Facultad de Ingeniería, UNAM
Sistemas Operativos (2024-1)
Cervantes García, Eduardo
Chagoya Gonzalez, Leonardo
Ingeniería en computación
Grupo 06
20 de noviembre de 2023

Algoritmo: Rivest, Shamir y Adleman, su relación en sistemas operativos

Abstract

Desde la llegada de la computación cuántica y el descubrimiento de las ventajas que pueden presentar los qbits frente a los bits en algunas operaciones. El mundo se dió cuenta que gran parte de los problemas matemáticos de cálculo bruto podrían ser resueltos de manera sencilla y en tiempo record. Para la informática, en específico la criptografía esto plantea un escenario sombrío debido a que gran parte de los algoritmos que aseguran la información actualmente podrían ser rotos sin problema alguno por esta tecnología. Esta noticia despertó interés en los involucrados en este texto, por lo cual decidimos investigar sobre los algoritmos criptográficos actuales, llegando a RSA (Rivest, Shamir y Adleman), uno de los algoritmos criptográficos más usados actualmente por la seguridad que provee. Así que en este artículos nos adentraremos en la criptografía para formar una noción general de esta, en el algoritmo de interés y cómo se relaciona con los sistemas operativos el cual es el interés de la materia. Esperando transmitir la importancia de este algoritmo en la actualidad, formando un panorama en el presente para poder afrontar el futuro.

Introducción

De manera concisa el cifrado es la transformación de datos o información a un formato codificado, los datos persisten, pero no cualquiera puede entenderlos si no conoce la codificación. Al proceso de recuperar o entender información codificada se le conoce como descifrado. Estos dos conceptos son la base o el nodo padre del árbol de la ciberseguridad. El fin de la ciberseguridad es proteger datos sensibles o simplemente información que cualquier persona o ente no quiere que sea de acceso público, teniendo en cuenta que cualquier cosa que se envía y recibe en una computadora pasa por varios nodos de la red de internet, antes de llegar a su destinatario.

El cifrado puede ser aplicado mediante hardware o software para el enfoque de este artículo se hará énfasis totalmente en el software. Revisando uno de los algoritmos más populares de llave pública, como se implementa o lo implementan los sistemas operativos.

“En modo más formal, el cifrado implica utilizar una clave criptográfica; un conjunto de valores matemáticos que acuerdan tanto el emisor como el receptor. El receptor utiliza la clave para descifrar los datos y volver a convertirlos en texto sin formato legible” (Kaspersky, 2021)

Hay dos conceptos más que resultan importantes, cifrado activo y en reposo. El primero hace referencia a cifrar datos que se mueven constantemente entre redes, mientras que el segundo a datos que suelen estar sin moverse en un medio de almacenamiento, como la mayoría de los que se tienen en las computadoras personales, sino se usa un servicio de nube. El cifrado en reposo suele ser más seguro por sus características, por tanto, se recomienda tener cifrados tus datos como medida de prevención. Las grandes compañías suelen cifrar su bases de datos y la mayoría de sistemas operativos proveen una u otra manera para hacerlo, ya que como se mencionó en clase, el código o los programas como un OS deben hacerse pensando en la seguridad.

Usos del cifrado:

- “Cada vez que usas un cajero automático o compras algo en línea mediante un teléfono, se usa cifrado para evitar que se desvíe la información.
- Para proteger dispositivos, como el cifrado para computadoras portátiles.
- La mayoría de los sitios web legítimos utilizan la "capa de sockets seguros" (SSL), lo cual es una forma de cifrar datos cuando se envían desde y hacia un sitio web. Esto evita que los atacantes accedan a esos datos mientras están en tránsito. Busca el ícono de candado en la barra de direcciones URL y la letra "s" en "https:/" para asegurarte de que realizas transacciones seguras y cifradas en línea.
- Tus mensajes de WhatsApp también están cifrados y es posible que también tengas una carpeta cifrada en tu teléfono.
- Tu correo electrónico también puede estar cifrado con protocolos como OpenPGP.
- Las VPN (redes privadas virtuales) utilizan cifrado y todo lo que almacenes en la nube debiera estar cifrado. Puedes cifrar todo tu disco duro e incluso realizar llamadas por voz cifradas.
- El cifrado se utiliza para demostrar la integridad y autenticidad de la información utilizando lo que se conoce como firmas digitales. El

cifrado es una parte integral de la administración de derechos digitales y de la protección contra copias.

