar el producto de dos números primos gigantes.
Los algoritmos de encriptación y de desencriptación son realiazados sobre un $\mathbb{Z}_n = \{0, 1, 2, \ldots, n-1\}$ y aritmética modular.

Definición: Sea $n \in \mathbb{N}$ la función Φ de Euler se define como la cantidad de números $m \in \mathbb{N}$, tales que $m < n$ y $\gcd(m, n) = 1$.

Ejemplo: Si $p \in \mathbb{N}$ es un número primo, entonces $\Phi(p) = p - 1$. Para $p = 7$ sus primos realitivos son $1, 2, 3, 4, 5$ y 6, esto es, $\Phi(7) = 6$.

El mensaje original $\boldsymbol{m}$ se representa como un elemento de $\mathbb{Z}_n$, que debe ser menor que n.
Generación de llaves: Para generar sus llaves pública y secreta Bob sigue los siguientes pasos:

1. Elige dos números primos distintos p y q extremadamente grandes (de 2048 bits o de 200 dígitos a lo menos) y de igual longitud para garantizar la seguridad del sistema. A continuación calcula

   $$n = pq$$

   Atención: Los dos números primos p y q deben permanecer secretos.

2. Calcula

   $$\Phi(n) = (p-1)(q-1)$$

3. Selecciona su llave pública (exponente) aleatoria $e \in \{1, 2, \ldots, \Phi(n) - 1\}$ tal que $\gcd(e, \Phi(n)) = 1$.

4. Calcula su llave privada (exponente) d tal que

   $$d \cdot e \equiv 1 mod\,(\Phi(n)) \Longleftrightarrow d \equiv e^{-1} mod\,(\Phi(n))$$

   y $\gcd(d, n) = 1$.

Algoritmo de Encriptación:
Bob publica n y e.
Alice encripta un mensaje $\boldsymbol{m}$ para Bob de la siguiente forma:

$$\boldsymbol{c} = \boldsymbol{E}_e(\boldsymbol{m}) \equiv \boldsymbol{m}^e mod(n)$$

donde $\boldsymbol{m}, \boldsymbol{c} \in \mathbb{Z}_n$.
Algoritmo de Desencriptación:
Bob recibe el mensaje encriptado $\boldsymbol{c}$ y lo desencripta de la siguiente forma:

$$\boldsymbol{m} = \boldsymbol{D}_d(\boldsymbol{c}) \equiv \boldsymbol{c}^d mod(n) = (\boldsymbol{m}^e)^d \, mod(n) = \left(\boldsymbol{m}^{ed}\right) mod(n)$$

donde $\boldsymbol{m}, \boldsymbol{c} \in \mathbb{Z}_n$.

Ejemplo: Para $p = 47$ y $q = 71$, $n = 3337$. La llave pública e no debe tener factores en común con $\Phi(3337) = (46)(70) = 3220$. Así, Bob puede elegir $e = 79$. Entonces, calcula su llave privada como $d \equiv 79^{-1} mod(3220) = 1019$.
Si Alice, quiere enviarle a Bob el mensaje $\boldsymbol{m} = 688$ lo encripta usando la llave pública de Bob $e = 79$ y $n = 3337$, y calcula

$$\boldsymbol{c} = \boldsymbol{E}_{79}(688) \equiv 688^{79} mod(3337) = 1570$$

A continuación, Bob recive el mensaje encriptado $\boldsymbol{c} = 1570$, y lo desencripta calculando

$$\boldsymbol{D}_{1019}(\boldsymbol{c}) \equiv 1570^{1019} mod(n) = \left(688^{79}\right)^{1019} mod(3337) = 688$$

El producto de las llaves pública $e = 79$ y privada $d = 1019$ de Bob se calcula $mod(3220)$

$$79 * 1919 = 80501 = 3220 * 25 + 1 \Leftrightarrow 80501 \equiv 1 \ mod(3220)$$

4.3. Firma digital con el Algoritmo RSA

Para implementar el uso de una firma digital con el Algoritmo RSA, Alice y Bob siguen los siguientes pasos:

a) Bob publica n y e.

b) Bob construye su firma digital usando el mensaje $\boldsymbol{m}$ de la siguiente forma:

$$s_b \equiv \boldsymbol{m}^d mod(n)$$

donde d es su llave privada.

c) Le transmite a Alice el par ($\boldsymbol{m}$,s_b).

d) Alice calcula

$$x \equiv (s_b)^e \, mod(n)$$

Entonces,

$$x \equiv \boldsymbol{m} \ mod(n) \Longrightarrow \text{ la firma } s_b \text{ es válida.}$$
$$x \not\equiv \boldsymbol{m} \ mod(n) \Longrightarrow \text{ la firma } s_b \text{ NO es válida.}$$

Así, Bob usa su llave privada d para firmar y Alice usa la llave pública e de Bob para verificar la autenticidad de la firma.

