CIFRADO SIMÉTRICO Y ASIMÉTRICO

La criptografía nació de la necesidad de hacer privada una transmisión de datos, así que en base, la criptografía es el método que se aplica a un mensaje para cifrar y descifrar de tal manera que solo sea visible el texto en claro para el emisor y el receptor de dicho mensaje, que sea totalmente ilegible para cualquiera que lo intercepte en mitad de la transmisión, y a su vez, que sea lo suficientemente seguro como para aguantar cualquier ataque que pueda comprometer la información que se está transmitiendo. Existen dos tipos de cifrado según el tipo de sus claves. La criptografía cuyos algoritmos solo usan una clave, o criptografía simétrica, y la criptografía de algoritmos de dos llaves, criptografía asimétrica. Aquí vamos a poder ver cómo funciona cada una y cuáles son sus propiedades y diferencias.

Criptografía simétrica o criptografía de una clave

La criptografía simétrica es la técnica criptográfica más antigua que existe, pero sigue ofreciendo un alto nivel de seguridad. Se basa en la utilización de una única clave secreta que se encargará de cifrar y descifrar la información, ya sea información en tránsito con protocolos como TLS, o información en un dispositivo de almacenamiento extraíble. La criptografía simétrica fue el primer método empleado para el cifrado de la información, se basa en que se utilizará la misma contraseña tanto para el cifrado como el descifrado, por tanto, es fundamental que todos los usuarios que quieran cifrar o descifrar el mensaje, tengan esta clave secreta, de lo contrario, no podrán hacerlo. Gracias a la criptografía simétrica, podremos realizar comunicaciones o almacenar archivos de forma segura.

El cifrado mediante una clave simétrica, significa que, como mínimo, dos usuarios deben poseer la clave secreta. Utilizando esta clave se cifran y descifran todos los mensajes transmitidos a través del canal inseguro, como Internet, de ahí que necesitemos el cifrado de los datos para mantener la confidencialidad. Cualquier usuario que quiera acceder al mensaje cifrado, deberá tener esta contraseña de descifrado, de lo contrario será ilegible.

El método para cifrar los datos se basa en que el emisor va a cifrar el mensaje con su clave privada, lo enviará a través del canal inseguro, y el destinatario lo tendrá que descifrar con la misma contraseña o clave privada que ha usado el emisor.

Hay varios puntos que debe cumplir un algoritmo de clave simétrica para que su uso sea seguro:

- Una vez que se cifra el mensaje, **no se podrá obtener la clave de cifrado ni tampoco el mensaje en claro** por ningún método.
- Si conocemos el mensaje en claro y el cifrado, **se debe gastar más tiempo y más dinero en obtener la clave para acceder al mensaje en claro, que el posible valor que pueda tener la información** que se consiga robar.

Los **ataques por fuerza bruta** son el **enemigo real de los algoritmos de criptografía simétrica**, hay que tener en cuenta que estos algoritmos son públicos y que la fuerza de los mismos depende directamente de lo complejo que sea el algoritmo internamente, y también de la longitud de la clave empleada para evitar estos ataques.
Lo más importante en la criptografía simétrica es proteger la clave privada o contraseña. El principal problema que se presenta es la distribución de esta clave privada o contraseña a todos los usuarios, tanto emisores como receptores de la información, para cifrar y descifrar la información del mensaje. Es fundamental que todos los usuarios del sistema protejan la clave privada o contraseña lo mejor posible, porque si cae en malas manos, todo el sistema se vería comprometido, teniendo que generar una nueva clave y volviendo a redistribuir la clave privada a los diferentes participantes.
Una de las diferencias que tenemos entre la criptografía simétrica y asimétrica, es que en la simétrica todos los usuarios tienen la misma clave de cifrado/descifrado, si tenemos más de un canal de comunicación, tendremos tantas claves privadas como canales de comunicación paralelos. Sin embargo, en los asimétricos cada usuario tendrá una pareja de claves (pública y privada) para todos los canales de comunicación, no importa el número de canales seguros a mantener, solamente deberemos proteger la clave privada.
La ventaja de los algoritmos de criptografía simétrica es que son rápidos, muy rápidos, además, en los últimos años se han ido incorporando a los procesadores de ordenadores, servidores, routers y otra serie de dispositivos la aceleración de cifrado por hardware, de esta forma, podremos transferir datos vía VPN de forma realmente rápida. La velocidad también depende del algoritmo de cifrado simétrico a utilizar, por ejemplo, AES y ChaCha20 son dos de los más rápidos y seguros que tenemos hoy en día, pero influye mucho el hardware de los equipos.

