Seguridad de los Sistemas Informáticos

Tema 3. La Criptografía y sus aplicaciones

Escuela de Ingeniería Informática
Universidad de Oviedo

Objetivos de aprendizaje

- Conocer los principios teóricos básicos de la criptografía.
- Ser capaz de conocer las distintas aplicaciones criptográficas que contribuyan a la seguridad de los sistemas informáticos.

Introducción

El objetivo principal de un sistema seguro es la protección de la información. Una forma de protegerla es mediante la ocultación. Hay dos técnicas para ocultar información:

- *Esteganografía*: ocultación de la **existencia** de la información (tinta invisible, micropunto, información incrustada en imágenes, etc.)
- *Criptografía*: ocultación del **significado** de la información. La información se transforma mediante un proceso de *codificación o cifrado* en otra cifrada, totalmente ilegible.

Esteganografía

Fichero contenedor

Información a ocultar

```
Este gatito es menos
inocente de lo que
parece. De hecho, está
ayudando a transmitir
un mensaje oculto.
```

Contraseña

Clave

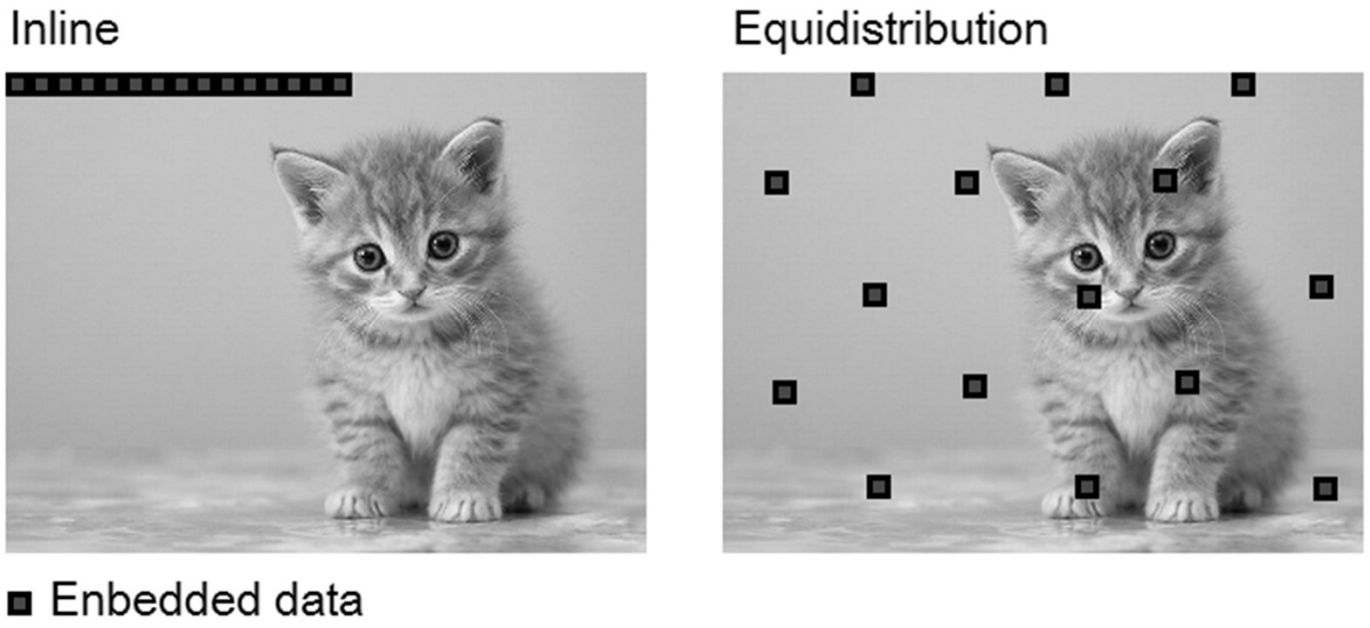
proceso esteganográfico

Esteganograma

Esteganograma: ¿Cómo se genera? (I)

1. **Sustitución** de bits en el objeto contenedor

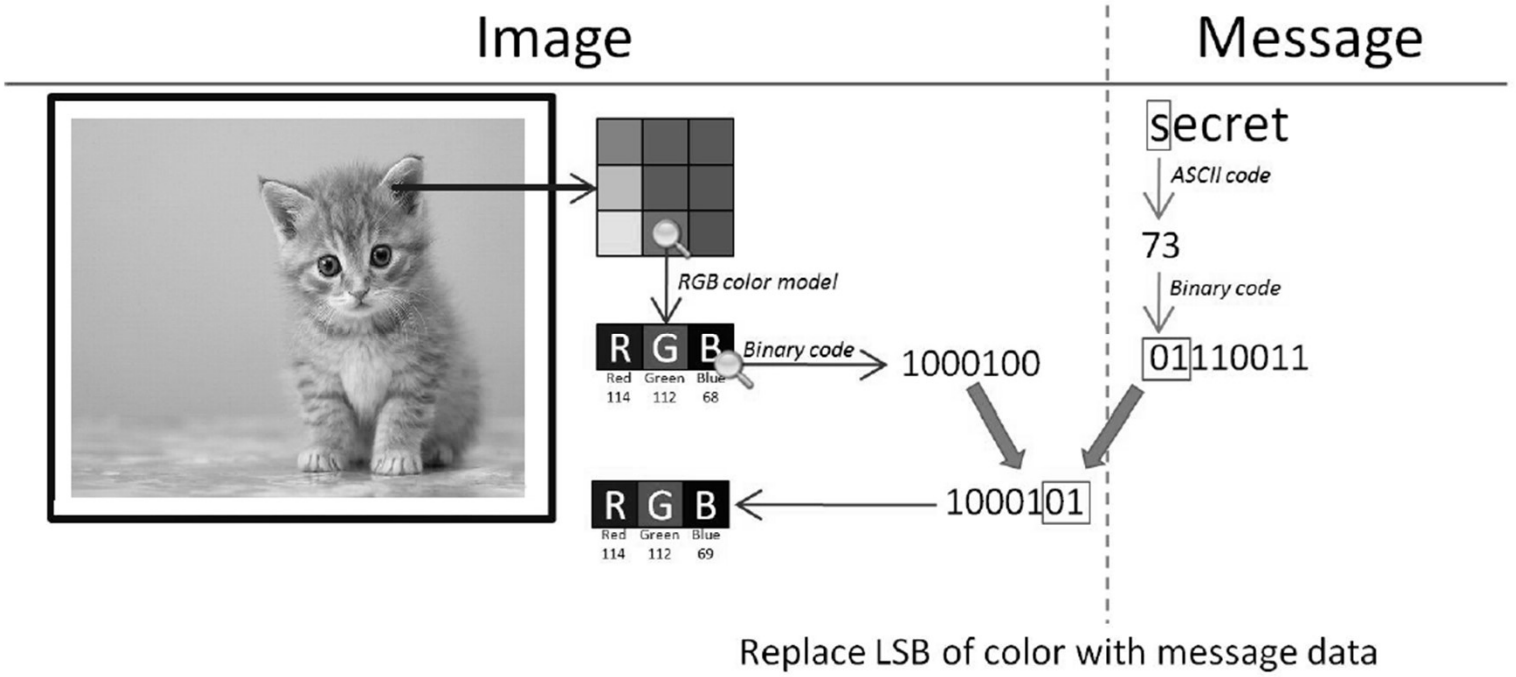
Consiste en sustituir ciertos bits del fichero contenedor por los correspondientes a la información a ocultar. La ventaja de este enfoque es que el tamaño del fichero contenedor no se ve alterado y, gracias a la redundancia y/o exceso de detalle en dichos ficheros, en la mayoría de las ocasiones tampoco su calidad.