4.4. Algoritmo de ElGamal

En 1985 T. ElGamal [8] usó el esquema de Diffie-Hellman para encriptar y desencriptar un mensaje. La seguridad de este criptosistema está basada en la imposibilidad de resolver el problema del logaritmo discreto.
El algoritmo de ElGamal se utiliza sobre un grupo $\mathbb{Z}_p^*$, con p un número primo gigante, que es público.
En este algoritmo, Bob usa como llave privada (secreta) x, con $x < p$, y como llave pública (β, y) donde $\beta < p$ e

$$y \equiv \beta^x \ mod\,(p)$$

Alice encripta un mensaje $\boldsymbol{m}$ eligiendo en forma aleatoria k, que es un primo relativo con $p-1$. A continuación calcula

$$\boldsymbol{a} = \beta^k mod\,(p)$$

$$\boldsymbol{b} = y^k \boldsymbol{m} mod\,(p)$$

Entonces, le transmite a Bob el mensaje encriptado en el par $(\boldsymbol{a}, \boldsymbol{b})$. Para desencriptar Bob hace los siguientes cálculos:

$$\left(\frac{\boldsymbol{b}}{\boldsymbol{a}^x}\right) mod\,(p) = \left(\frac{y^k \boldsymbol{m}}{(\beta^k)^x}\right) mod\,(p) = \left(\frac{(\beta^x)^k \boldsymbol{m}}{(\beta^k)^x}\right) mod\,(p) = \boldsymbol{m}$$

Ejemplo: Bob elige $p = 13$, $\beta = 6$, $x = 5$ y calcula $y \equiv 6^5 mod\,(13) = 7776 mod\,(13) = 2$. Y a continuación, publica $p = 13$, $\beta = 6$ e $y = 2$.
Alice encripta el mensaje $\boldsymbol{m} = 10$ eligiendo en forma aleatoria $k = 4$. Entonces, determina los valores de

$$\boldsymbol{a} = 6^4 mod\,(13) = 1296 mod\,(13) = 9$$

$$\boldsymbol{b} = 2^4 \mathbf{10} mod\,(13) = 160 mod\,(13) = 4$$

Entonces, le transmite a Bob el mensaje encriptado en el par $\boldsymbol{c} = (\boldsymbol{a}, \boldsymbol{b}) = (9, 4)$.
Bob desencripta el mensaje encriptado $\boldsymbol{c} = (\boldsymbol{a}, \boldsymbol{b}) = (9, 4)$ de la siguiente forma:

$$\left(\frac{\mathbf{4}}{\mathbf{9}^5}\right) mod\,(13) = \left(\frac{(2)^4 \mathbf{10}}{(6^4)^5}\right) mod\,(13) = \left(\frac{(6^5)^4 \mathbf{10}}{(6^4)^5}\right) mod\,(13) = \mathbf{10} = \boldsymbol{m}$$

4.5. Firma digital usando el Algoritmo de ElGamal

Bob selecciona un número primo p, y dos números aleatorios β y x, tales que $\beta < p$ y $x < p$. Y calcula

$$y \equiv \beta^x mod\,(p)$$

Bob publica y, β y p.
Para firmar un mensaje $\boldsymbol{m}$, Bob elige un número aleatorio secreto k que es un primo relativo con $p-1$, y calcula $\boldsymbol{a} = \beta^k mod\,(p)$.
A continuación, usa el algoritmo extendido de Euclides para encontrar el valor de $\boldsymbol{b}$ en la ecuación:

$$\boldsymbol{m} = (x\boldsymbol{a} + k\boldsymbol{b})\, mod\,(p-1)$$

El par $(\boldsymbol{a}, \boldsymbol{b})$ es la firma digital de Bob.
Alice verifica la firma de Bob calculando:

$$y^{\boldsymbol{a}} \boldsymbol{a}^b mod\,(p) = (\beta^x)^{\boldsymbol{a}} \left(\beta^k\right)^b mod\,(p) = \beta^{(x\boldsymbol{a}+k\boldsymbol{b})} mod\,(p) = \beta^{\boldsymbol{m}} mod\,(p)$$

Ejemplo: Bob elige $p = 11$, $\beta = 2$, $x = 8$ y calcula $y \equiv 2^8 mod\,(11) = 256 mod\,(11) = 3$.
Bob publica $y = 3$, $\beta = 2$ y $p = 11$, y mantiene secreto $x = 8$. Para firmar el mensaje $\boldsymbol{m} = 5$, selecciona un número aleatorio secreto $k = 9$ que es un primo relativo con $p - 1 = 10$.
Calcula $\boldsymbol{a} = \beta^k mod\,(p) = \boldsymbol{a} = 2^9 mod\,(11) = 512 mod\,(11) = 6$. A continuación, usa el algoritmo extendido de Euclides para encotrar el valor de $\boldsymbol{b}$ en la ecuación:

$$\mathbf{5} = (8 \cdot \mathbf{6} + 9 \cdot \boldsymbol{b})\, mod\,(10) = (8 + 9 \cdot \boldsymbol{b})\, mod\,(10)$$

resultando $\boldsymbol{b} = 3$. Y el par $(\boldsymbol{a}, \boldsymbol{b}) = (6, 3)$ es la firma digital de Bob.