Ahora os vamos a hablar en detalle de dos algoritmos simétricos que se utilizan continuamente en la actualidad, ya sea en las conexiones HTTPS con TLS 1.2 y TLS 1.3, en las redes privadas virtuales con IPsec, OpenVPN o WireGuard, y también en otros ámbitos donde se utilice cifrado de datos, como en VeraCrypt.

AES (Advanced Encryption Standard)

El algoritmo simétrico AES fue el encargado de **sustituir al DES**, y es el empleado actualmente en todos los canales y protocolos seguros como TLS, FTPES, redes privadas virtuales y mucho más. El cifrado de AES puede ser empleado tanto en software como en hardware, AES es un algoritmo de cifrado por bloques, **el tamaño fijo del bloque es de 128 bits**. La longitud de la clave se puede elegir, y tenemos disponible **128, 192 y 256 bits**, siendo la longitud de 128 bits el estándar, pero también es muy utilizado los 256 bits.
Un aspecto muy importante, es que AES se encarga de generar una matriz de 4×4, y posteriormente se le aplican una serie de rondas de cifrado. Para una clave de 128 bits se aplican 10 rondas de cifrado, para una clave de 192 bits se aplican 12 rondas, y para una clave de 256 bits las rondas aplicadas son 14. Desde los inicios, muchos criptógrafos dudan de su seguridad, y es que se han registrado ataques a un número de rondas cercanas a la ronda final, concretamente se han podido descifrar 7 rondas para claves de 128 bits, 8 rondas para claves de 192 bits y 9 rondas para claves de 256 bits.
El modo de cifrado es la forma en que se gestionan los bloques del mensaje cifrado con AES, existen diferentes tipos, y cada uno de ellos funciona de una forma diferente. Por ejemplo, existe el **AES-CBC, AES-CFB y AES-OFB**, os vamos a explicar qué es exactamente esto que aparece en las librerías criptográficas como OpenSSL y LibreSSL.

- **CBC (Cipher-block chaining)**: este modo de cifrado ha sido ampliamente utilizado junto con una función hash para comprobar la autenticidad de los datos, y hoy en día se sigue utilizando. Este modo de cifrado consiste en que a cada bloque de texto plano, se le aplica la operación XOR con el bloque de cifrado anterior. Cada bloque cifrado, depende de lo anterior procesado hasta ese punto. Para realizar esta opción XOR con el primer bloque de texto, se hace uso de un vector de inicialización IV. Este modo de cifrado se realiza de forma secuencial, no permite ser tratado de forma paralela para aumentar el rendimiento en el cifrado/descifrado de los datos.
- **OFB (Output feedback)**: en este modo se utiliza la clave secreta para crear un bloque pseudoaleatorio al que se le aplica la operación XOR con el texto en claro para crear el texto cifrado. En este caso también se necesita un vector de inicialización que debe ser único para cada mensaje cifrado. Si no se utiliza un IV diferente, se compromete la seguridad del sistema. Tampoco se puede parelizar.
- **CFB (Cipher feedback)**: se hace igual que en OFB, pero para producir el keystream cifra el último bloque de cifrado, en lugar del último bloque del keystream como hace OFB. El cifrado no puede ser paralelizado, sin embargo, el descifrado sí.
- **GCM** (**Galois/Counter Mode)**: este modo de cifrado es uno de los mejores en cuanto a seguridad y velocidad, GCM permite el procesamiento en paralelo y es compatible con procesadores AES-NI para acelerar el rendimiento en cifrado/descifrado de los datos. Este modo de cifrado es AEAD, además de cifrar los datos, también es capaz de autenticarlos y comprobar la integridad de los datos, para asegurarnos que no se ha modificado. GCM puede aceptar también vectores de inicialización aleatorios.