Fuentes material original: www.incibe. es, www.silenteye.org

Esteganograma: ¿Cómo se genera? (II)

Caso de imágenes: algoritmo de sustitución LSB (Least Significant Byte).

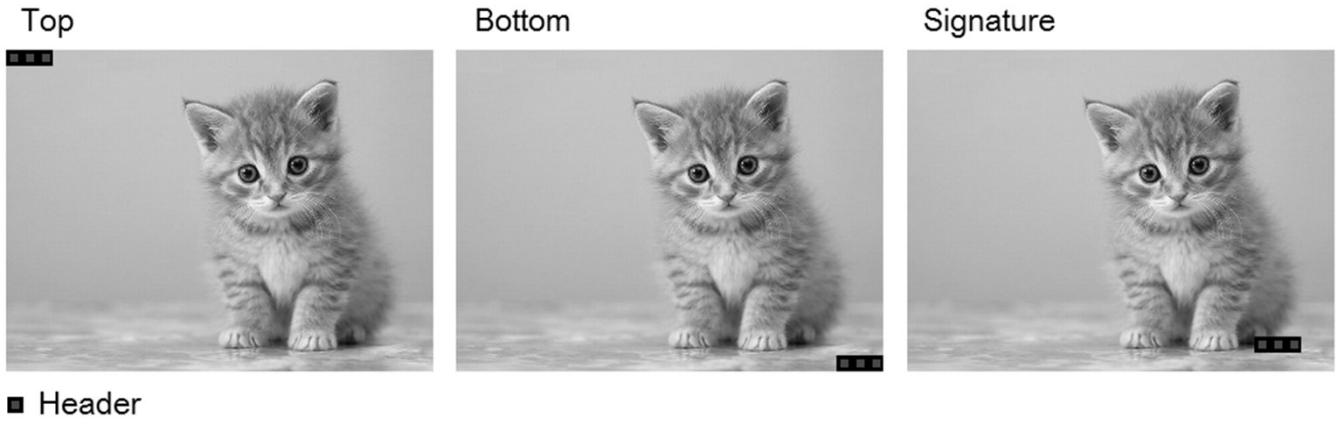


Fuentes material original: www.incibe. es, www.silenteye.org

Esteganograma: ¿Cómo se genera? (III)

2. **Inserción** de bits en el objeto contenedor

En este caso se añaden los bits de información a partir de una determinada marca estructural del fichero (EOF, espacios de padding o alineamiento, metadatos, etc). Esta opción presenta el inconveniente de que sí se modifica el tamaño del objeto contenedor, con lo cual puede levantar sospechas (si los datos a ocultar son considerablemente grandes, sería sospechoso tener un icono de 16x16 píxeles que ocupe 5 MB, por ejemplo)



Fuentes material original: www.incibe. es, www.silenteye.org

Generación de esteganogramas

- Hay muchas herramientas generadoras de esteganogramas:
  - steghide
  - Steganos Security Suite
  - Steganography
  - Outguess
  - Stegtools
  - SilentEye
- La primera está disponible en ubuntu:

```
# sudo apt-get install steghide

$ steghide embed -cf imagen.jpg -ef mensaje.txt

$ steghide extract -sf imagen.jpg
```

Esteganografía + Criptografía

- Para que la esteganografía sea de más utilidad se **combina con la criptografía**. El mensaje a intercambiar se ha de cifrar (de forma robusta) y luego introducir en el objeto contenedor. De esta forma, aunque un interceptor descubra el patrón esteganográfico, no puede llegar a conocer el mensaje intercambiado.
- La combinación de estas dos técnicas tiene otra ventaja adicional: cuando se emplea la criptografía en solitario se conoce que se están intercambiando mensajes, lo cual puede servir como punto de partida para un ataque con el fin de descubrir dicho mensaje. Al introducir la esteganografía, en una gran mayoría de casos ni siquiera se conoce que existe una comunicación cifrada.

Criptografía: Introducción

- **La criptología** es la ciencia que trata los problemas relacionados con la seguridad en el intercambio de mensajes en clave entre un emisor y un receptor a través de un canal de comunicaciones.
- Está dividida en dos grandes ramas:
  - **Criptografía**, disciplina que se ocupa del cifrado de mensajes en clave y del diseño de criptosistemas, y
  - **Criptoanálisis**, que trata de descifrar los mensajes en clave, rompiendo así el criptosistema.
- Tiene una larga historia (más de 2000 años). Su función básica ha sido y es el aseguramiento de las comunicaciones (diplomáticas, militares, etc).

Finalidad de la criptografía

- Actualmente la criptografía es una herramienta básica en la seguridad de los sistemas informáticos.
- La criptografía contribuye a conseguir:
  - *Confidencialidad*. Sólo los que sepan cómo descifrar la información pueden conocerla. Aplicable tanto a comunicaciones como a información almacenada.
  - *Integridad*. Verificación de que unos datos no se han cambiado. Suelen usarse funciones hash (MD5, por ejemplo).
  - *Autenticación*. Verificación de que alguien es quien realmente dice ser (origen de e-mail, máquinas, etc). Suelen usarse las *firmas digitales* y los *certificados*.