5. Curvas Elípticas sobre Campos de Galois $\mathbb{Z}_p$ y el Problema del Logaritmo Discreto

En esta sección se muestra como se hace el intercambio de llaves con el esquema de Diffie-Hellman usando curvas élipticas sobre campos de Galois $\mathbb{Z}_p$.

Definición: Una curva elíptica sobre $\mathbb{Z}_p$ es el conjunto de puntos $(x, y) \in \mathbb{Z}_p$ que satisfacen la ecuación

$$y^2 = \left(x^3 + ax + b\right) mod\,(p)$$

junto con un punto en el infinito $\widetilde{O}$, donde $a, b \in \mathbb{Z}_p$ y se cumple la condición

$$4a^3 + 27b^2 \neq 0 \ mod\,(p)$$

Ejemplo: Si la curva elíptica

$$y^2 = x^3 - 3x + 3$$

la definimos sobre $\mathbb{R} \times \mathbb{R}$ su gráfica sería como la mostrada en la figura 5.

Como se puede observa en la figura 5 la curva no se intersecta a sí misma y no tiene vértices, es decir, es una curva no-singular. Esta característica se obtiene cuando en la definición de una curva elíptica se tiene que el discriminante de la curva

$$-16\left(4a^3 + 27b^2\right) \neq 0$$

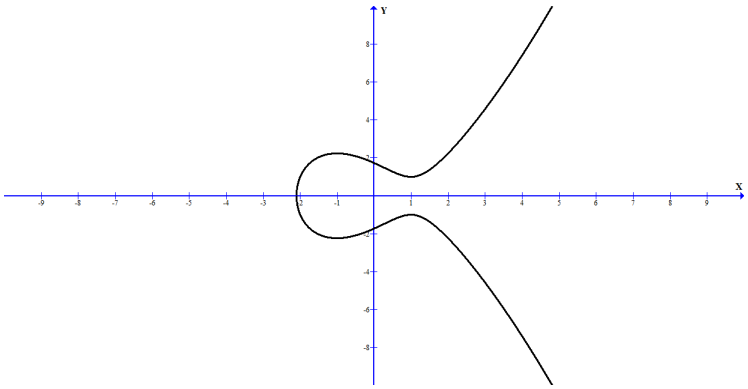

Figura 5: Curva Elíptica sobre $\mathbb{R}$

5.1. Operaciones sobre curvas elípticas

Dados dos puntos sobre una curva elíptica $P(x_1, y_1)$ y $Q(x_2, y_2)$ la suma de éstos genera un tercer punto $R(x_3, y_3)$ sobre la misma curva, tal que,

$$P + Q = (x_1, y_1) + (x_2, y_2) = (x_3, y_3) = R$$

Para una curva elíptica definida sobre $\mathbb{R}$ la adición $P+Q = R$ se puede representar geométricamente como lo muestra la figura 6.
Por otra parte, a la operación de sumar un punto $P(x_1, y_1)$ consigo mismo se llama doblaje y corresponde a

$$P + P = 2P$$

En forma similar a la suma $P + Q = R$, el doblaje $2P$ se puede representar geométricamente como lo muestra la figura 7.
Por otra parte, el inverso aditivo de un punto $P(x_1, y_1)$ se define como $-P(x_1, -y_1)$, y es tal que

$$P + (-P) = \widetilde{O}$$

y se puede representar geométricamente como lo muestra la figura 8.

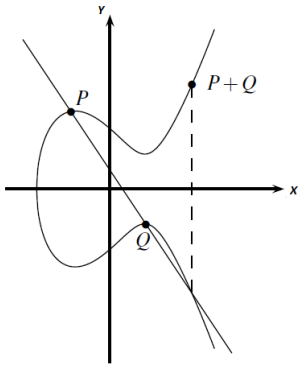

Figura 6: Suma de puntos de una curva Elíptica sobre $\mathbb{R}$

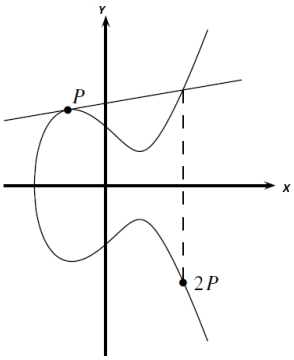

Figura 7: Doblaje de puntos de una curva Elíptica sobre $\mathbb{R}$

Definición: Para dos puntos $P(x_1, y_1)$ y $Q(x_2, y_2)$ sobre una curva elíptica definidad sobre $\mathbb{Z}_p$ la adición y el doblaje se calculan como

$$x_3 = s^2 - x_1 - x_2 \ mod\,(p)$$

$$y_3 = s\,(x_1 - x_3) - y_1 \ mod\,(p)$$

donde

$$s = \begin{cases} \frac{y_2 - y_1}{x_2 - x_1} mod\,(p) & \text{si } P \neq Q \\ \frac{3x_1^2 + a}{2y_1} mod\,(p) & \text{si } P = Q \end{cases}$$

