Rastreando documentos en Internet.

https://www.youtube.com/watch?v=VdLOkKYLHd8

En la entrada anterior vimos someramente la tecnología TPM a fin de garantizar la seguridad en nuestras comunicaciones pero *¿Sirve acaso esto para hacer seguimiento de nuestra información confidencial, si esta llegare a filtrarse al público?* **Veamos.**

https://twitter.com/chemaalonso/status/1411588696906866692

Como cosa rara (sarcasmo) que nos enteramos del acontecer tecnológico y político por medio de la red social Twitter, llamó nuestra atención este mensaje del jáquer español Chema Alonso. Habla en su entrada en su blog acerca de un servicio que permite determinar a ciencia cierta si la información ha escapado de nuestra empresa.

Aunque se habla allí de "capturar al traidor" recordemos también que virus y troyanos también pueden extraer nuestros documentos valiosos de nuestros ordenadores y/o dispositivos de almacenamiento. Lo cierto del caso es que necesitamos saber si la información que entregamos a una persona natural o persona jurídica ha sido divulgado, ***ese documento en particular***.

https://twitter.com/Snowden/status/1411703748238401536

Tesis acerca de cómo funciona

1.- Un solo fichero y un solo destinatario

Acá nuestra tesis de cómo funciona, que consiste de:

- Bien sabemos que con la tecnología MD5, aplicada a un fichero cualquiera obtendremos obtendremos una huella digital única. **Pero después fue demostrado que existen las** colisiones de *hash* por ello luego fueron inventados muchos otros tipos de cifrado. Acá, por el nombre dado a la aplicación, imagino utilizan SHA, del cual conocemos a la fecha 5 variantes, unas más seguras que otras.
- Si a ese fichero le cambiamos, agregamos o quitamos un simple bit y le calculamos de nuevo su *hash* veremos que ha cambiado completamente.
- Lo importante aquí es que nos quedemos con ese concepto de *hash*: una cadena de caracteres (prácticamente única o única, dependiendo del algoritmo utilizado) que identifican a una archivo.
- La novedad del asunto es prestar un servicio donde nosotros "recibimos" un fichero, calculamos su *hash* (con el algoritmo que mejor nos parezca) y lo almacenemos asociado con el nombre del cliente y el nombre del archivo.
- El proceso descrito en el paso anterior (el cálculo del *hash*) puede ser hecho en la misma máquina del cliente por medio de una aplicación web publicada en HTTPS. **Eso evita (y garantiza) que en ningún momento el fichero sale de manos del cliente, y nos libera de la responsabilidad de haber tocado información alguna con "nuestras manos."**
- Obviamente que dicha aplicación ejecutada por un navegador web, el usuario deberá identificarse en nuestra red, le enviamos un *token* de sesión por correo electrónico, SMS, etc. o usamos un segundo factor de autenticación. **Debemos asegurarnos que nuestro cliente es quien dice ser y debe ser demostrado por algo que tiene, por algo que conoce e incluso ante un tercero que lo asevere.** Dicha aplicación debe tener derecho de lectura sobre el fichero (obvio).

Como bien dice el encabezado de esta sección, eso funciona para un solo fichero enviado a un solo destinatario: si se publica y le aplicamos la fórmula de *hash* y coincide con la que nosotros almacenamos, podremos decirle a nuestro cliente (y demostrar por "fe pública") que efectivamente ese documento pertenece a nuestro cliente y fue difundido sin su consentimiento.

2.- Un fichero y uno o varios destinatarios

Tal como lo leen, este caso incluye uno o más destinatarios, lo que cambiaría yo en nuestra hipotética aplicación web:

- La aplicación web ejecutada por un navegador web por nuestro cliente, en un equipo al cual solo tenga acceso dicho cliente, **debe tener derechos de lectura y escritura en la carpeta donde esté almacenado el documento a enviar, a uno o a distintos destinatarios.**

- Nuestra aplicación web deberá tomar nota de los nombres a quienes les será enviado ese documentos.
- Acá es sumamente importante saber cuál tipo de formato gobierna al archivo: PDF, MP3, JPG, etc. (ya veremos por qué).
- *Como cada archivo produce un único hash (teóricamente), lo que debemos hacer es crear copias del documento con modificaciones aleatorias (por eso* hablé del dispositivo TPM*, si el cliente tiene uno, pues que mejor para introducir modificaciones aleatorias) para cada uno de los destinatarios y guardarlos en el directorio de marras.*
- Desde luego, tenemos que buscar las librerías necesarias para tanto "leer" adecuadamente el contenido del fichero como para su modificación y posterior guardado.
- A cada una de esas copias modificadas ligeramente ("el código invisible") le calcularemos su *hash* correspondiente y lo guardaremos con el nombre correspondiente que el cliente le dio (nombre del destinatario).
- Demás está decir que el cliente se encargará de hacer llegar, de la manera que más le convenga, **dicho documento distinto a cada destinatario distinto.**

3.- Para rastrear elementos multimedia

¿Qué tal si el cliente nos pide que rastreemos imágenes (o audio, vídeo, etc.) dentro de un documento?

Acá se complica un poco el asunto. Un solo fichero puede contener una foto, audio o vídeo, *el fichero completo, quiero decir*. Hasta ahí no tenemos problema alguno para identificar de forma única (teóricamente) un archivo de tal naturaleza (ver primer paso de la tesis). **Pero para un archivo que contenga texto y contenido multimedia *debemos conocer muy bien cómo es la estructura de datos que alberga y cómo los guarda.*** Además, tenemos que tener en cuenta lo que señalamos en el paso 2 de esta tesis de cómo funciona, para cada uno de los diferentes tipos de contenido multimedia.