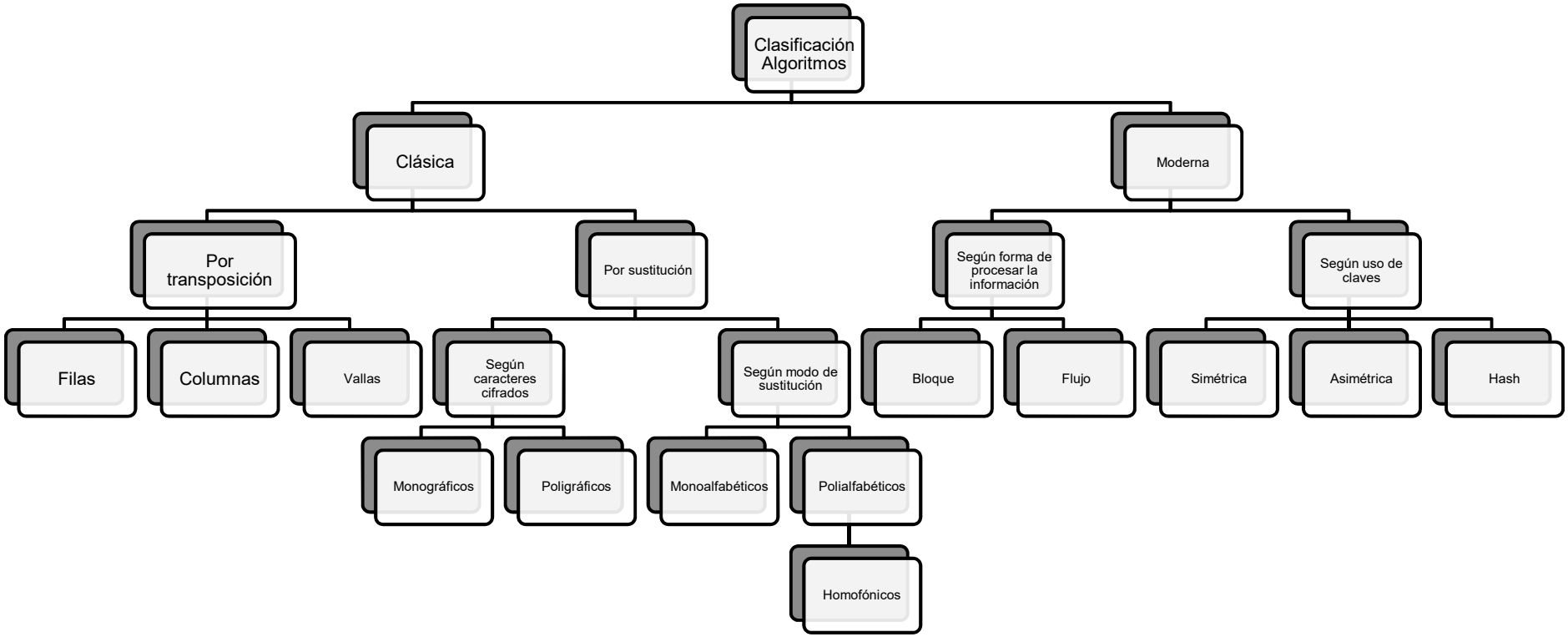
Algoritmos de cifrado

Son las funciones de cifrado y descifrado que transforman un objeto mediante la aplicación de procedimientos que permiten ocultar el contenido del mismo o volverlo a su forma original.

Terminología

- *Texto plano*: información original.
- *Cifrado*: proceso por el cual se transforma el texto plano en algo no entendible.
- *Texto cifrado, criptograma*: información tras el proceso de cifrado.
- *Criptoanálisis*: intento de obtener la información original a partir únicamente del criptograma.

Algoritmos de cifrado: clasificación



Clasificación “clásica” (I)

Transposición o Permutación:

- Reordenan la estructura del objeto de forma que unidades de texto se cambian de posición siguiendo un esquema definido.
- Se **mantienen todos los caracteres** del texto original, pero en distinto orden.
- Tipos:
  - Por filas
  - Por columnas
  - Por vallas (Rail Fence)

  } Se puede añadir fortaleza al cifrado incorporando una clave

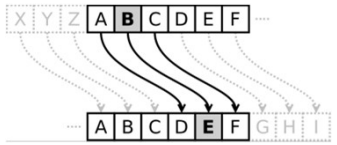
- Utilizados en los algoritmos **DES** (Data Encryption Standard) y **AES** (Advanced Encryption Standard)

Clasificación “clásica” (II)

Sustitución

- Cambian parte del texto original por otros textos. Esta técnica conserva el orden de los caracteres del texto original pero **los caracteres son diferentes**.
- Tipos:
  - Según los caracteres cifrados:
    - Monográficos: el cifrado se hace carácter a carácter.
    - Poligráficos: se cifran grupos de caracteres
  - Según el modo de sustitución:
    - Monoalfabéticos: cada carácter se sustituye por otro símbolo que siempre es el mismo. Fácilmente atacable mediante análisis de frecuencia.

      Ej: la cifra del César



    - Polialfabéticos. Cambian cada carácter por otro dependiendo de la posición en el mensaje original utilizando diferentes alfabetos. Ejemplo: Cifrado de Vigenère (26 alfabetos)
      - Homofónicos. Tipo especial de algoritmo polialfabético. Sustituye cada letra por un código de una lista de códigos asociados (para evitar el análisis de frecuencias)

Clasificación “moderna”: según la forma de procesar la información

- En los algoritmos modernos no se trabaja a nivel de carácter, sino que se manipula la información almacenada en forma de **bits y bytes**.
- Desde este punto de vista, los sistemas se clasifican en:
  - Cifrado en **bloque** (IDEA, AES, RSA ...). 64, 128 bits
  - Cifrado en **flujo** (A5, RC4, SEAL ...). Cifrado bit a bit

Clasificación “moderna”: según uso de claves

- Algoritmos sin clave
- Algoritmos con clave
  - Algoritmos Simétricos
  - Algoritmos Asimétricos
- Algoritmos de una sola dirección (*funciones Hash)*

Principios de Kerckhoffs

- Si el sistema no es teóricamente irrompible, al menos debe serlo en la práctica.
- **La efectividad del sistema no debe depender de que su diseño permanezca en secreto.→ La importancia de la clave**
- La clave debe ser fácilmente memorizable de manera que no haya que recurrir a notas escritas.
- Los criptogramas deberán dar resultados alfanuméricos.
- El sistema debe ser operable por una única persona.
- El sistema debe ser fácil de utilizar.
- *Enunciados en 1883*

