El Papel Digital*

Juan F. Codagnone
juam@ {arnet.com.ar, users.sourceforge.net}
Revisión: 1.0

Noviembre 2002

Índice

1. Introducción

El desarrollo de Internet estuvo marcado por tecnologías que no contemplaban la necesidad de garantizar la privacidad e integridad de la información que por ella viaja. Hoy en día, por ejemplo, en un correo electrónico (cuya estandarización data de fines de la década del setenta) un agente puede falsear el campo del remitente de forma muy simple[1]. Las computadoras personales llegaron a las oficinas y las montañas de papeles, pasaron a ser papeles virtuales. La automatización permitió agilizar trámites, pero se perdió la posibilidad de validar (autentificar) estos papeles digitales. Un borrón en una hoja para el usuario promedio es más fácil de detectar que un cambio en un par de ceros y unos.

El objetivo de este artículo es introducir los conceptos y las herramientas necesarias para que quien maneja documentos digitales lo pueda hacer con la misma comodidad con que lo hacía cuando trabajaba con sus papeles. En la primera parte se presentan varias nociones sobre los sistemas criptográficos necesarios, para luego describir usos de éstos en la oficina digital.

2. Problemas

Por un momento piense en el siguiente manejo de stock de un laboratorio de electrónica: los alumnos piden componentes y elementos en el laboratorio pero para comprar nuevos componentes (en caso que no queden en stock) se requiere autorización de algún profesor. Entonces si el alumno pide un componente que no existe debe enviarle un mail a su profesor justificando su uso, y si éste lo considera adecuado le enviará al encargado del material la autorización, también por correo.

La forma de trabajo es versátil y cómoda: se puede autorizar aún no estando en la facultad, y se ahorra tiempo. Sin embargo, un alumno aburrido,

[1] Pregúntese si alguien verificó que usted era realmente quien decía, y que su dirección de correo fuera cierta, la última vez que configuró su programa favorito para manejar correos electrónicos.

y con pocos escrúpulos puede producir la compra de material innecesario escribiendo simplemente:

```
$ telnet smtp.universidad.edu.ar 25
Trying 192.168.159.1...
Connected to smtp.universidad.edu.ar
Escape character is '^]'.
220 smtp.universidad.edu.ar ESMTP
MAIL FROM: profesor@universidad.edu.ar
250 Ok
RCPT TO: deposito@universidad.edu.ar
250 Ok
DATA
354 End data with <CR><LF>.<CR><LF>
To: deposito@universidad.edu.ar
Subject: Autorizacion compra material

Autorizo la compra de 3 soldadores.
.
250 Ok: queued as E9AE211097
quit
221 Bye
Connection closed by foreign host.
```

Si bien ésto parecerá extraño al lector iniciado, recuerde que ningún programa puede chequear la correctitud de los datos que usted ingregesa cuando configura su cuenta de correo. Si el ejemplo le parece absurdo, entonces piense en un administrador enviándole mensajes a sus usuarios (ejecute este programa, envíe ciertos archivos, ...).

3. Encriptación

La encriptación[2] es la acción de convertir un mensaje (cualquier tipo de dato) en otro, con la intensión de que sólo unos pocos puedan obtener el mensaje original. ¿Para que se utiliza? Suponga que Alicia recibió una carta de amor de su amante, y la quiere guardar de forma tal que otros ojos no puedan leerla. Si el mensaje estuviera en una hoja, Alicia la podría guardar en una caja con candado; pero debido a que este Romeo es del siglo XXI, Alicia recibió la carta en un archivo. La forma que tiene Alicia de esconder el mensaje, es transformarlo en otro mensaje que signifique algo diferente.

Existen diferentes forma de encriptar datos.

