En esta ocasión seguiremos con la parte de seguridad de la información, ya que el capítulo 4 quedo un poco incompleto.

‘‘**Ponele que me robo una base de datos de usuarios y contraseñas. Viene cifrada ¿No?”**

Manolo, buen día. En realidad, no. En las **bases de datos no residen las contraseñas**, ni tampoco encriptadas.

‘‘**¿Es broma? Pero entonces jamás sabrían si mi contraseña que ingresé es correcta.”**

Sí lo hacen. Para eso se usan los **hashes**.

Los **hashes** son el **resultado** de una serie de **operaciones establecidas matemáticas**, que se le aplican a los datos y que son **irreversibles**. Generalmente están compuestos por **valores numéricos** de cualquier base que llegan a obtener la **misma longitud**, en el mismo algoritmo. Y se **intenta** de que sólo puedan generarse hashes **únicos**.

‘‘**Que en español, quiere decir...”**

Jajajajaja, muy bien. En español, tenés alguna palabra, le aplicas una cuenta matemática, y obtenés un **hash** que es **único**. Es decir que si le cambias una letra a la palabra o aunque sea cambias de minúscula a mayúscula y cambia el hash enteramente. Además el hash, sin importar la longitud de la palabra o frase (o archivo), **siempre** va a ser de la **misma longitud**. Y sobre todo, es **irreversible**. Es decir, que teniendo el hash **no** podés **conocer** su verdadero **origen**.

Desde aquí, ellos guardan el hash de tu contraseña y le aplican a tu contraseña que intentas para loguearte el mismo algoritmo utilizado para el hash. Si ambos hashes coinciden, el logueo es exitoso.

Vamos a dar ejemplos para que entiendan verdaderamente de lo que hablo.

Apliquémosle a una palabra, el hash “**md5**”:

hola **---------->**4d186321c1a7f0f354b297e8914ab240

Hola**---------->**f688ae26e9cfa3ba6235477831d5122e

holaa**---------->**8b334cc5ca63a2511b3f1bc3a9a56f98

mi perro se llama jose. viva el rey**.---------->**1db33c95cd541b3af78f812fab382001

archivo.txt(con contenido que no se cual es, dentro)**---->**fe9792ca99ee0c9b856dbf73795f3ddf

Veamos, que no importa la longitud de la frase, el hash **no varía su longitud**.

Otra cosa que podemos observar es que cambiando **una** simple letra, el **hash** varía en su **totalidad**- que en realidad aunque cambie un mínimo bit va a pasar lo mismo. Así que nunca sabremos si estamos cerca de tal o cual valor.

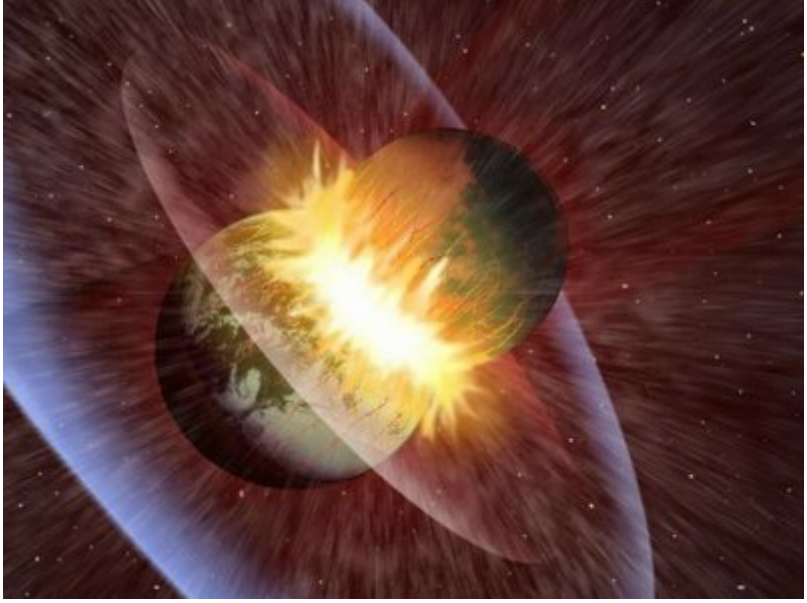
‘‘**Espera, espera. ¿Pero cómo sabés que es realmente único cada hash?”**

Muy buena pregunta. En realidad, nadie esta completamente seguro. Pero **mientras** la teoría **no** sea **refutada** por algún genio, el **hash** sigue siendo **fuerte**.