Clave

- Patrón que usan los algoritmos de cifrado y descifrado para manipular los mensajes en ambos sentidos.
- Añaden más seguridad a los algoritmos de cifrado
  - Distintas claves-> distintos mensajes cifrados con el mismo algoritmo
- La fuerza de una clave se indica a través de los bits o dígitos de longitud. Una clave de 3072 bits es mucho más compleja y difícil de romper que una clave de 128 bits, aunque también requiere una mayor potencia de cálculo.
- Hay que recuperar el objeto original cifrado O

  Objeto O = Descifrado(Clave K, Cifrado(Objeto O, Clave K))

Algoritmos simétricos

Simétricos (o de clave privada)

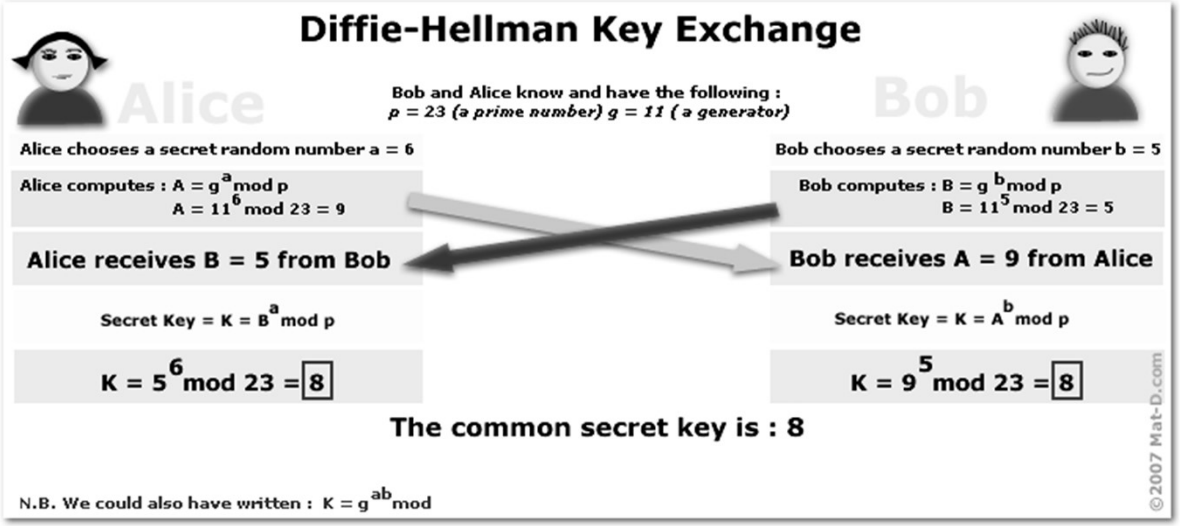
- Emisor y receptor comparten la misma clave K
- La función de Descifrado es la inversa de la de Cifrado
- Problemas:
  - La distribución de claves: el remitente y el destinatario se tienen que poner de acuerdo en la clave a emplear.
  - La seguridad del sistema depende de que nadie más conozca la clave. Esto tiene el problema de que hay que transmitirla al receptor en algún momento y puede ser interceptada.

Ejemplos de algoritmos simétricos

- **DES** (Data Encription Standard) (años 70). Basado en el sistema Lucifer de IBM limitado a claves de 56 bits, divide el mensaje en grupos de 64 bits y para cada bloque traspone los bits en función de la clave 16 veces. Hoy no se considera seguro. Una mejora es el Triple DES.
  - El sistema de cifrado *DES* se actualizaba cada 5 años. En el año 2000, durante su última revisión y después de un proceso de evaluación que duró 3 años, el *NIST* (Instituto Nacional de Estándares y Tecnología) seleccionó como nuevo estándar un algoritmo diseñado conjuntamente por dos candidatos belgas, *Vincent Rijmen* y *Joan Daemen*. El nuevo algoritmo, llamado por sus inventores **RIJNDAEL** y conocido como **AES** , reemplazará al DES.
- **AES** (Advanced Encription Standard) (2001). Adoptado como estándar efectivo de cifrado por el gobierno de los Estados Unidos en 2006. Divide los datos en bloques de 128 bits. Utiliza claves de 128, 192 ó 256 bits.
- **International Data Encryption Algorithm** o **IDEA** es un cifrador por bloques diseñado por Xuejia Lai y James L. Massey, descrito por primera vez en 1991. Fue también un algoritmo propuesto como reemplazo del DES (Data Encryption Standard)
- Otros: BlowFish, Serpent, TwoFish, RC6…

Algoritmo de establecimiento de claves Diffie-Hellman

- El algoritmo de Diffie-Hellman (1976) permite acordar una clave simétrica entre dos máquinas, a través de un canal inseguro y enviando únicamente dos mensajes. La clave resultante no puede ser descubierta por un atacante, aunque éste obtenga los dos mensajes enviados por el protocolo. Premio A.M Turing de la ACM de 2015 por este trabajo.
- Un interlocutor elige dos números (p,g) y se los envía al otro. Cada uno de ellos, además, elige otro número <p que guarda en secreto. Usando una fórmula matemática, que incluye la exponenciación, cada interlocutor hace una serie de operaciones con los dos números públicos y el secreto. A continuación los interlocutores se intercambian los resultados de forma pública.
- Su seguridad radica en que, en teoría (no demostrado), revertir esta función es tan difícil como calcular un logaritmo discreto (Un millón de millones de cuatrillones de veces más costosa que la exponenciación usada para transformar los números)
- La principal aplicación de este protocolo es acordar una clave simétrica con la que posteriormente cifrar las comunicaciones entre dos máquinas.
- Es atacable mediante Man-In-The-Middle.