3.1. Tipos de Encriptacion

3.1.1. Simétrica

Supongamos que Alicia cambia cada letra de su carta siempre por una misma letra. Julio Cesar aparentemente utilizaba esta forma de cifrado para comunicarse con sus generales. Sólo quien conozca el truco decifrará el mensaje, y esto lo hace vulnerable. Imagine que un general del Cesar en un acto de repudio a su política, o en un acto de borrachera, revela la forma de leer estos mensajes. Entonces los mensajeros ahora podrían leer estos mensajes, y una nueva forma se debería crear. Además, este método es sensible a la información que se oculta: Sabemos que las cartas formales empiezan con frases como '*Me dirijo a usted*', o que ciertas letras suelen aparecer más veces que otras. Una forma de resolver estos problemas, es combinar en alguna operación cada bloque (o letra) del mensaje a encriptar con letras (o bloques) de otro texto. A este texto extra se lo llama *llave*. Con la llave, el método de codificación puede ser conocido por agentes hostiles, pero sólo quien conozca la llave (ej: un párrafo del Quijote) puede leer el mensaje. La desventaja que presenta este método es que la llave debe ser entregada de forma confidencial al destinatario. Esto crea problemas de logística. No se puede enviar la llave por un mensajero, porque no se sabe si éste se la mostrará a terceras personas. De aquí la simetría.

Los algoritmos de encriptación hoy en día más usados son:

DES utilizados alrededor de los años 80 por el gobierno norteamericano para datos no clasificados, hoy en día es débil debido al tamaño de la llave y del bloque.

3DES tres encriptaciones DES con 3 llaves diferentes. Hoy en día es un método todavía fuerte, pero a expensa de tiempo de procesamiento.

AES que es el nuevo método del gobierno norteamericano para remplazar el uso del DES,

Blowfish libre de patentes,

[2] el término encriptación no existe en el diccionario de la Real Academia Española. El término que se suele usar es cifrado. En este articulo se utilizaran indistintamente los términos derivados de encriptación como de cifrado.

IDEA se deben pagar regalías para uso comercial,

RC4 cuyo algoritmo vive en el dominio publico.

3.1.2. Asimétrica

En el año 1976 Diffie y Hellman [2] inventaron la *Infraestructura de la llave publica* (PKI). Este sistema permitió el cifrado y la firma de mensajes (éste último tema se verá en la sección 3.2). La diferencia fundamental con los métodos que se mencionaron anteriormente es la eliminación de la distribución de las llaves: no se necesita un medio seguro para transmitir la llave. Las comunicaciones pueden darse sobre un medio donde otras personas personas vean lo que se transfiere.

El trabajo de Diffie se basa en la existencia de dos llaves para cada persona: una *llave pública* o llave de encriptación (E), y una *llave privada* o llave de desencripación (D). Cada par de llaves está relacionada, pero conocida una no se puede sintetizar la otra.

Entonces suponga que existen dos usuarios (Alicia y Bob) que desean comunicarse de forma confidencial por Internet. Alicia tiene un par de llaves E_A y D_A, y Bob E_B, D_B. Recuerde que E_A y E_B son llaves públicas, y están disponibles a todo el mundo. Entonces, para realizar la comunicación, Alicia cifra el mensaje utilizando la llave publica de Bob (E_B). Dado que Bob es la única persona que conoce la D_B, el sólo podrá entender el mensaje.

Usted se preguntará cómo se produce un par de llaves para no se pueda obtener desde de la llave publica, la privada. En el año 1977 Rivest, Shamir y Adelman [6] inspirados en el trabajo de Diffie publicaron un trabajo donde describían un problema numérico que se amoldaba a esta situación. Este problema se relaciona a la enorme dificultad de factorear números.

¿Cuan fuerte es este problema? La resolución del mismo en el tipo de máquinas que hoy en día se usa, necesita tiempo exponencial. Sin embargo se ha encontrado una forma de factorear números en tiempo acotado [7] (polinomial para ser correcto) en las llamada computadoras cuánticas. Esto no es un grave problema debido a que no se espera tener una de estas con suficiente capacidad de procesamiento hasta un par de décadas [!!!]. Aún cuando éstas tomen la delantera, las propiedades del los fotones aparentemente permitirán una encriptación imposible de violar [3].