AES es actualmente uno de los algoritmos de cifrado simétrico más importantes y utilizados en todo el mundo, sin embargo, el modo de cifrado más recomendable es AES-GCM ya que incorpora AEAD. Cuando nosotros establecemos una conexión TLS 1.2 o TLS 1.3, el canal de datos casi siempre hace uso de AES-128-GCM o AES-256-GCM, ya que son los dos algoritmos de cifrado simétricos más utilizados por las conexiones HTTPS. Este algoritmo también es ampliamente utilizado en las conexiones FTPES para transferir datos cifrados y autenticados a través de servidores FTP, FTPES hace uso del canal de control basado en TLS 1.2 o TLS 1.3, y el canal de datos utiliza exactamente los mismos algoritmos que las conexiones HTTPS. Finalmente, AES también es ampliamente utilizado por las redes privadas virtuales (VPN) como IPsec o OpenVPN, en ambos protocolos VPN tenemos la posibilidad de elegir este tipo de cifrado simétrico para las conexiones.
Un detalle muy importante es que los procesadores actuales disponen de AES-NI, esto significa que tendremos aceleración de cifrado por hardware, por lo tanto, el cifrado y descifrado de los datos es realmente rápido, casi tanto como si no estuvieran cifrado, por lo que es ideal para conseguir un mayor ancho de banda al usar las VPN, y una mayor velocidad de lectura y escritura al usar servicios FTPES.

ChaCha20

El algoritmo ChaCha20 es un algoritmo de cifrado simétrico que **soporta claves de 128 y 256 bits** y de alta velocidad, a diferencia de AES que es un cifrado por bloques, ChaCha20 es un cifrado de flujo. Tiene características similares a **su predecesor Salsa20** pero con una función primitiva de 12 o 20 rondas distintas. Su código fue publicado, estandarizado por la IETF en la RFC 7539 y en implementaciones de software, es mucho más eficiente y rápido que AES, por lo que rápidamente se ha hecho un hueco dentro de los algoritmos más usados en la actualidad.

Para saber por qué se ha hecho tan famoso, vamos a meter a Google de por medio para que se pueda entender todo mucho más rápido. Las conexiones HTPPS están enfocadas a ofrecer la máxima seguridad en las webs que visitamos todos los días, fue el siguiente paso del protocolo HTTP el cual no tenía protección alguna. El cifrado, sin embargo, varía de un navegador a otro. Hasta hace algunos años, Chrome para Android ha estado utilizando AES-GCM como algoritmo de cifrado simétrico, sin embargo, Google lleva trabajando desde hace muchos años en cifrados más actuales, seguros y rápidos.

El salto de popularidad se produce cuando, tras su puesta en marcha en la versión de escritorio de Chrome, llega a Android el nuevo **ChaCha20 para la encriptación y Poly1305 para la autenticación.** Un trabajo titánico que se traduce en un algoritmo simétrico que ofrece más seguridad, y que es inmune a varios tipos de ataques. Sin embargo, lo más destacable es que consigue un rendimiento tres veces superior a protocolos algo más antiguos como puede ser AES, de esta manera, también se aprovechan mejor las capacidades de la CPU y se hace notar una reducción del 16% en el ancho de banda utilizado, lo que hace que se pueda aprovechar aún más la conexión.

ChaCha20 se utiliza ampliamente en las conexiones HTTPS, actualmente si queremos tener la mejor seguridad podemos elegir entre AES o ChaCha20, no obstante, este último nos proporciona un mayor rendimiento aunque tengamos aceleración de cifrado por hardware. Este protocolo también se utiliza en las conexiones SSH para administrar servidores, de esta forma, podremos no solamente administrarlos, sino hacer uso del protocolo de transferencia de ficheros basado en SSH que es SFTP, por lo que tendremos un gran rendimiento. Por último, otro protocolo muy popular que usa ChaCha20 es VPN WireGuard, de hecho, este protocolo de VPN solamente nos permite usar ChaCha20, por este motivo (entre otros) es mucho más rápido que OpenVPN o IPsec aunque tengamos aceleración de cifrado por hardware.