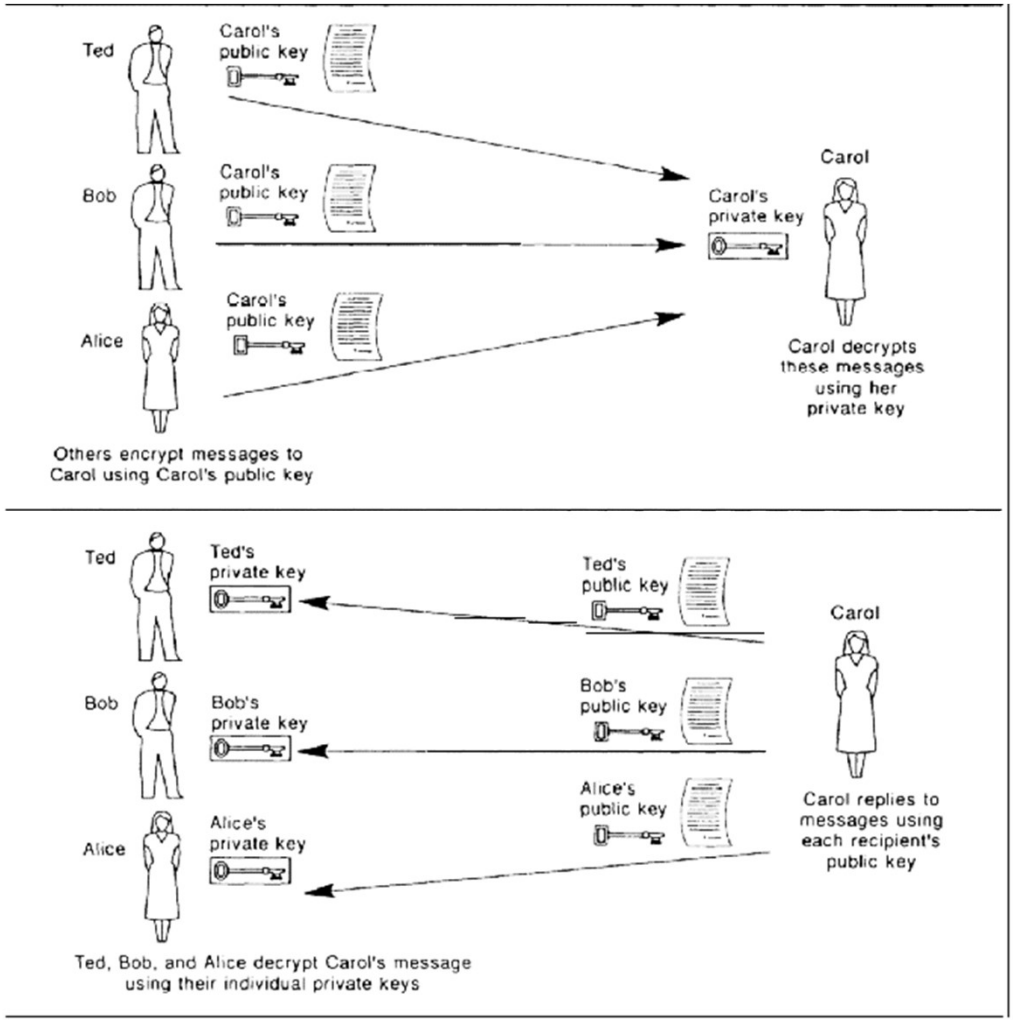
http://www.slideshare.net/samehelhakim/introduction-to-vpn-47256821

Algoritmos asimétricos (I)

Asimétricos (o de clave pública)

- Los sistemas de cifrado de clave pública o asimétricos surgen para evitar el problema del intercambio de claves de los sistemas de cifrado simétricos: no es necesario que el remitente y el destinatario acuerden la clave.
- Emisor y receptor tienen cada uno una pareja de claves. Una es privada (clave de descifrado KD) y debe ser mantenida en secreto. La otra es pública (clave de cifrado KC) y se distribuye a todos los posibles destinatarios
- Si un usuario quiere recibir mensajes cifrados, envía la clave pública KC al remitente y se queda con la de descifrado KD.
- Lo que cifra una clave privada sólo puede ser descifrado con la clave pública correspondiente, y viceversa.
- Los métodos criptográficos garantizan que esa pareja de claves sólo se puede generar una vez.
- La clave privada no puede deducirse de la pública, por lo que no hay peligro en transmitir las claves públicas por la red.

Algoritmos asimétricos (II)



https://www.w3.org/Talks/971008-tpp71j/slide5.htm

Ejemplos de algoritmos asimétricos. RSA (I)

Sistema RSA (Rivest, Shamir y Adleman) (1977)

- Origen de los sistemas de clave pública.
- Las claves están basadas en dos números primos (*p* y *q*).
- La clave pública es un par (n,f(p,q)), siendo n=p*q.
- La clave privada es un par (n, g(p,q))
- El cifrado y descifrado de bloques requiere muchas operaciones con números muy grandes, lo que hace que sean operaciones muy lentas.
- A partir de la clave pública es prácticamente imposible calcular la clave privada. La única forma de atacar el sistema es factorizando n: la factorización necesita mucho cómputo para su cálculo (multiplicar dos números y obtener un resultado es muy fácil y se tarda muy poco. Pero en el otro sentido, obtener qué dos números se han multiplicado para obtener el resultado (factorización) es muy complejo)
  - Actualmente una clave de 256 bytes puede factorizarse en unas cuantas horas.
  - En 2009 una clave de 768 bytes se factorizó usando 80 ordenadores durante 6 meses.
  - Hay diseños hardware para factorizar claves de 1024 bits.
  - Se considera que claves de 2048 bits son seguras hoy en día.

Ejemplos de algoritmos asimétricos. RSA (II)

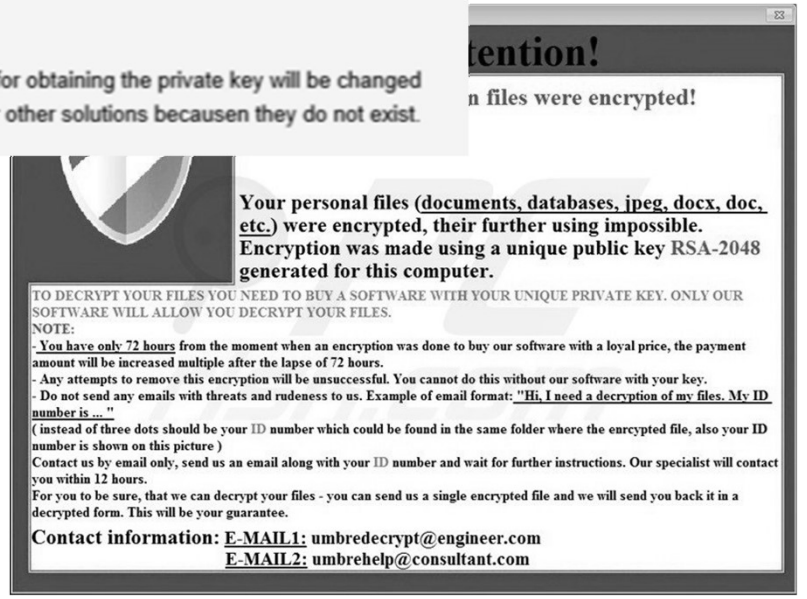
NOT YOUR LANGUAGE? USE Google Translate

What happened to your files?
All of your files were protected by a strong encryption with RSA4096
More information about the encryption RSA4096 can be found http://en.wikipedia.org/wiki/RSA_(cryptosystem)

What does this mean?
This means that the structure and data within your files have been irrevocably changed, you will not be able work with them, read them or see them, it is the same thing as losing them forever, but with our help, you can restore them

How did this happen?
Especially for you, on our SERVER was generated the secret key
All your files were encrypted with the public key, which has been transferred to your computer via the Internet.
Decrypting of YOUR FILES is only possible with the help of the private key and decrypt program which is on our Secret Server!!!