Ejemplo: Para la curva

$$C : y^2 = x^3 + 2x + 2 \ mod\,(17)$$

definida sobre el campo $\mathbb{Z}_{17}$ y el punto $P(5, 1)$:
Se verifica que $P(5, 1)$ está en la curva, ya que reemplazando $x = 5$ e $y = 1$ en la ecuación se tiene que:

$$y^2 = 1 \ , \ x^3 + 2x + 2 \ mod\,(17) = 125 + 10 + 2 \ mod\,(17) = 137 \ mod\,(17) = 1$$

A continuación, se calcula el doblaje de $P(5, 1)$:

$$s = \frac{3x_1^2 + a}{2y_1} mod\,(p) = \frac{3 \cdot 5^2 + 2}{2 \cdot 1} mod\,(17)$$

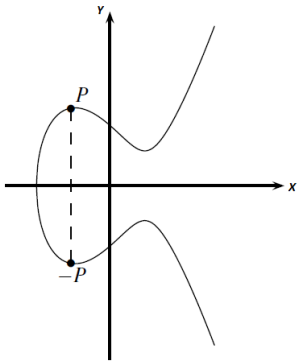

Figura 8: Inverso aditivo de un punto sobre una curva Elíptica sobre $\mathbb{R}$

$$s = \frac{3x_1^2 + a}{2y_1} mod\,(p) = \frac{3 \cdot 5^2 + 2}{2 \cdot 1} mod\,(17)$$

$$s = (2 \cdot 1)^{-1} (3 \cdot 25 + 2) \ \ mod\,(17) = 2^{-1} \cdot 9 = 9 \cdot 9 = 81 \ mod\,(17) = 13$$

$$x_3 = s^2 - x_1 - x_2 = 13^2 - 5 - 5 = 159 = 6 \ mod\,(17)$$
$$y_3 = s\,(x_1 - x_3) - y_1 = 13\,(5 - 6) - 1 = -14 = 3 \ mod\,(17)$$

Así,

$$Q = 2P = (6, 3)$$

y que también es un punto sobre la curva.
Ahora, se calcula la adición $P + Q$:

$$s = \frac{y_2 - y_1}{x_2 - x_1} mod\,(p) = \frac{3 - 1}{6 - 5} mod\,(17) = 2$$

$$x_3 = s^2 - x_1 - x_2 = 2^2 - 5 - 6 = -11 = 10 \ mod\,(17)$$
$$y_3 = s\,(x_1 - x_3) - y_1 = 2\,(5 - 10) - 1 = -11 = 6 \ mod\,(17)$$

Así,

$$P + Q = (10, 6)$$

y que también es un punto sobre la curva.

El siguiente teorema es muy importante para el intercambio de llaves de Diffie-Hellman usando curvas elípticas.

Teorema: Los puntos de una curva elíptica junto con $\widetilde{O}$ forman un grupo cíclico.

En particular, debe existir un elemento primitivo que genera a todos los elementos del grupo como potencias.

Ejemplo: Para la curva

$$C : y^2 = x^3 + 2x + 2 \ mod\,(17)$$

definida sobre el campo $\mathbb{Z}_{17}$ tiene $\#\,(C) = 19$ puntos, esto es, el grupo es de orden primo. Así, todos los puntos son elementos primitivos del grupo. Usando como generador al punto

$P = (5, 1)$ se obtienen los siguientes puntos:

$$\begin{array}{cc} P = (5,1) & 11P = (13,10) \\ 2P = (6,3) & 12P = (0,11) \\ 3P = (10,6) & 13P = (16,4) \\ 4P = (3,1) & 14P = (9,1) \\ 5P = (9,16) & 15P = (3,16) \\ 6P = (16,13) & 16P = (10,11) \\ 7P = (0,6) & 17P = (6,14) \\ 8P = (13,7) & 18P = (5,16) \\ 9P = (7,6) & 19P = \widetilde{O} \\ 10P = (7,11) & 20P = 19P + P = \widetilde{O} + P = P \\ & 21P = 20P + P = P + P = 2P \end{array}$$

5.2. El Problema del Logaritmo Discreto sobre Curvas Elípticas

El Problema del Logaritmo Discreto sobre una curva elíptica C (Elliptic Curved Discrete Logarithm Problem, ECDLP) consiste en que dados dos puntos P y Q conocidos de la curva, se intenta encontrar un entero d tal que $1 \leq d \leq \#(C)$ y

$$Q = \underbrace{P + P + \cdots + P}_{d - veces} = dP$$

Así, Bob tendría como llave pública al punto Q y como llave privada el entero d.
La notación usada en el ECDLP corresponde a una multiplicación, que en realidad consiste en sumar un punto con sigo mismo varias veces.