Criptografía asimétrica o criptografía de clave pública

La criptografía de clave asimétrica también es conocida como clave pública, **emplea dos llaves diferentes en cada uno de los extremos de la comunicación para cifrar y descifrar**. Cada usuario de la comunicación tendrá una clave pública y otra privada. **La clave privada tendrá que ser protegida y guardada por el propio usuario,** será secreta y no la deberá conocer absolutamente nadie ni tampoco debe ser enviada a nadie. La clave pública será accesible por todos los usuarios del sistema que quieran comunicarse.
La fortaleza del sistema por el cual es seguro este tipo de algoritmo asimétrico, es que está basado en funciones matemáticas las cuales **son fáciles de resolver en un sentido**, pero que **su resolución en sentido contrario es extremadamente complicada**, a menos que se conozca la clave. Las claves públicas y privadas se generan simultáneamente y están ligadas la una a la otra. La relación entre ambas debe ser muy compleja, para que resulte muy difícil que obtengamos una clave a partir de la otra, en este caso, que obtengamos la clave privada puesto que la pública la conoce toda persona conectada al sistema.
Las parejas de claves tienen varias y muy importantes funciones, entre las que destacamos:

- Cifrar la información.
- Asegurar la integridad de los datos transmitidos.
- Garantizar la autenticidad del emisor.

Cifrado con clave asimétrica

Si una persona con una pareja de claves cifra un mensaje con la llave pública del receptor, ese mensaje sólo podrá ser descifrado con la llave privada asociada. Si encriptamos un mensaje con la clave privada, no podremos desencriptar con la propia clave privada, deberemos usar la pública (en este caso no se considera cifrado, sino que se comprueba la autenticidad del emisor, con ello comprobaremos que el emisor es quien realmente dice ser).

La estructura matemática del funcionamiento del cifrado asimétrico es esta:

- Mensaje + clave pública = Mensaje cifrado
- Mensaje encriptado + clave privada = Mensaje descifrado
- Mensaje + clave privada = Mensaje firmado
- Mensaje firmado + clave pública = Autenticación

Como hemos comentado antes al hablar de la criptografía de clave simétrica, el cifrado simétrico aporta confidencialidad (sólo podrá leer el mensaje el destinatario). La criptografía asimétrica proporciona otras propiedades: autenticidad, integridad y no repudio. Para que un algoritmo sea considerado seguro debe cumplir lo siguiente:

- Si se conoce el texto cifrado, **debe resultar muy difícil o prácticamente imposible extraer el texto en claro y la clave privada** por cualquier método.
- Si se conoce el texto en claro y el cifrado, **debe resultar más costoso obtener la clave privada que el texto en claro**.
- Si los datos han sido cifrados con la clave pública, **sólo debe existir una clave privada capaz de descifrarlo**, y viceversa.

La ventaja del cifrado asimétrico sobre el simétrico, radica en que la clave pública puede ser conocida por todos los usuarios del sistema, sin embargo, no ocurre esto con la clave privada, y por parte del cifrado simétrico deben conocer la misma clave los dos usuarios (y la clave debe hacerse llegar a cada uno de los distintos usuarios por el canal de comunicación establecido).

Funcionamiento del sistema simétrico y asimétrico

El principal inconveniente que tiene este tipo de cifrado **es la lentitud**, el empleo de este tipo de claves ralentiza el proceso de cifrado de la comunicación. La solución a esto es usar tanto el cifrado asimétrico como el simétrico (como hacen protocolos como el IPSec o OpenVPN para las redes privadas virtuales, el HTTPS para las conexiones web seguras, o en las conexiones SFTP/FTPES).

Esta combinación de cifrados sucede de la siguiente manera. Creamos la clave del algoritmo simétrico, la ciframos con la clave pública del receptor, enviamos los datos cifrados por el canal de comunicación inseguro, y a continuación, el receptor descifrará los datos mediante su llave privada. Con la clave del algoritmo simétrico en los dos puntos, es cuando puede empezar la comunicación mediante el cifrado simétrico, lo que hace que la comunicación sea mucho más rápida que si usáramos solo criptografía asimétrica en toda la comunicación.