What do I do?
Alas, if you do not take the necessary measures for the specified time then the conditions for obtaining the private key will be changed
If you really need your data, then we suggest you do not waste valuable time searching for other solutions becausen they do not exist.



Ejemplos de algoritmos asimétricos. PGP

PGP (Pretty Good Privacy) (1991)

- Desarrollado por Phil Zimmerman
- Sistema ***híbrido*** de clave pública
- Operaciones de cifrado
  - Cifra mensaje con sistema simétrico (IDEA) con clave aleatoria
  - Codifica la clave con RSA
  - Envía el mensaje y la clave codificados
- Menos sobrecarga de encriptación/desencriptación
- Gratuito www.pgpi.org. No hay versiones nuevas desde 2002.
- Un “sucesor” suyo es GnuPG (GNU Privacy Guard): www.gnupg.org

Criptografía cuántica (I)

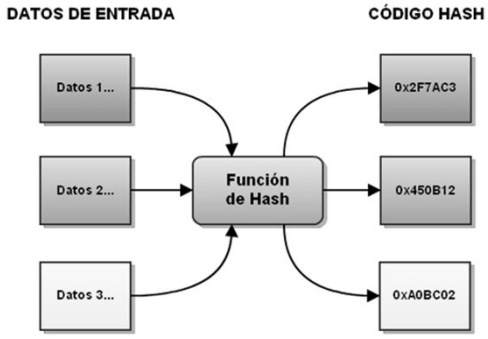
- La **criptografía cuántica** utiliza principios de la mecánica cuántica para cifrar y descifrar la información, garantizando la absoluta confidencialidad de la misma.
- La seguridad de la criptografía cuántica descansa en las bases de la mecánica cuántica, a diferencia de la criptografía de clave pública tradicional que se basa en la complejidad computacional de ciertas funciones matemáticas.
- Una de las propiedades más importantes de la criptografía cuántica es que si un tercero escucha durante la creación de la clave secreta, el proceso se altera advirtiéndose a las partes antes de que se transmita información privada. Esto es una consecuencia del principio de incertidumbre de Heisenberg, que dice que el proceso de medir en un sistema cuántico perturba dicho sistema.

Criptografía cuántica (II)

- Ya se han desarrollado protocolos cuánticos de distribución de claves, denominados QKD ('Quantum Key Distribution')
- Desde 2007, la transmisión del recuento de votos en las elecciones federales de Ginebra ya se ha protegido con criptografía cuántica. Algunos prestigiosos bancos suizos, por ejemplo, también están haciendo uso de esta técnica.
- La criptografía cuántica está cercana a una fase de producción, aunque actualmente resulta muy cara. Se utilizan láseres para emitir información en el elemento constituyente de la luz, el fotón, y conduciendo esta información a través de fibras ópticas.

Función Hash o de resumen

- Función matemática que crea a partir de una entrada (texto, contraseña o archivo) una salida alfanumérica de longitud fija. Esta salida representa la entrada de forma compacta.
- Lo que se obtiene al aplicar una función hash criptográfica a un mensaje (flujo de datos, o más usualmente, un archivo) se llama resumen criptográfico, huella digital o message digest
- Características:
  - Es poco costoso (computacionalmente) de obtener.
  - Es imposible en la práctica obtener el texto original a partir de su hash.
  - Es prácticamente imposible encontrar dos textos cuyos hash sean iguales.
- Hay muchos algoritmos de este tipo:
  - MD5 (Message Digest Algorithm, v5). Resumen de 128 bits
  - SHA-1 (Secure Hash Algorithm). Es parte de DSA (Digital Signature Algorithm). Resumen de 160 bits.
  - SHA-3. Estandar actual del NIST. Resúmenes de 224, 256, .., 512 bits. (http://csrc.nist.gov/publications/fips/fips180-4/fips-180-4.pdf)



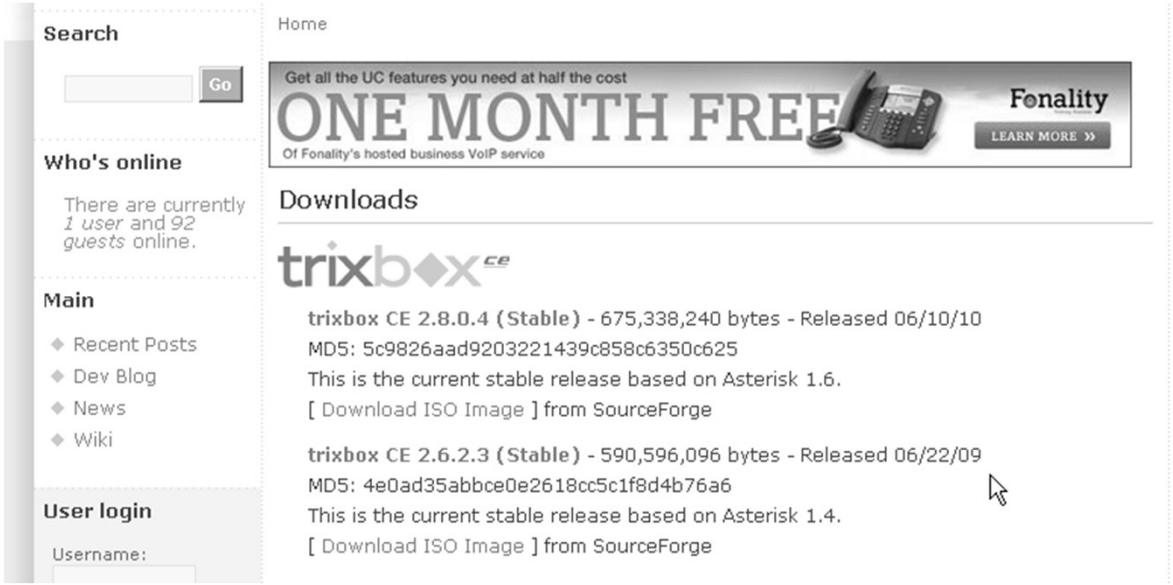
Funciones Hash: Aplicaciones

- **Almacenamiento de passwords**. Es imposible desencriptarlas (salvo utilizando *fuerza bruta*). Si se utiliza *sal* es mucho más seguro.

```
pepe:$1$vjFpMmig$2yJJhqiYADW0w03tPQZB9.:50:0:99999:7:
::
manolo:$1$0QT3fPmk$UHSi7AcfaLvXP49P3gbvb/:100:0:99999
:7:::
jose:$1$Je3O8E5l$KtQdySeElHaWqCmvm4HPy/:121:0:99999:7
:::
```