Ejemplo: Para la curva

$$C : y^2 = x^3 + 2x + 2 \ mod\,(17)$$

definida sobre el campo $\mathbb{Z}_{17}$ con $\#(C) = 19$ para los puntos $P(5,1)$ y $Q(16,4)$ se tiene que

$$Q = 13P$$

Por lo tanto, dado que Bob publica el punto $Q(16,4)$ como su llave pública, la atacante Eve intentará calcular el entero $d = 13$ a partir del punto $P(5,1)$ que también es público.

5.3. Intercambio de llaves de Diffie-Hellman sobre Curvas Elípticas

A continuación se explica el Intercambio de llaves de Diffie-Hellman sobre Curvas Elípticas (elliptic curve DiffieHellman key exchange, ECDH).
En primer lugar, se deben definir los parámetros del ECDH. Estos son:
1. Un número primo p.
2. Una curva elíptica $C : y^2 = x^3 + ax + b \ mod\,(p)$
3. Un elemento primitivo P, es decir, un punto generador para el grupo.
Ahora se detallan todos los pasos del ECDH.
1. Alice y Bob eligen y publican un punto P generador para el grupo.
2. Alice y Bob eligen sus llaves secretas y aleatorias a y b, respectivamente.

$$a, b \in \{2, 3, \cdots, \#(C) - 1\}$$

3. Alice y Bob, por separado, calculan sus respectivas llaves públicas A y B como

$$A = aP \ , \ B = bP$$

4. Alice y Bob publican A y B, respectivamente.
5. Alice y Bob, por separado, calculan la llave secreta común, de la siguiente forma:
Bob calcula usando la llave pública de Alice

$$K = bA = b\,(aP) = baP$$

Alice calcula usando la llave pública de Bob

$$K = aB = a\,(bP) = abP$$

dado que la multiplicación es conmutativa ambos obtienen el mismo resultado.
Como ya se explicó anterirmente, esta llave común no es transmitida por ningún canal. Y la seguridad de este protocolo está basada en que Eve, la atacante, debe resolver los dos problemas del logaritmo discreto

$$A = aP\ ,\ B = bP$$

para calcular la llave común K, esto es, debe encontrar los valores de a y b.
La llave común K ahora se puede usar en un algoritmo de encriptación/desencriptación, como por ejemplo el Algoritmo AES (Advanced Encryption Standard) creado en Bélgica y adoptado por Estados Unidos de 2001.

Ejemplo: Siguiendo los pasos discutidos se eligen:
1. El número primo $p = 17$, es decir, el campo $\mathbb{Z}_{17}$.
2. La curva elíptica $C : y^2 = x^3 + 2x + 2\ mod\,(17)$, con $\#\,(C) = 19$
3. El elemento primitivo $P\,(5, 1)$, es decir, un punto generador para el grupo.
Ahora se detallan todos los pasos del ECDH.
1. Alice y Bob eligen y publican el punto $P\,(5, 1)$.
2. Alice y Bob eligieron sus llaves secretas y aleatorias $a = 3$ y $b = 10$, respectivamente.

$$a, b \in \{2, 3, \cdots, 18\}$$

3. Alice y Bob, por separado, calculan sus respectivas llaves públicas A y B como

$$A = 3P = (10, 6)\ \ ,\ \ B = 10P = (7, 11)$$

4. Alice y Bob publican $A = (10, 6)$ y $B = (7, 11)$, respectivamente.
5. Alice y Bob, por separado, calculan la llave secreta común, de la siguiente forma:
Bob calcula usando la llave pública de Alice

$$K = 10A = 10\,(3P) = 10 \cdot 3P$$

Alice calcula usando la llave pública de Bob

$$K = 3B = 3\,(10P) = 3 \cdot 10P$$

Para aplicaciones prácticas

6. Conceptos Básicos en Ciberseguridad

Lo primero es plantear la siguiente pregunta:

6.1. ¿Qué es la ciberseguridad?

La ciberseguridad abarca todas las medidas y prácticas para proteger las redes de sistemas informáticos, dispositivos y datos frente a ataques y accesos no autorizados.
En la figura 9 se describe la taxonomía de la ciberseguridad, en la que se consideran las siguientes áreas:

1. Seguridad de redes: considera todas las acciones y políticas para proteger una red informática de los ataques dirigidos u oportunista de los intrusos.

2. Seguridad de aplicaciones: se centra en mantener el software y los dispositivos libres de amenazas, ya que una aplicación comprometida podría proporcionar acceso a los datos, que supuestamente va a proteger.