Desafío-Respuesta

Para aumentar la seguridad, **este método comprueba que el emisor es realmente quien dice ser**, para ello se envía un texto al emisor y éste lo cifrará con su clave privada (lo que está haciendo realmente es firmarlo), el emisor nos enviará el texto cifrado (firmado) y nosotros desciframos la clave (comprobaremos la firma) aprovechando que tenemos la clave pública del emisor, y por último, compararemos que el mensaje obtenido sea el mismo que enviamos anteriormente.
Si algún usuario se hace pasar por el emisor real, no tendría la clave privada por lo que el «desafío» no hubiera resultado satisfactorio y no se establecería la comunicación de los datos.

Firma digital

La firma digital **permite al receptor de un mensaje que el origen es auténtico**, también podremos comprobar si el mensaje ha sido modificado. Falsificar una firma digital es casi imposible a no ser que conozcan la clave privada del que firma (y ya hemos dicho anteriormente que la clave privada debe estar guardada, y que no la debe saber nadie). Aquí están las dos fases para la realización de la firma digital:

- Proceso de firma: el emisor cifra los datos con la clave privada y lo manda al receptor.
- Verificar la firma: el receptor descifra los datos usando la clave pública del emisor y comprueba que la información coincide con los datos originales (si coincide es que no se ha modificado).

En las firmas digitales se hace uso de las funciones hash como SHA2-256 y SHA2-512 ya que, como hemos explicado anteriormente, el cifrado asimétrico es lento. El emisor de la comunicación aplicará la función HASH al mensaje original para obtener con ello la huella digital. A continuación, se cifrará la huella con la clave privada y se envía al destinatario por el canal inseguro para que la descifre. El destinatario también aplicará la función hash a sus datos y comparará los resultados (la que ha obtenido y la que ha recibido). Si el resultado de la comparación de estos datos es negativo, es decir, que hay diferencias entre lo obtenido y lo recibido, la información ha sido alterada y los datos de la huella digital habrán cambiado. Si el resultado es el mismo, se llevará a cabo la comunicación sin problemas.

Con todo esto hemos cumplido:

- Autenticidad, **el emisor es quien dice ser**. La firma en origen y destino es la misma.
- Integridad, **el mensaje no ha sido modificado**. Lo obtenido y lo recibido es igual.
- No repudio, **el emisor no puede negar haber enviado el mensaje al receptor**. La firma digital no varía.

Si queremos introducir la confidencialidad a la comunicación, lo único que hay que hacer es que el emisor cifre el mensaje original con la clave pública del receptor.

Algoritmos de cifrado de clave asimétrica

Ahora listamos los dos principales algoritmos asimétricos que se utilizan en la actualidad y os explicaremos el funcionamiento de los mismos.

Diffie-Hellman

No es un algoritmo asimétrico propiamente dicho, es un protocolo de establecimiento de claves, **se usa para generar una clave privada a ambos extremos de un canal de comunicación inseguro**. Se emplea para obtener la clave privada con la que posteriormente se cifrará la información junto con un algoritmo de cifrado simétrico. El punto fuerte del Diffie-Hellman es que, su seguridad radica en la dificultad de calcular el logaritmo discreto de números grandes (Diffie-Hellmann también permite el uso de curvas elípticas).

El problema de este algoritmo es que no proporciona autenticación, no puede validar la identidad de los usuarios, por tanto, si un tercer usuario se pone en medio de la comunicación, también se le facilitaría las claves y, por tanto, podría establecer comunicaciones con el emisor y el receptor suplantando las identidades. Para evitar esto existen varias soluciones que mitigan y solucionan el problema, como hacer uso de certificados digitales.

RSA

El algoritmo asimétrico por excelencia, **este algoritmo se basa en la pareja de claves**, la pública y la privada de las que ya hemos hablado con anterioridad. La seguridad de este algoritmo radica en el problema de la factorización de números enteros muy grandes, y en el problema RSA, porque descifrar por completo un texto cifrado con RSA no es posible actualmente, aunque sí un descifrado parcial. Algunas características muy importantes de RSA es la longitud de clave, actualmente como mínimo se debe utilizar una longitud de 2048 bits, aunque es recomendable que sea de 4096 bits o superior para tener una mayor seguridad.

Ventajas:

- Se resuelve el problema de la distribución de las llaves simétricas (cifrado simétrico).
- Se puede emplear para ser utilizado en firmas digitales.