- **Huellas digitales (*fingerprint*)** de ficheros. Aseguran que el fichero es el original, es decir, que no ha sufrido modificaciones durante la transmisión. Así, por ejemplo, un programador que hace público un archivo ejecutable de un programa, puede calcular su hash (bien MD5, bien SHA) y publicarlo también. De esta forma, se sabe que cualquier otro archivo que no sea exactamente ese que ha publicado el autor, tendrá un resultado hash diferente.
- **Firmas digitales**. Aseguran que el fichero ha sido realizado por la persona u organización que afirma haberlo creado.

FirgerPrint: Ejemplo



Fuente: Inteco

- Una vez descargado ejecutaríamos (una vez instalado md5sum.exe, disponible en http://www.etree.org/cgi-bin/counter.cgi/software/md5sum.exe)

  c:\>md5sum trixbox-2.8.0.4.iso

  y devolvería el siguiente resultado:

  5c9826aad9203221439c858c6350c625 * trixbox-2.8.0.4.iso

- Como los dos hashes coinciden (el publicado en la página y el resultado de la operación de cálculo que se ha realizado) podemos decir que el archivo mantiene su integridad.

Firma digital

- La firma digital es el resultado de una operación criptográfica que relaciona el documento firmado con un componente personal. Este componente sería el Certificado digital.
- Una firma digital certifica un documento y le añade una marca de tiempo. Si posteriormente el documento fuese modificado de cualquier modo, el intento de verificar la firma fallaría.
- Las firmas suelen ser archivos con extensión SIG o ASC que resultan de firmar criptográficamente con la clave privada del autor el hash de un fichero. Si posteriormente se comprueba, a través de la clave pública, que el fichero firmado concuerda con la firma, es que se está ante un fichero realmente creado por quien dice haberlo hecho, y no modificado desde que se firmó.

Firma digital. Objetivos

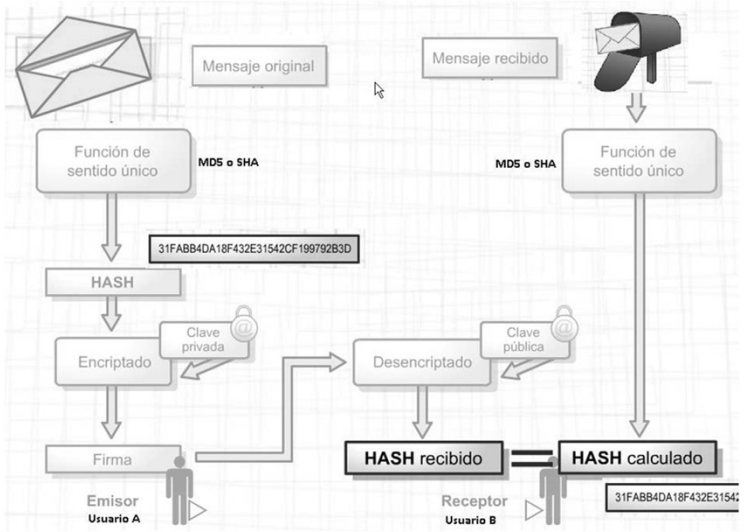
- La firma de un documento o mensaje tiene, por tanto, los siguientes objetivos:
  - **Autentificación del origen**: la firma ha de convencer al receptor que el emisor ha firmado el documento deliberadamente.
  - **Integridad**: debe asegurar que el documento/mensaje no ha sido modificado.
  - **No repudio**. Al garantizarse las dos anteriores, el autor no puede negar el contenido del mensaje.

Firma digital. Propiedades

- Para garantizar los objetivos anteriores, la firma debe tener una serie de propiedades:
  - Debe ser infalsificable.
  - No debe ser reutilizable: debe formar parte del documento y no poderse trasladar a ningún otro.
  - El documento no debe poder alterarse una vez firmado.
- Hoy en día, esto se implementa gracias a los ***certificados digitales***.

Firma digital y Funciones hash

- En criptografía asimétrica se cifra con la clave pública y se descifra con la privada.
- Análogamente, un texto se puede cifrar mediante la clave privada y descifrar mediante la pública.
- Este es el fundamento de la firma digital, que se basa en la firma del hash de un mensaje (puesto que firmar todo un mensaje es computacionalmente más costoso).
- Dado que el *hash* es siempre de un tamaño fijo y menor que el mensaje original, pero lo representa unívocamente, firmar el *hash* es equivalente a firmar un mensaje o archivo cualquiera



Hispasec

Firmas digitales. Sistemas existentes

Existen distintas formas de lograr las firmas digitales:

- ***DSS*** *(Digital Signature Standard)*
- ***RSA-PSS,*** **PKCS#1, PGP, GnuPG, …**

Firmas digitales: DSS

- ***DSS*** *(Digital Signature Standard)*.
  - Provee la firma de documentos, pero éstos no se encriptan.
  - Utiliza un algoritmo de clave pública (DSA, RSA, ECDSA) para firmar los datos.
  - Utiliza un función hash (SHA-1, SHA-2) sobre los datos del mensaje. Este *hash* del mensaje se encripta con la clave privada del emisor del mensaje.
  - Se transmiten conjuntamente el mensaje y su hash encriptada.
  - Cuando el mensaje llega a su destino, el receptor utiliza la clave pública del emisor para descifrar la *hash*. Aplica la misma función hash a los datos y si la hash así calculada es igual a la que traía el mensaje podemos estar seguros del origen y de la integridad de los datos.

Firmas digitales: RSA-PSS

- ***RSA-PSS,* PKCS#1, PGP, GnuPG, …**
  - Provee la firma de documentos, y además se encriptan.
  - Es similar al anterior, pero antes de transmitir el conjunto (mensaje, hash-encriptado) se encripta con la clave pública de encriptación del receptor del mensaje.
  - Podría simplificarse el proceso encriptando con la clave privada del emisor el mensaje, pero es computacionalmente mejor obtener la hash y cifrar ésta que cifrar directamente el mensaje.