- El cifrado puede utilizarse para borrar datos. Debido a que la información borrada a veces puede recuperarse mediante herramientas de recuperación de datos, si cifras los datos primero y descartas la clave, lo único que alguien podrá recuperar es el texto cifrado y no los datos originales".

  (Kaspersky, 2021)

La criptografía en computación es un concepto que pretende mediante la automatización con algoritmos garantizar las propiedades de confidencialidad e integridad de información no pública. Estos algoritmos o mecanismos garantizan en cierta medida las propiedades mencionadas, ya que es imposible una garantía completa porque la comunicación implica sistemas no cerrados. Uno de los algoritmos provenientes de la criptografía y su impacto en los sistemas operativos, será parte de lo visto en el artículo.

Desarrollo

¿Qué es la criptografía ?

La palabra criptografía proviene del griego kryptos: "ocultar", y grafos: "escribir". En pocas palabras, significa "escritura oculta". De forma más explícita, la criptografía se ocupa de concebir funciones o dispositivos capaces de convertir mensajes legibles en mensajes cifrados,utilizando técnicas matemáticas , de modo que tanto la transformación (cifrado) como su inversa (descifrado) sólo sean viables con el conocimiento de una o varias claves. Es clara la relación con el cifrado, por lo cual puede deducirse que la criptografía es la ciencia encargada del cifrado de datos. A partir de la relación anterior, en párrafos subsecuentes se utilizará "cifrado y encriptado" así como "descifrado y desencriptado " de forma indistinta, siendo los primeros mencionados un sinónimo más accesible de los conceptos, al dirigirse a todo el público.

La criptografía puede clasificarse de muchas formas, según sea el criterio: Históricamente se encuentra la criptografía clásica así como la criptografía moderna. Y dentro de la criptografía moderna encontramos la clasificación por el tipo de llave que se utiliza: La simétrica y la asimétrica. Dentro de estas se encuentran nuestro algoritmo de interés: RSA (Rivest, Shamir y Adleman).

“En su clasificación dentro de las ciencias, la criptografía proviene de una rama de las matemáticas, que fue iniciada por el matemático Claude Elwood Shannon en 1948, denominada: “Teoría de la Información”. Esta rama de las ciencias se divide en: “Teoría de Códigos” y en “Criptología”. Y a su vez la Criptología se divide en Criptoanálisis y Criptografía.” (Granados, 2006)

En contraparte, el criptoanálisis es la ciencia que estudia los métodos que se utilizan para, a partir de uno o varios mensajes cifrados, recuperar los mensajes en claro en ausencia de la(s) llave(s) y/o encontrar la llave o llaves con las que fueron cifrados dichos mensajes.

Como se mencionó, la criptografía históricamente se divide en clásica y moderna. La clásica abarca todo lo realizado en la criptografía hasta la mitad del siglo XX. La moderna puede decirse que tiene su desarrollo junto a las computadoras modernas, siendo esta la que se abordará en las siguientes hojas ya que el cifrado de llave pública, inventado en 1976 termina siendo parte de la criptografía moderna. Es importante aclarar que los conceptos vistos desde cifrado hasta el momento son aplicables para la criptografía moderna, ya que es el caso de estudio, dentro de la criptografía moderna existen más formas de subdividir sus partes lo cual ayudará a profundizar en el tema.

Criptografía Simétrica

“Imaginemos que necesitamos enviar por Internet un dato confidencial, y queremos que solo el destinatario pueda verla. Una forma de hacerlo sería meter la información en una caja fuerte y cerrarla con una determinada combinación. Luego enviamos la caja fuerte al destinatario, y el destinatario podrá ver su contenido siempre y cuando sepa la misma combinación que se utilizó para cerrarla” (Córdoba, 2016)

Esto es el principio de la criptografía simétrica, los algoritmos que pertenecen a este tipo son conocidos como criptosistemas de clave privada o de clave secreta, como apreciamos en el ejemplo, este tipo de criptografía se caracteriza por tener una sola llave para cifrar y descifrar el mensaje por lo cual es algo insegura pero a su vez también es más veloz.