Las desventajas de esta forma de cifrado frente al simétrico, son

- la necesidad de realizar mayor cantidad de operaciones matemáticas para realizar una misma tarea debido a la separación de llaves;
- la necesidad de proteger el acceso a llave privada. Para esto se suele usar métodos simétricos (aplicación que no tiene el problema de la distribución de llave ya que la llave debe estar sólo en poder de dueño de la llave privada).

Una gran ventaja de las asimétricas, es que si usted tiene n contactos con los cuales se va a comunicar, usted sólo necesita un único par de llaves. Si deseamos comunicarnos con estos n contactos de forma simétrica, debemos producir n llaves y distribuirlas de forma confidencial.

Además, la *infraestructura de la llave pública* introduce el concepto de firma digital, tema que se tratará a continuación.

3.2. Firma

Suponga que Usted acaba de transferir un archivo entre dos computadoras y quiere verificar que ambos archivos son idénticos. Algunos intentarían volver a transferir el archivo y compararlos. Éste proceder no es descabellado, pero es muy costoso (en tiempo y en dinero) y nadie garantiza que los mismos posibles errores de transmisión ocurran nuevamente (si es que realmente tiene la posibilidad de retransmitirlo). Entonces lo que se suele hacer, es leer de a bloques cada archivo por su cuenta, y aplicarles a ambos una serie de operaciones previamente definidas para obtener un número. Esta serie de operaciones definen una función *one-way-hash*[3] o un *message digest*[4] , es decir, una función sin inversa. Es por esto que los valores de la función estarán asociados a varias secuencias de datos diferentes. Este camino también es probabilista, pero no necesita de otra copia del mensaje.

Los algoritmos más comunes para generar estos números son MD4[4], MD5[5] y SHA. MD4 y

[3] La frustrada lectura de libros originalmente escritos en inglés, traducidos al español (ya sea en México o en España), enseña que no se deben inventar términos españoles para cada palabra inglesa.

[4] Algo así como *forma reducida del mensaje*.

MD5 generan un número de 128 bits, mientras que SHA genera uno de 160 bits. Tanto MD5 como SHA derivan de MD4 , y solucionan diferentes ataques que eran posibles sobre MD4.

De esta forma, se puedo resolver el problema de saber si un dato (mensaje/archivo) que se obtuvo sufrió algún cambio durante su transferencia, pero no se tiene información para autentificar su origen. Suponga que Bob le envía a Alicia un mensaje, junto a su valor MD5. Alicia puede chequear que el mensaje no ha sido cambiado, pero no sabe si realmente Bob fue quien creó el mensaje ya que todo el mundo puede generar valores MD5(todo el mundo debe saber como calcularlos para poder verificar sus archivos). El mensaje pudo haber sido editado, y el valor MD5 regenerado. Para poder autentificarlo entonces se requiere que el message digest dependa tanto del mensaje como del emisor. Bob entonces crearía una message digest con el mensaje y su llave privada, de forma tal que este número pueda ser verificado con su llave pública. Los messages digest que salen de funciones de esta manera se las llama *firmas* y poseen las mismas propiedades que las firmas holográficas. Una firma digital, es más dificil de falsear, que una holográfica, la digital por su naturaleza, corre más peligro de ser destruida.

3.3. Revocaciones

Cuando su tarjeta de crédito queda comprometida, ya sea por robo físico de la tarjeta, extravío, o porque otra persona conoce su número, usted la revoca indemediatamente. Si su llave privada es comprometida, usted debe tener alguna forma veraz de revocarla, de decirle a los otros poseedores de llaves, que su llave ya no es válida y que no puede ser confiada. Según el sistema PKI, existen listas públicas de revocaciones, o uno simplemente se entera. Si no hay una autoridad central que maneje las listas de revocaciones entonces es importante tener generado una revocación en un lugar aparte. Si usted pierde su llave publica, entonces podrá revocarla.

4. Confianza

Un mensaje, no es lo único que se puede querer validar. ¿Existe alguna forma de demostrar que una llave del usuario X corresponde a un persona física? ¿Quien está del otro lado del cable es realmente Alicia?