Desventajas:

- La seguridad depende de la eficiencia de los ordenadores.
- Es más lento que los algoritmos de clave simétrica.
- La clave privada debe ser cifrada por algún algoritmo simétrico.

DSA

Este algoritmo es también puramente asimétrico, una desventaja de DSA es que quiere mucho más tiempo de cómputo que RSA a igualdad de hardware. DSA se utiliza ampliamente como un algoritmo de firma digital, es actualmente un estándar, pero DSA no se utiliza para cifrar datos, solamente como firma digital. Este algoritmo se utiliza ampliamente en las conexiones SSH para comprobar la firma digital de los clientes, además, existe una variante de DSA basada en curvas elípticas (ECDSA), y está disponible en todas las librerías criptográficas actuales como OpenSSL, GnuTLS o LibreSSL. Otra característica de DSA es la longitud de clave, la mínima longitud de clave es de 512 bits, aunque lo más habitual es usar 1024 bits.

El algoritmo DSA es el utilizado de forma predeterminada por OpenSSH, el popular software del servidor/cliente SSH que usamos ampliamente en cualquier servidor NAS, router o switch compatible con este protocolo, además, tenemos también una

compatibilidad perfecta con los diferentes sistemas operativos, independientemente del software utilizado. DSA suele ser más utilizado en SSH que el popular RSA.

Ahora conoces los dos tipos de criptografía y las propiedades de cada una, con ello sabrás **dónde merece la pena usar un tipo y dónde el otro**. Con el tiempo estos métodos variarán o se actualizarán a unos más seguros, ya que con el crecimiento del rendimiento de los ordenadores se consigue potenciar los ataque a este tipo de método de seguridad, pero ahora mismo, los que están vigentes y siguen usándose por no haber sido desbancados aún, son los que os acabamos de explicar.

Diferencias cifrado simétrico y asimétrico

Después de explicar cómo funcionan los diferentes tipos de cifrados, vamos a hablar de las **diferencias** que hay entre el simétrico y asimétrico. Ambos permiten cifrar archivos y usar servicios como el correo electrónico con mayores garantías, pero cuentan con ciertas peculiaridades en cada caso.

La diferencia principal es que **el cifrado simétrico utiliza una única clave**. Por ejemplo, para enviar un correo electrónico a otra persona ciframos ese mensaje con una llave. Cuando llega al destinatario, éste tiene que usar esa misma clave para descifrarlo y poder leerlo. Por tanto, podemos decir que en este caso el emisor y receptor van a utilizar la misma clave para ese mensaje. Previamente, antes de enviar ese mensaje ambas partes deben comunicarse para conocer cuál es esa llave que van a tener que utilizar.

En cambio, **el cifrado asimétrico utiliza dos claves**. Una de ellas va a ser privada, mientras que la otra es pública. La clave pública va a encargarse de cifrar el mensaje de ese correo. La privada la va a utilizar el destinatario para descifrar el mensaje. Esto hace que sea más sencillo distribuir las claves y también es más seguro, ya que la clave privada no la va a conocer nadie más.

En este sentido, el cuidado que debemos tener es mayor en el caso del cifrado simétrico, ya que al compartir las claves pueden terminar en malas manos. El cifrado asimétrico no va a poner en riesgo la clave privada, por lo que no debemos de tener tanto cuidado en ese sentido para poder estar protegidos.

Otra diferencia es la **longitud de las claves**. En el caso de la simétrica, puede ser más corta que la asimétrica, que necesita tener una mayor longitud. La asimétrica va a necesitar un mensaje que ocupa más espacio y necesita más tiempo para procesarlo, además de que requiere de mayor potencia de los equipos.

El tipo de algoritmos que se utiliza varía según el tipo de cifrado. En el caso del simétrico, utiliza algoritmos como AES, DES, 3DES o RC4. Respecto al asimétrico, usa algoritmos como DSA o RSA.

En definitiva, como has podido ver existen diferencias entre el cifrado simétrico y asimétrico. El más reciente, el que además se utiliza más hoy en día, es el asimétrico. Es más seguro y ofrece diferentes ventajas respecto al anterior, como hemos explicado. No obstante, tiene también algunos puntos negativos, como es el hecho de necesitar una mayor longitud de clave y tardar más en el proceso.