Criptografía asimétrica

“Imaginemos ahora otro escenario. Supongamos que la caja fuerte ahora tiene dos cerraduras, cada una con una llave, una para cerrarla, y otra para abrirla. Ninguna llave puede duplicarse y

la llave que cierra no puede usarse para abrir la caja, ni la llave que abre puede usarse para cerrar la caja.

El destinatario tiene esta caja y nos la envía, junto con la llave para cerrarla, mientras que él se queda con la llave para abrirla. Nosotros podríamos introducir el mensaje en dicha caja, cerrarla con la llave provista, y enviársela al destinatario.“ (Córdoba, 2016)

Esto es como se maneja la criptografía asimétrica, en ella cada usuario tiene dos llaves: una pública y otra privada. Esto con el fin de cifrar el mensaje con una de las llaves y descifrar con la llave restante. De esta forma los atacantes podrán interceptar el mensaje cifrado, así como conseguir la llave pública , pero no conseguirán ver el contenido real del mensaje ya que aún tendrán que conseguir la llave privada, es en esta llave donde radica la seguridad puesto que se implementan algoritmos matemáticos con el fin de generar las llaves de longitudes mayores , en comparación con los de llave privada, y por ende más seguras para cada usuario. Algunos ejemplos de este tipo de criptografía son RSA, El Gamal y Curvas Elípticas.

Algoritmos criptográficos de llave pública

La criptografía de llave pública fue el verdadero antes y después de la criptografía ya que está basada en funciones matemáticas aprovechando el poder computacional de la era moderna. Estos algoritmos son asimétricos ya que como se verá posteriormente, hace uso de dos llaves.

La noción de estos algoritmos consiste en que las llaves criptográficas de los algoritmos pueden venir en pares, con una llave para encriptar la información y otra para desencriptarla y que no se puede generar fácilmente una a través de la otra. El concepto fue inventado por Whitfield Diffie y Martin Hellman en 1976, al mismo tiempo y de forma independiente por Ralph Merkle. A partir de la invención de este concepto, se han generado muchos algoritmos con él; algunos algoritmos inseguros, otros que aún son seguros, algunos imprácticos con llaves muy largas o imprácticos porque la información encriptada ocupa más almacenamiento que la información legible

Solo algunos algoritmos de llave pública han conseguido ser prácticos y seguros, entre ellos están el que se abordará: el algoritmo Rivest, Shamir y Adleman. La mayoría de estos algoritmos funcionales provienen de sólidas bases matemáticas o están inspirados en

complejos problemas matemáticos. Lo cual los convierte en algoritmos seguros contra ataques de fuerza bruta.

En este punto se debe dejar claro que la seguridad de un sistema provista por la criptografía depende del largo de la o las llaves y el esfuerzo computacional que requiere deshacer la encriptación. Y que, aunque este tipo de algoritmos sean lo más famosos en la actualidad no dejan obsoletos a los algoritmos criptográficos considerados como simétricos ya que estos pueden ser mejores en algunas aplicaciones.

Principios de llave pública

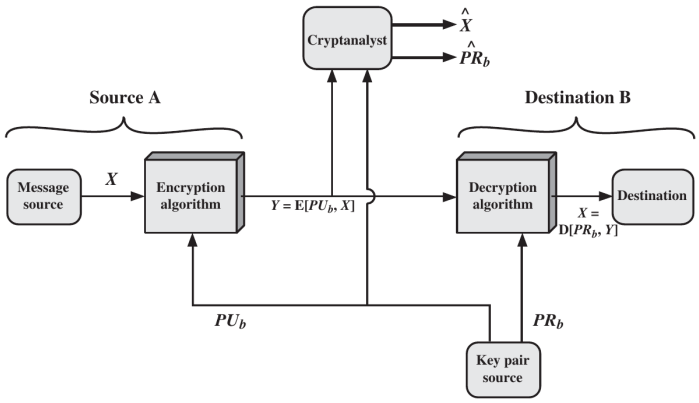
Surgen como la resolución de un problema para algunas aplicaciones de algoritmos simétricos, como, que en la encriptación simétrica se requiere que los dos nodos que se comunican hayan ya compartido su llave o la hayan obtenido de alguna manera con un centro de distribución de claves, lo cual no permite mantener anonimato total en la comunicación.