Una opción es llamar por teléfono a Alicia (si es que le conoce la voz) o concertar una cita y pedirle los detalle de la llave y un documento de identidad. Esto no es realmente cómodo. Validar las llaves es costoso (los dueños de los bares contentos). Una solución es que exista una *Autoridad Certificante* (CA) quien firme todas las llaves públicas que sabe que coinciden con las personas físicas (o agentes). Entonces las llaves se distribuyen firmadas por el CA. Ahora, Alicia puede bajar la llave de Bob desde Internet (medio hostil), y chequear que sea válida, la que firma que incrustó el CA. Si lo es, entonces Alicia sabe (si es que confía en el CA) que esa llave es del Bob que ella conoce y no de un impostor. De esta forma, cada miembro de la comunidad, debe realizar solamente una única validación física.

Pero vivimos en un mundo globalizado, y distribuido, donde lo centralizado es costoso. El modelo de un único CA funciona bien en un entorno pequeño. Por ejemplo, en una universidad el secretario académico puede actuar como CA y firmar las llaves de profesores y alumnos. En comunidades mas grandes, se puede tener un CA raiz, que jerarquicamente delege atribuciones y reconocimientos a otros CA's.

¿Qué sucede si un CA pierde la cordura y comienza a firmar cosas que son mentiras? Todas los contactos realizados hasta la fecha de locura son válidos.

Nada evita que los participantes actúen como CA. Entonces se va produciendo la *Red de Confianza*[5]. En esta red de confianza la promiscuidad de las personas hacen débil (o no!) ciertas partes de la red. Por esto, si usted planea participar en una comunidad, donde la forma de autentificación el mediante este sistema, es importante planificar su presencia en la red. Es conveniente que en esta planificación, haya una cantidad de caminos bidireccionales alrededor de su llave, para aumentar la cantidad de gente a la que puede llegar, y disminuir la incertidumbre.

Por lo dicho antes, lo que llamabamos llave pública, ahora también puede cumplir la función de Documento de Identidad (autentificación): Si quiere convencer a alguien que está del otro lado del cable de que usted es quien dice ser, envíele un texto firmado con su llave. Si el incrédulo puede trazar un

[5]Web of Trust

camino por la red de confianza y llegar a su llave, entonces sabrá que usted es esa persona, ya que sólo una persona puede firmar un mensaje con esa llave, y otras personas certifican la valides de la llave.

5. Dificultades

Al usar estas tecnologías, hay que conocer sus limitaciones y posible problemas. Para los sistemas que se han descrito, siempre se necesita que se confíe en los lugares donde se cifra y firma (computadora?). Por ejemplo, PKI depende de que la llave privada se mantenga en secreto. La matemática de estas cosa, suele ser compleja, pero el mecanismo es simple: Alicia desea firmar un documento, ingresa la clave que protege simétricamente su llave privada, obtiene su llave privada, y el programa entonces la usa con el documento parar generar una firma que pueda ser chequeada con su llave publica. Matemáticamente todo cierra, pero semanticamente no: Cuando usted firma algo en la vida real, se tiene un documento en la mano que esta siendo firmado, pero aquí quien realmente firma es la máquina. Es un acto de fe o confianza creer que el programa no va a enviarle la llave privada a alguna otra persona.

Otra dificultad es, que no se suele comentar en los manuales de las aplicaciones de estos sistemas, es el problema de la *Firma y Encriptación*[1]. Recuerde que que para encriptar se utiliza las llaves de los destinatarios, y nada de los emisores. Entonces suponga que Alicia le envía un mensaje a Bob firmado y encriptado diciendo '*Te amo!*', y Bob, enojado decide jugarle una broma y encripta el mensaje para Carlos. Carlos recibirá el mensaje encriptado, y verá la firma de Alicia, y creerá en el mensaje:

$$A \rightarrow B : \{\{\text{"}Te\ amo!\text{"}\}^a\}^B$$
$$B \rightarrow C : \{\{\text{"}Te\ amo!\text{"}\}^a\}^C$$

¿ El problema ? No hay indicadores de quien encriptó el mensaje, o algún número alguna referencia al mensaje. Vale aclarar, que no todos los sistemas que se basan en PKI son vulnerable a estos ataques.