3. Seguridad operacional: incluye los procesos y decisiones para manejar y proteger los activos de datos (permisos de los usuarios, cómo y dónde se pueden almacenar o compartir los datos).

4. Recuperación ante desastres y continuidad del negocio: cómo responde una organización a un ataque que provoque la pérdida de funcionamiento o datos, es decir, cómo la organización restaura sus operaciones e información para volver a operar normalmente como antes del ataque.

5. Seguridad de la información: se enfoca en cómo se protege la integridad y privacidad de los datos, tanto en su almacenamiento como en su transmisión a través de algún canal.

6. Educación del usuario final: se enfoca en las personas (el factor de la ciberseguridad más impredecible) y cómo se educan, por ejemplo, para evitar introducir accidentalmente un virus en el sistema o a eliminar archivos adjuntos de correo electrónico sospechosos.

La ciberseguridad no es solo una medida puntual en el tiempo. Es un proceso continuo de concientización sobre seguridad, planificación estratégica, implementación, monitoreo y evaluación.

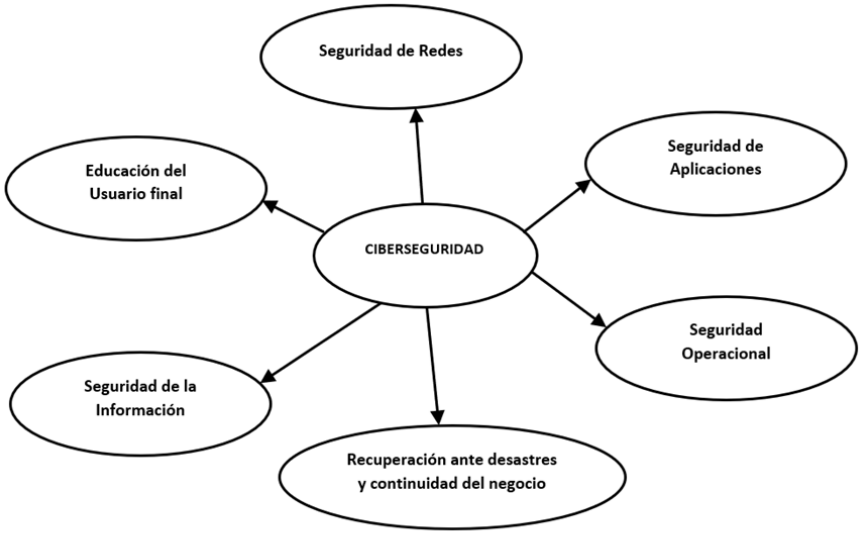

Figura 9: Taxonomía de la Ciberseguridad

6.2. ¿Qué es un ciberataque?

Un ciberataque consiste en acceder ilegalmente a las plataformas informáticas (servidores, computadores, redes o software) de una organización y controlarla para robar información, dañar los datos e interrumpir el funcionamiento de la organización.
La figura 10 muestra una taxonomía sobre los ciberataques más recurrentes, que incluye:

1. Malware: son los software maliciosos, que un hacker crea para interrumpir o dañar las plataformas informáticas de una organización. Por ejemplo, virus, troyanos, spyware, ransomware y botnets.

2. SQL inyección: son ataques para obtener el control de una base de datos, a través de la inyección de códigos SQL maliciosos, que le permiten al hacker obtener acceso a la información almacenada en la base de datos atacada.

3. Phishing: un hacker envía a las víctimas correos electrónicos que parecen ser de una empresa legítima para pedir información confidencial, como por ejemplo, datos de las tarjetas de crédito.

4. Denial-of-Service (DoS): uno hacker o varios atacan un sistema informático para evitar que atienda todas las solicitudes legítimas, sobrecargando las redes y los servidores con tráfico.

5. Man-in-the-Middle (MiM): un hacker intercepta la comunicación entre dos usuarios para robar datos. Por ejemplo, en una red WiFi pública en un aeropuerto, un hacker podría interceptar los datos que se transmiten desde el smartphone de la víctima y la red.