Estos algoritmos contienen una llave para encriptar y una distinta, pero relacionada, para desencriptar. Su principal característica es que hacen que sea inviable computacionalmente determinar la llave de desencriptación teniendo solo conocimiento del algoritmo y la clave de cifrado. Algo que no es propio de todos, pero sí de algunos y vale la pena comentar al ser parte de RSA es que: ambas claves pueden usarse para encriptar con la otra siendo usada para desencriptar.

Un algoritmo de llave pública tiene 6 partes:

- **Texto sin formato (Plaintext):** Este es el mensaje o los datos legibles que se ingresan al algoritmo como entrada.
- **Algoritmo de cifrado (Encryption algorithm):** El algoritmo de cifrado realiza diversas transformaciones en el texto sin formato.
- **Claves pública y privada:** Es un par de claves seleccionadas de manera que si una se usa para cifrar, la otra se utiliza para descifrar. Las transformaciones exactas realizadas por el algoritmo dependen de si se proporciona la clave pública o privada como entrada.
- **Texto cifrado (Ciphertext):** Este es el mensaje encriptado que se produce como salida. Depende del texto sin formato y de la clave. Para un mensaje dado, dos claves diferentes generarán dos textos cifrados diferentes.
- **Algoritmo de descifrado (Decryption algorithm):** Este algoritmo acepta el texto cifrado y la clave correspondiente, y produce el texto sin formato original.

Las 6 partes se ven involucradas de la siguiente forma: estos algoritmos comienzan con un usuario generando su par de claves para encriptar y desencriptar, una de las claves es expuesta de forma pública ya sea como parte de un mensaje o en un registro público de la aplicación que usa el cifrado. Cuando un usuario quiere mandar un mensaje confidencial a otro, toma la clave pública de ese destinatario del registro y aplica el algoritmo a su mensaje con esa clave pública, entonces los datos se envían cifrados. Los datos cifrados sólo podrán ser descifrados con la llave privada del destinatario y con ninguna otra, de esta manera se asegura que sólo el destinatario correcto y con su clave privada en posesión, pueda acceder a la información. Es por lo mencionado que ambas claves de un usuario deben estar relacionadas.



Aplicaciones de llave pública

Dependiendo de la aplicación, el remitente usa su llave privada o la llave pública del destinatario, o las dos para ejecutar algún tipo de función criptográfica. Frente a esto se pueden clasificar las aplicaciones de los algoritmos de llaves públicas en 3 distintas:

- **Cifrado/Descifrado:** El remitente cifra un mensaje con la clave pública del destinatario.
- **Firma digital:** El remitente "firma" un mensaje con su clave privada. La firma se logra mediante un algoritmo criptográfico aplicado al mensaje o a un pequeño bloque de datos que es una función del mensaje.
- **Intercambio de claves:** Dos partes cooperan para intercambiar una clave de sesión. Se pueden utilizar varios enfoques diferentes, que involucran la(s) clave(s) privada(s) de una o ambas partes.

No todos los algoritmos son útiles para todas las aplicaciones, en el caso de RSA es adecuado y puede ser usado en cualquiera de ellas

Seguridad de los algoritmos

La seguridad de estos algoritmos es muy alta debido a que su principio base es hacer computacionalmente inviable el descifrado si no se tiene la clave adecuada, así se tenga una y se conozca el algoritmo. Aunque es posible realizar ataques de fuerza bruta conociendo el algoritmo, o el algoritmo y una de las llaves, es inviable computacionalmente por la naturaleza binaria de las computadoras que usamos. Los recursos necesarios para realizar fuerza bruta suelen ser más costosos que el valor que se obtiene por el descifrado, y en la mayoría de los casos es prácticamente imposible el descifrado. Si la seguridad en estos algoritmos no fuera suficiente, basta con hacer más largas las llaves.

RSA (Rivest, Shamir y Adleman)

Es un algoritmo de cifrado de clave pública el cual fue desarrollado por los informáticos y criptógrafos del MIT Ron Rivest, Adi Shamir y Leonard Adleman en el año de 1977. La fortaleza de este sistema radica en la dificultad de la factorización de números primos grandes, pues funciona multiplicando dos números primos para generar un semiprimo, que crea una clave pública.