En realidad, ahora que mencionás esto, el algoritmo md5, no es un hash demasiado fuerte ya que presenta algunas colisiones (ya demostradas). Entonces más de una palabra genera el mismo hash.

Y el **peligro** de esto, es que cuando uno guarda el hash de una contraseña, se pueda entrar no sólo con la contraseña que nosotros habíamos registrado, sino con las mismas que también generan el mismo hash. Además que no sabemos cuales son las que generan la **colisión**.

Digamos que nuestra contraseña es esta: 12hn5t3o1I!·%HI%N, pero el algoritmo que se le aplico es débil y presenta colisiones con: hola. ¿Cuál sería el sentido de teclear una contraseña fuerte si al fin y al cabo es tan débil como “hola”?



“**¿Y para qué otras cosas se utiliza el hash?”**

El hash se usa, también, para **saber** si un archivo es **modificado**. Porque sin importar que tanto hayan cambiado el archivo, el hash va a ser distinto al original siempre.

¿Viste esa cadena de caracteres larga que aparece en algunos lugares de descarga? Bueno eso es el hash para que, vos puedas **comprobar** que es verdaderamente el **archivo** que querés descargar, y **no** uno **modificado** con algún troyano dentro.

Algo que me falto comentar, es que al hash de un archivo, se le llama la “**firma digital”** ya que es única para cada archivo.

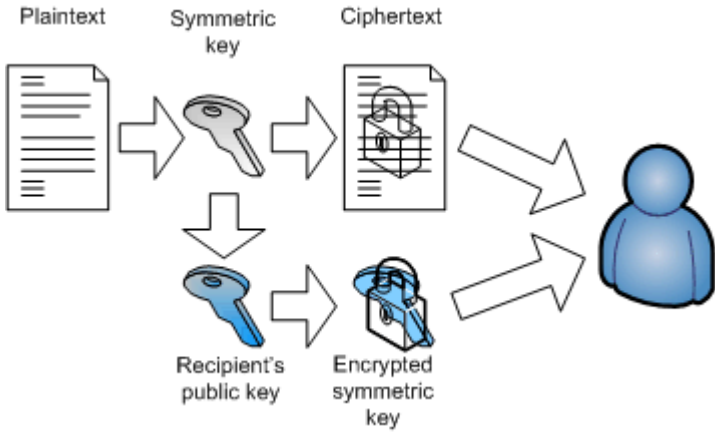
Volviendo atrás al tema de mensajería. ¿Se acuerdan que les comenté sobre la encriptación asimétrica? Voy a declararles que mentí al decir que con la **clave pública,** únicamente se podía cifrar y con la **privada**, únicamente se puede descifrar. Ambas tienen otro tipo de **utilidad**.

Cuando a un archivo o mensaje le aplicas tu **clave privada**, se le genera una **firma digital** y la **clave pública** sirve para **comprobar** esa firma digital. Es decir que con eso si yo te mando un archivo firmada con mi clave privada, cuando le apliques mi clave pública, sabrás exactamente si fui yo el que te mandé aquel archivo.

Esto lleva a una tercera forma de cifrado: **encriptación híbrida**. Ésta es una **mezcla** entre la simétrica y la asimétrica. Veamos como funciona.

Supongamos que un sujeto A quiere hacer llegar de manera segura, un mensaje a persona B.

1. **A**, le aplica a su **mensaje** una encriptación con una **clave simétrica**. Ahora debería pasarle a B, el archivo cifrado y la clave.
2. Para esto, **A** primero le aplica su propia **clave privada** al **archivo** cifrado.
3. Luego, **A** le aplica a la **clave simétrica**, la **clave pública de B**. Entonces esa clave sólo podrá ser vista con la clave privada de B
4. Le **envía** ambos elementos de alguna manera.
5. **B** realiza el proceso inverso. Primero aplicando su propia **clave privada** a la **clave simétrica** para poder leerla correctamente.
6. Le aplica al **mensaje** cifrado la **clave pública de A**, para **verificar** si el contenido no fue modificado.
7. Descifra el **mensaje** con la **clave simétrica** obtenida anteriormente, y así poder leerlo.



Cualquier cosa pueden mandarme mail a: r0add@hotmail.com

Para donaciones, pueden hacerlo en bitcoin en la dirección siguiente:

1HqpPJbbWJ9H2hAZTmpXnVuoLKkP7RFSvw

Roadd.