6. Usos

6.1. TLS, SSL, SSH

Internet crecía, y el comercio electrónico lo hacía con ella; pero todas las comunicaciones podían ser interceptada por muchas personas. Entonces Netscape, implementó el *Secure Sockets Layer*, que entre otras cosas tenía características como la autentificación mutua, y la encriptación. Gracias a la generalidad de SSL hoy cualquier protocolo puede ser enmarcado en una conexión SSL, desde el envío y recepción de sus correos electrónicos, como las paginas que navega por http. Sin embargo, en la Argentina, ningún proveedor de Internet ofrece conexiones seguras (¿momento de exigirlas?).

Otro uso, es en la autentificación de usuarios. Suponga que Alicia desea iniciar una sesión en una computadora de su empresa. Alicia le dice al servicio que ella es Alicia y que desea ingresar. El sistema conoces la llave pública de Alicia, y encripta una mensaje a esa llave. Si el usuario es realmente quien dice ser, entonces podrá leer el mensaje, y enviárselo al servicio. Este desafío, suplanta al uso de contraseñas.

6.2. Ley 25506

La ley 25506 promulgada el 11 de diciembre de 2001 en Argentina, '*... reconoce el empleo de la firma electrónica y de la firma digital y su eficacia jurídica en las condiciones que establece la presente ley*'. Establece que la firma digital puede ser utilizado en todos los lados que la ley requiera una firma holográfica, excluyendo a actos jurídicos por causa de muerte, del derecha de familia, actos de personalismo en general. Las llaves utilizadas, deben ser certificadas por la Autoridad Certificante Licenciadas del Gobierno. Además, *... la exigencia legal de conservar documentos, registros o datos, también queda satisfecha con la conservación de los correspondientes documentos digitales firmados digitalmente, según los procedimientos que determine la reglamentación.*

La ley, no prevee nada de lo mencionado en sección 5 de éste artículo, pero la ley es una buen camino para acelear la burocracia, los trámites de ministerios y secretarías. A mi entender, el público general no está listo para usar esta ley, pero si lo están los sector ligados al estado, como los bufetes

de abogados.

6.3. Fechado digital

El fechado digital es otra utilidad de PKI. La entrega de trabajos que existen únicamente en forma digital es algo en algunas facultades (por ejemplo sistemas y diseño). En general estos trabajos tiene una hora tope de presentación del material. Se puede crear con facilidad un sistema donde los alumnos envíen sus trabajos (firmados) y este sistema agregue al mensaje una estampilla con la hora y se lo envíe al docente, quien deberá generar un mensaje de confirmación y firmarlo. El alumno entonces posee una prueba de que entregó el trabajo (nunca tuve una prueba de que realmente entregué un trabajo), el profesor también posee una prueba que el trabajo entregado es de un alumno particular (el alumno no puede alegar que fue modificado para perjudicarlo), y además se tiene la fecha de entrega. Nada de esto es necesario si se actúa de buena fe, pero puede ahorrar problemas de forma simple.

7. Conclusión:

Se presentaron los conceptos necesarios para manejar papeles digitales, de forma confiable, pero queda para otro articulo presentar diferentes implementaciones y sistemas criptográficos que pueda emplear.

Referencias

[1] Davis, D. Defective sign encrypt in S/MIME, PKCS7, MOSS, PEM, PGP, and XML.

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

[3] Mullins, J. Making unbreakable code. *Spectrum of IEEE* (May 2002), 40–45.

[4] Rivest, R. L. The MD4 message digest algorithm. Tech. rep., IETF Network Group, October 1990.

[5] Rivest, R. L. The MD5 message-digest algorithm. Tech. rep., IETF Network Group, April 1992.

[6] Rivest, R. L., Shamir, A., and Adelman, L. M. A method for obtaining digital signatures and Public-Key Cryptosystems. Tech. Rep. MIT/LCS/TM-82, 1977.

[7] Shoar, P. W. Polynomial-time algorithms for prime factorization and discrete logarithms on a quantum computer. *SIAM Journal of Computing 26* (1997), 1484–1509.