La parte más difícil de este sistema es la generación de claves, para la cual se siguen una serie determinada de pasos:

1. Seleccionamos dos primos p y q. Se recomienda que ambos primos sean de tamaño mínimo de 512 bits

   Elegimos números pequeños para ejemplificar, siendo $p = 89\, y\, q = 31$.
2. Calculamos un parámetro llamado n y otro parámetro φ. $n = p * q$ mientras que $\varphi = (p - 1)(q - 1)$.

   Los numero φ serán llamados también como “trampa”

$$n = 89 * 31 = 2\,759$$
$$\varphi = 88 * 30 = 2\,640$$

3. Se busca un número “e” impar que no tenga múltiplos comunes con φ. Debemos seleccionar de manera aleatoria un número “e” que cumpla que dicho número sea mayor a 1 pero menor que φ ($1 < e < \varphi$). Lo cual será nuestra llave pública

   Elegimos a $e = 29$ cumple con qué mayor a 1 y menor a $2\,640$ y no tienen múltiplos comunes, pues

$$29 * 91 = 2639\ y\ 29 * 92 = 2668$$

4. Para calcular la llave privada nos apoyamos en el algoritmo extendido de euclides. Calculamos un parámetro llamado $d = inv(e, \varphi)$. Lo cual será nuestra llave privada.

Importante mencionar que inv es el inverso multiplicativo modular.

$$d = inv[e, \varphi] = [29, 2640] = 2549$$

Para cifrar y descifrar ,la llave privada estará compuesta por (d, n). Mientras que la llave pública estará compuesta por (e, n)

Cifrado y descifrado

Una vez que se tienen los parámetros públicos y privados se puede pasar a cifrar o descifrar cualquier mensaje mediante fórmulas basadas en el teorema chino de restos.

El criptograma C estará dado por

$$C = M^e mod\, n$$

Donde M es el mensaje claro es el número primo de la clave pública n el parámetro calculado a raíz de la multiplicación de los dos números primos seleccionados. La fórmula para descifrar el criptograma C es

$$M = C^d mod\, n$$

Donde M es el mensaje descifrado

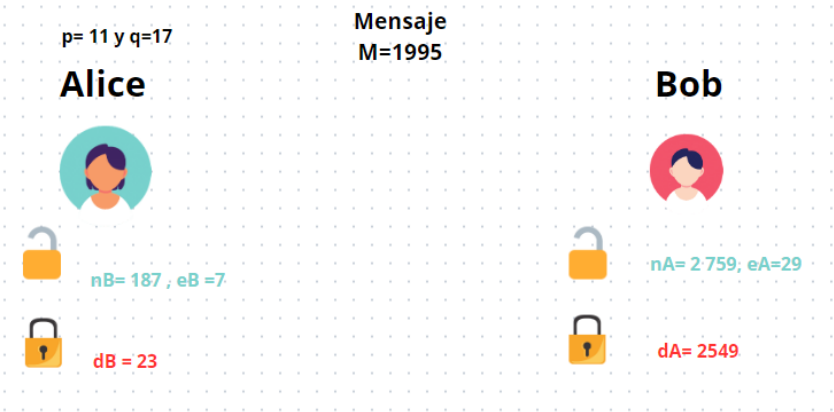
C es criptograma

d es la llave privada

n el parámetro calculado a raíz de la multiplicación de los dos números primos seleccionados

Es importante mencionar que el mensaje M tiene que encontrarse codificado para poder aplicar la encriptación. Vamos con un ejemplo de intercambio de mensajes entre dos usuarios.

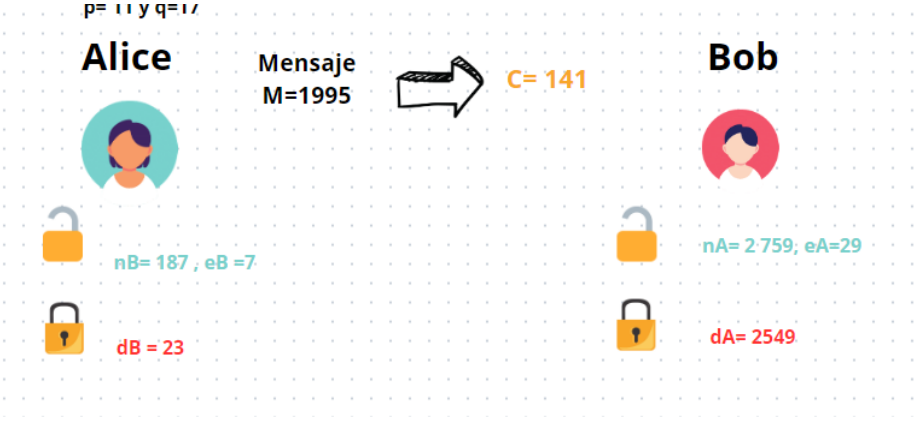
Suponiendo que tenemos a Alice y a Bob y ya calculamos sus respectivos parámetros para las claves públicas y privadas. Y teniendo un mensaje codificado $M = 1\,995$