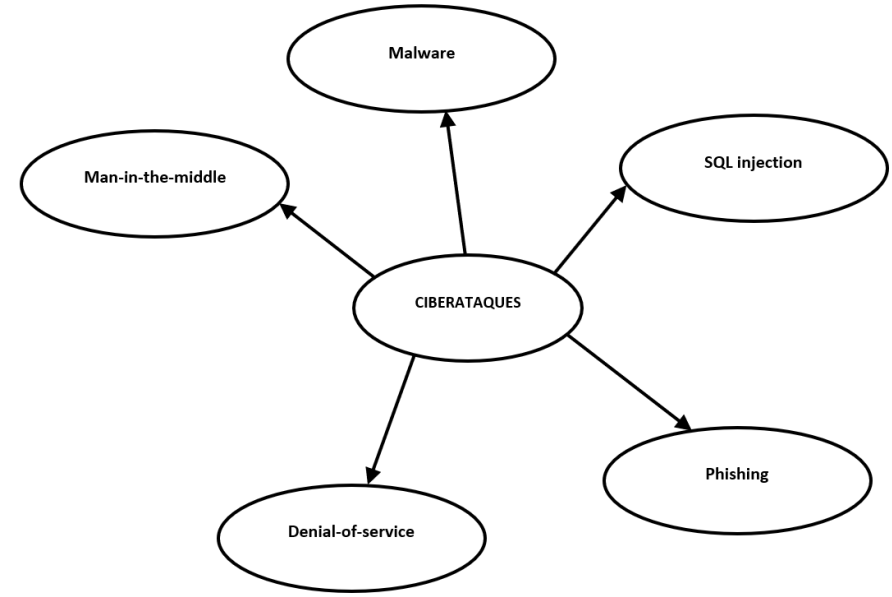

Figura 10: Taxonomía de Ciberataques

6.3. Herramientas básicas y algunas recomendaciones para la Ciberdefensa

Las herramientas básicas de protección contra ciberataques son mantener actualizados:

1. Firewall
2. Antivirus
3. Antispyeware y Antispam
4. Sistemas operativos
5. Navegadores y motores de búsqueda

Algunas recomendaciones son:

1. El antivirus se inicie automáticamente con el inicio de su computador.
2. El antivirus siempre está .ªctivado"mientras la computador está encendido.
3. Descargar e instalar siempre las últimas actualizaciones de los proveedores para actualizar la base de datos de definiciones de virus.
4. Ejecutar regularmente escaneos rápidos en su computador.
5. Nunca abra correos electrónicos de usuarios desconocidos.
6. Utilice grupos de usuarios independientes basados en contraseñas seguras.
7. No guarde las contraseñas en los navegadores.
8. Educar continuamente a las personas sobre ciberseguridad.
9. Entrenar para las posibles formas de ingeniería social que explotan la inocencia de las personas

6.4. Ciberataque a un banco

El ransomware (secuestro) es un software malicioso que le impide al propietario acceder a sus archivos importantes o a su dispositivo, y lo chantajea para que pague un rescate a fin de impedir que los datos se eliminen o se utilicen de manera inadecuada.
En 2020 el Banco Estado de Chile sufrió de un ciberataque del tipo ransomware. Este ciberataque se habría debido a que durante el periodo de teletrabajo fuera de la red segura del banco, un trabajador fue víctima de un phishing que infectó su computador, y luego conectado a la red del banco permitió que ransomware entrara al sistema. La solución de emergencia fue apagar los servidores para detener la propagación del virus, y desconectarse de Internet para detener el robo de datos y dinero. De acuerdo a lo publicado en la prensa, el ransomware era Sodinokibi, el cual está formado por una combinación de:

1. Advanced Encryption Standard (AES 2001): En 1997 el NIST del gobierno de Estados Unidos llamó a un concurso para definir un nuevo estándar de encriptación avanzado (Advanced Encryption Standard, AES). Este concurso fue ganado en 2001 por el algoritmo Rijndael, que encripta bloques de bytes, desarrollado por los criptógrafos belgas Joan Daemen y Vincent Rijmen. En la figura 11 se muestra una ronda del algoritmo AES, donde los bloques de entrada $A_i, i = 0, 1, \cdots, 15$ son de 16 bits cada uno. A continuación, cada uno entra en una caja S y son transformados en los bloques $B_i, i = 0, 1, \cdots, 15$; cada uno con 16 bits. Estos bloques B_i son permutados en la capa "ShiftRowsz mezclados en la capa "MixColumn"produciendo los bloques $C_i, i = 0, 1, \cdots, 15$. En el último paso de la ronda, se hace un xor entre la subllave k_i. Así, AES encripta bytes por bytes.

2. Algoritmo Salsa20: es un algoritmo de encriptación por flujo diseñado en 2005 por Daniel J. Bernstein, profesor de la Universidad de Illinois en Chicago, Estados Unidos de América. Consiste en una secuencia de tres operaciones simples con palabras de 32 bits. En la primera operación se calcula la suma $a + b \ mod \left(2^{32}\right)$, donde a y b son palabras de 32 bits. En la segunda se calcula el o-exclusivo (exclusive-or, xor), $a \bigoplus b$, donde a y b son palabras de 32 bits. En la tercera operación se hace una rotación constante de 32 bits a la izquierda de n bits, $a <<< n$, donde a es una palabra de 32 bits. Combinando estas operaciones Salsa20 construye una función hash, que comienza con 16 palabras (la clave de 32 bytes, el número único de un mensaje de 8 bytes, el contador de bloques de 8 bytes y 16 bytes de constantes). Realiza 320 modificaciones invertibles de las palabras en un patrón particular, y al final suma las 16 palabras resultantes con las 16 palabras originales.