Repito, para calcular el *hash* de un fichero cualquiera completo no nos importa para nada cómo está escrito y qué representa: solo solo unos y ceros a ser calculados por el algoritmo. Si es para varios destinatarios, pues ya vimos que sí, necesitamos saber cuál formato es al archivo (SVG, PNG, TXT, etc.)

Para los elementos multimedia en un fichero (según el formato que debemos conocer), deberemos extraer en un archivo temporal (en el directorio donde el cliente guarda sus documentos confidenciales) **todos y cada uno de estos elementos multimedia, modificarlos ligeramente respetando el formato que lo gobierna, calcular su *hash* y asociarlo con el hash principal que habremos calculado en pasos anteriores.**

Pero eso abre otra incertidumbre: ¿Cuál destinatario extrajo un elemento multimedia y causó la fuga de información?

Acá lo que debemos hacer, de nuevo, es repetir el ciclo: crear una copia ligeramente modificada de todos y cada uno de los elementos multimedia y volverlos a guardar en una nueva copia ligeramente modificada del archivo contenedor (documento completo). Luego, con el texto que le acompaña, volver a calcular el hash del documento ligeramente modificado en sus elementos multimedia y en su texto. ***Por que sí, de seguro algo de texto tendrá el documento, las cadenas de caracteres (texto plano o enriquecido) estarán rodeando todo ese contenido multimedia: veamos el siguiente paso.***

Elementos de texto

Un texto está conformado por palabras que están delimitadas por espacios o signos ortográficos (no todos los idiomas cumplen con esto, el idioma japonés no separa las palabras). Acá lo que nos interesa sería crear *hash* completos **de texto plano** de cada uno de los párrafos e incluso de cada una de las oraciones. De las palabras creo que ninguna.

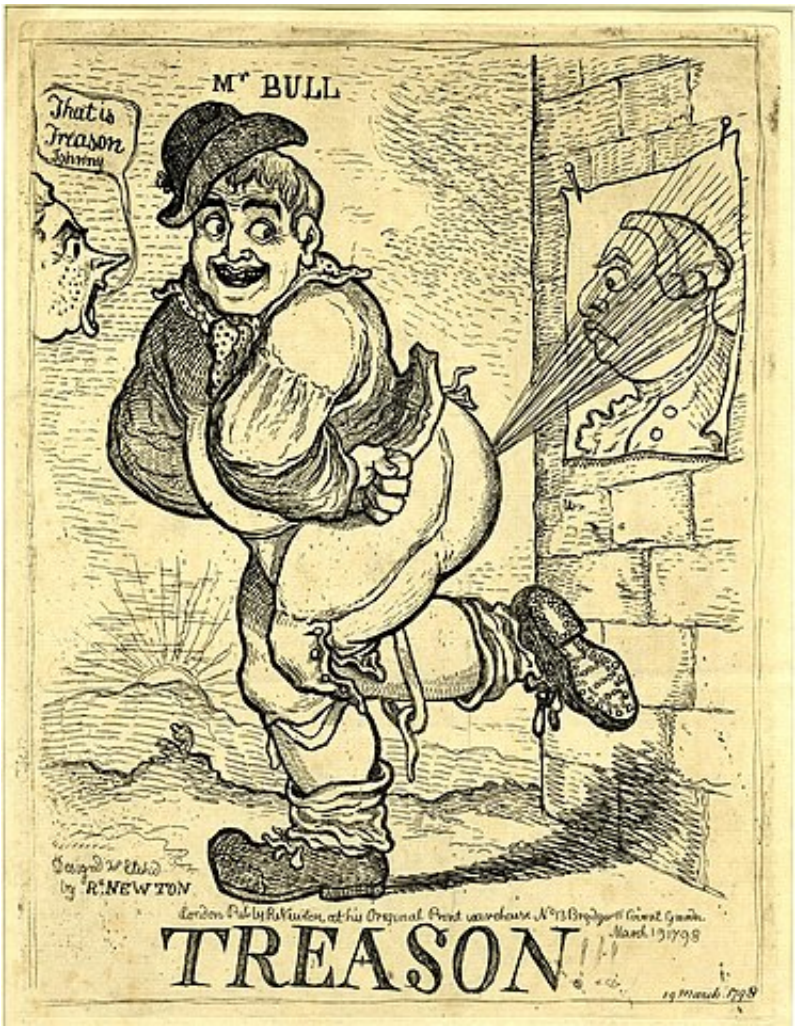
Lo anterior tiene unas cuantas excepciones que se deben tener en cuenta: si un documento cita otro documento (por ejemplo un contrato de trabajo cita la ley del trabajo respectiva) *esas cadenas de texto nunca podrán ser usadas para rastrear al documento*. La buena noticia es que siempre estos documentos legales hacen mención de datos personales que en conjunto forman una huella única. Deberemos entonces prestar atención a esos detalles en el supuesto que debamos ofrecer *fe pública* de dicho documento, por demás todo funciona de la misma manera

Despedida

Este artículo va publicado de forma didáctica, no intenta hacer ingeniería inversa ni busco de aprovechar de forma monetaria alguna (de hecho está publicado najo licencia Creative Commons, ver pie de página web).

Tampoco estamos asociados a esa empresa que ofrece Software como Servicio (SaaS) ni guardamos relación alguna con el señor Chema Alonso más allá de los intereses comunes acerca de la programación y el aprendizaje, la búsqueda del conocimiento. Tengan todas y todos un muy

buen día.



Treason to George III King of the United Kingdom (© The Trustees of the British Museum, released as CC BY-NC-SA 4.0)