Si alice quiere mandar su mensaje a bob tiene que encriptarlo mediante la fórmula

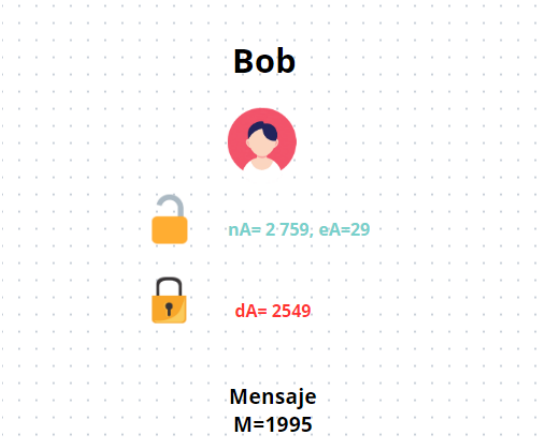
$C = M^e mod\, n$ tomando las credenciales públicas de Bob (nB y eB). De esta forma el mensaje de Alice encriptado seria

$$C = 1995^{(29)} mod\, 2759 = 141$$



Y Bob para descifrar el mensaje de Alice tiene que ocupar su propia llave privada haciendo uso de la fórmula $M = C^d mod\, n$. Es así que

$$M = C^d mod\, n = 141^{2549}\, mod\, 2759$$
$$M = 1995$$



De esta forma a Bob le puede llegar de manera segura el mensaje de Alice.

Relación a los sistemas operativos

Implementación Nativa

Después de una amplia búsqueda de aplicaciones. Se encontró que de una u otra forma este algoritmo es implementado por la mayoría de Sistemas Operativos famosos. Su implementación es parecida a los visto en clase de hilos en Windows o Linux . Si bien el usuario puede llamar a estas implementaciones mediante lenguajes de programación y usarlas, de los detalles de bajo nivel se hace cargo el sistemas operativo. Los sistemas operativos suelen incluir bibliotecas que facilitan la aplicación de este algoritmo y es usado en algunos de sus servicios.

Windows provee la implementación de RSA y la creación de llaves privadas de manera nativa con su biblioteca Wyncript. Además, Windows suele usar o dar como opción este algoritmo al usar algunos de sus servicios como Bitlocker o autenticaciones para escritorio remoto. En el caso de Bitlocker es un tema complicado ya que si el usuario que cifró su información o una unidad de almacenamiento con RSA pierde su clave privada, como se vió en la explicación del algoritmo, no hay forma de que recupere sus datos.

Otro sistema que implementa mediante una biblioteca nativa este algoritmo es la distribución de Linux RedHat, la cual está centrada en ciberseguridad y pone al algoritmo RSA en su documentación como un estándar de cifrado de llave pública.

Timing attacks

Estos ataques tienen su base en el tiempo de ejecución del algoritmo de descifrado. Paul Kocher se dió cuenta que en algunas implementaciones(KOCH96, KALI96b]) del algoritmo si se mide el tiempo que este tarda en ejecutarse, puede determinarse la llave privada de un usuario. Este ataque puede tomar desprevenido a cualquiera ya que deja en evidencia que un ataque puede venir o ser desarrollado en una perspectiva inesperada y ataca a partir de sólo texto cifrado. Esto también pone en evidencia lo difícil que es asegurar la seguridad criptográfica de un algoritmo. Este tipo de ataque es similar o se puede hacer una analogía a algún ladrón que descifra la clave de una caja fuerte observando el tiempo que le toma a alguien pasar de número a número de la clave al introducirla.