3. Intercambio de llaves de Diffie-Hellman sobre curvas elípticas

¿Cómo funciona Sodinokibi? Con el algoritmo AES encripta los datos de la sesión y los datos que se envían al servidor de control, mientras que con el algoritmo Salsa20 encripta los datos individuales. Con el intercambio de llaves de Diffie-Hellman sobre curvas elípticas crea las llaves de encriptación y desencriptación.

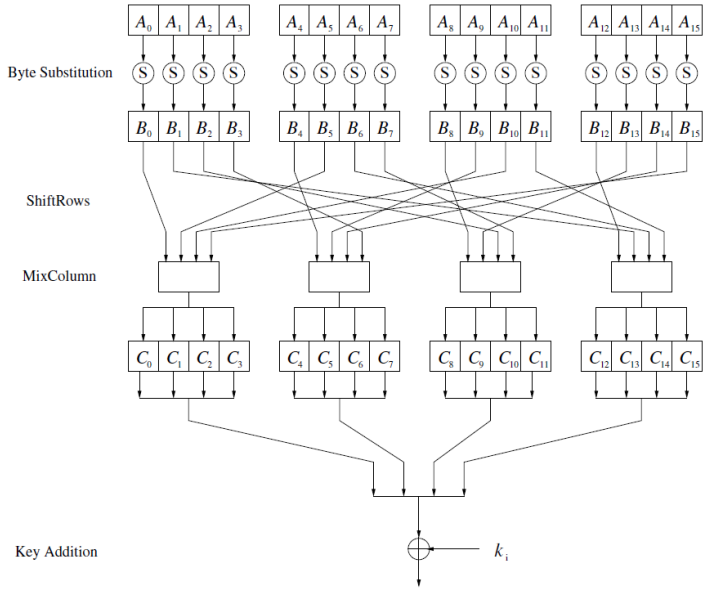

Figura 11: Una ronda del AES

Referencias

[1] Internet Live Stats, *http://www.internetlivestats.com/*.

[2] S. Vaudenay, *A Classical Introduction to Cryptography: Applications for Communications Security*, ISBN-13:978-1-4419-3797-1. Springer, 2010.

[3] A. Menezes, P. van Oorschot, S. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1997. Available at: www.cacr.math.uwaterloo.ca/hac

[4] C. Paar, J. Pelzl, *Understanding Cryptography. A Textbook for Students and Practitioners*, ISBN:978-3-642-04100-6. Springer-Verlag, Berlin, 2010.

[5] B. Schneier, *Applied Cryptography:Protocols, Algorithms, and Source Code in C*, Second Edition, ISBN:0471117099. John Wiley and Sons, Inc. 1995.

[6] W. Diffie, M.E.Hellman, *New directions in cryptography*, IEEE Transactions on Information Theory 22 (1976), pp. 644-654.

[7] R. Rivest, A. Shamir, L. Adleman, *A Method for Obtaining Digital Signatures and Public-Key Cryptosystems*, Communications of the ACM, Vol. 21 (2), pp.120-126, 1978.

[8] T. ElGamal, *A Public-Key Cryptosystem and a Signature Scheme Based on Discrete Logarithms*, IEEE Transactions on Information Theory 31 (4): 469Ű472, 1985.

[9] N. Koblitz, *Algebraic Aspect of Cryptography. Algorithms and Computation in Mathematics*, ISBN 3-540-63446-0, Springer-Verlag, Berlin, 1998.

[10] H. Cohen, G. Frey, Et. Al., *Handbook of Elliptic and Hyperelliptic Curve Cryptography: Theory and Practice*, ISBN: 1584885181. Chapman and Hall/CRC, 2005.

[11] H. C. A. van Tilborg, *Fundamentals of Cryptology. A Professional Reference and Interactive Tutorial*, ISBN: 0792386752. Kluwer Academic Publishers, 2002.

[12] T. W. Hungerford, *Abstract Algebra, an Introduction*, Brooks Cole, 1986.

[13] S. Lin, D. Costello *Error Control Coding. Fundamentals and Applications*, ISBN-10: 0130426725, Prentice-Hall, 2 edition. 2004.

[14] W. Fulton, *Algebraic Curves. An introduction to Algebraic Geometry*, W.A. Benjamin, Inc, N.Y. 1969.

[15] Welivesecurity 2020, *Ataque de ransomware afecta a Banco Estado en Chile* https://www.welivesecurity.com/la-es/2020/09/08/ataque-ransomware-afecta-bancoestado-chile/

[16] Bernstein, D.J. *The Salsa20 family of stream ciphers*, 2005 https://cr.yp.to/salsa20.html

[17] Bernstein, D.J. *Salsa20 design*, 2005 https://cr.yp.to/snuffle.html