Con el ejemplo anterior, el ataque es más fácil de explicar. Suponiendo que el algoritmo de RSA utiliza una función de multiplicación modular que en algunos casos será muy rápida, pero en algunos

otros toma más tiempo que una exponenciación modular promedio. Frente al conocimiento de los tiempos de ejecución de la operación y el texto cifrado, se puede decir que el atacante va adivinando los bits promediando con distintas ejecuciones del algoritmo al descifrar hasta que logra adivinar todos los bits luego de muchas ejecuciones, o fuerza bruta.

Hay distintas soluciones para este programa, algunas mejores que otras, o con efectos secundarios como pérdidas de rendimiento pero, se abordará la relacionada con la materia.

Una de las soluciones es el “random delay” o hacer que el proceso que ejecuta este algoritmo tenga esperas activas aleatorias, algo que conocemos no es muy bueno para la concurrencia en un sistema, pero termina haciendo inconsistentes los tiempos de ejecución, volviendo la obtención de la clave pública más difícil o imposible. Aunque si no se hace de la manera correcta puede solo hacer que sea más complejo para el atacante, pero al final de cuentas lo mismo.

La solución siguiente no fue encontrada, pero fue pensada a partir de las solución de “random delay”. Esta solución depende del uso que se le dé al sistema donde se desencripta, pero por ejemplo: si es una base de datos, manejada con una terminal la cual no espera interactividad y una UI para el usuario podría usarse un planificador de procesos como “Round Robin” con un quantum establecido que favorezca al servidor, esto hará que las veces que el servidor ejecute el algoritmo de descifrado siempre duren lo mismo sus operaciones y por lo tanto el atacante no podría realizar su ataque de esa forma y los datos permanecerán seguros.

Conclusiones

Después de estudiar el algoritmo y sus implementaciones puede dimensionarse el nivel de importancia que tiene este en el mundo, al ser implementado de manera nativa por sistemas operativos y hasta ser considerado un estándar de llave pública. Este algoritmo es un punto de inflexión en la criptografía por lo cual como ingenieros en computación es importante conocer de él, más si se piensa en dedicarse a ciberseguridad; para entender la criptografía moderna debe entenderse RSA y los algoritmos de llave pública. Sin embargo, este algoritmo también logra enseñar que no existe el algoritmo de seguridad perfecto en sistemas que se comunican y pueden analizarse algunos de sus defectos mediante lo aprendido en el curso. De esta forma este artículo logra ser una introducción amena a la criptografía junto a conceptos ya conocidos.

Referencias

alvinashcraft. (2023, agosto 27). *CRYPT_ALGORITHM_IDENTIFIER (wincrypt.h) - Win32 apps*.Learn.microsoft.com.https://learn.microsoft.com/es-es/windows/win32/api/wincrypt/ns-wincrypt-crypt_algorithm_identifier

Córdoba, D. (2016, noviembre 8). *RSA: ¿Cómo funciona este algoritmo de cifrado?* Junco TIC. https://juncotic.com/rsa-como-funciona-este-algoritmo/

Cyber zaintza. (n.d.). *RSA (rivest, shamir y adleman) | BCSC*. Www.ciberseguridad.eus. Recuperado noviembre 21, 2023, from https://www.ciberseguridad.eus/ciberglosario/rsa-rivest-shamir-y-adleman

Granados, G. (2006). *INTRODUCCIÓN A LA CRIPTOGRAFÍA*. https://www.revista.unam.mx/vol.7/num7/art55/jul_art55.pdf

Kaspersky. (2021, enero 13). *¿Qué es el cifrado de datos?* Latam.kaspersky.com. https://latam.kaspersky.com/resource-center/definitions/encryption

RedHat Customer Portal. (n.d.). *A.2. Cifrado de llave pública Red Hat Enterprise Linux 6 | Red Hat Customer Portal*. Access.redhat.com. Recuperado noviembre 21 , 2023, from https://access.redhat.com/documentation/es-es/red_hat_enterprise_linux/6/html/security_guide/apas02

Schneier, B. (1996). *Applied cryptography, second edition : protocols, algorithmos, and source code in c* (2nd ed.). J. Wiley.

Stallings, W. (2011). *Cryptography and network security : Principles and practice* (5th ed.). Prentice Hall.