Firmas digitales: Localizar claves públicas de usuarios

- A la hora de recibir un mensaje o un documento encriptado y/o firmado, necesitamos conocer la clave pública del remitente.
- Si bien puede habérnoslo hecho llegar por otros medios, siempre podemos intentar buscarlo en algún repositorio de claves públicas PGP.
- Existen muchos, donde podemos buscar según distintos criterios:
  - https://**keyserver**.**pgp**.com/
  - www.rediris.es/**keyserver**/
  - …
- En cualquier caso, cualquiera puede crear una clave con cualquier identidad y remitirla a esos servidores.

Autenticación de claves de usuario

- Como hemos visto, cualquier usuario puede crear claves con la identidad que quiera.
- Por lo tanto, necesitamos algún mecanismo adicional para asegurarnos que el poseedor de una clave es quien dice ser.
- Hoy en día pueden utilizarse dos mecanismos:
  - Certificados.
  - Anillos de confianza (en GnuPG y similares).
- En los dos casos la mecánica es la misma:
  - Un usuario U1 firma los certificados (claves) de otro usuario U2.
  - Nosotros confiamos en U1, por lo que aceptamos como válido el certificado de U2

Certificados y anillos de confianza: Diferencias

- Cuando se trabaja con certificados, cada certificado está firmado por únicamente una autoridad certificadora.
  - En función del nivel de confianza que tengamos en la autoridad certificadora podremos aceptar o no como válido ese certificado.
- Cuando se trabaja con anillos de confianza, cada *certificado* puede estar firmado por múltiples usuarios.
  - En función del nivel de confianza que tengamos en esos usuarios y en el número de firmas que haya, podremos aceptar o no como válido ese certificado.

Certificados

- Cuando se distribuyen claves públicas ¿cómo podemos estar seguros que una clave es de quien dice ser?
- Para autentificar las claves públicas se utilizan los ***certificados digitales***.
  - Un *certificado digital* es un documento que contiene información acerca de su dueño y la clave pública del mismo.
  - Esta clave es parte del par de claves publica-privada del usuario dentro de un esquema de criptografía de clave pública.
- Un certificado consiste, por tanto, en la asociación entre una entidad física y una firma, realizado por una entidad confiable. Dicho de otro modo, un certificado es similar a un carnet de identidad: sirve para identificar al usuario.
- El certificado será tan confiable como lo sea la autoridad que lo emite.
- DNI en España: http://firmaelectronica.gob.es/Home/Ciudadanos/DNI-Electronico.html

Autoridad Certificadora (I)

- Los certificados debe emitirlos una autoridad certificadora (*CA*). Esta "firma" ese certificado con su propia clave privada para garantizar su autenticidad. Actúa en cierta manera como un notario, certificando la validez del certificado.
- Si el receptor confía en la autoridad certificadora, entonces la información contenida en dicho certificado puede ser de confianza y el emisor del mensaje es quien dice ser.
- Además de garantizar el autor del mensaje/documento, por medio de una función hash (dependiente del propio certificado y del contenido del mensaje) se puede garantizar su integridad (evita *falsificaciones*).

Autoridad Certificadora (II)

- Hoy en día existen *CAs* públicas conocidas (*Ceres*, *Verisign*, *Thawte*, …).
  - Estas emiten certificados a petición de sus clientes, pero a un precio establecido.
- Es posible crear autoridades certificadoras privadas de una organización.
  - El coste de este tipo de *CAs* es mantenerlas.
  - Además, las personas ajenas a la empresa pueden no confiar en la misma.

Certificados digitales: Tipos

Hay 4 tipos de certificados:

- **Personales**: Emitidos a usuarios para tareas típicas como autentificación o emitir firmas digitales. Un usuario puede tener varios, cada uno firmado por la *CA* que lo emitió.
- **De servidores o máquinas**: También las máquinas pueden necesitar demostrar su identidad a otras entidades.
- **De software**: Estos certificados se envían a desarrolladores de *software* para que firmen sus programas antes de su producción o distribución.
- **De autoridad certificadora**: Los que una *CA* usa para firmar aquellos certificados que envía a otras entidades.

Certificados digitales personales: Tipos

A su vez, a nivel de certificados personales se suelen distinguir cuatro tipos de certificados:

- **Personal:** Acredita la identidad de una persona.
- **De pertenencia a empresa:** Además de la identidad de la persona, certifica que pertenece a una determinada empresa.
- **De representante de empresa:** además de lo anterior, certifica que el propietario del certificado tiene poderes de representación de la empresa.
- **De persona jurídica:** identifica a una empresa como tal a la hora de realizar trámites administrativos.

Certificados digitales: Estructura

- Todos los certificados, independientemente del tipo que sean, tienen la misma estructura.
- Entre otras cosas, incluyen:
  - Identidad del propietario del certificado.
  - Clave pública del mismo.
  - Objeto del certificado.
  - Identidad del emisor del certificado (*CA*)
  - Periodo de validez.
  - Algoritmos utilizados.
  - Huella digital (para probar su integridad).

Certificados digitales: Usos más frecuentes

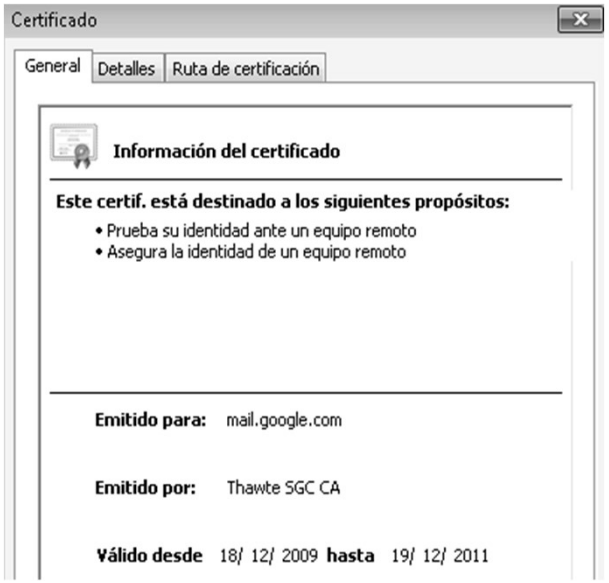
- Identificación de servidores
- Autentificación de clientes en transacciones web
- Autentificación de mensajes de correo electrónico
- Autentificación de ficheros
- Autentificación de código

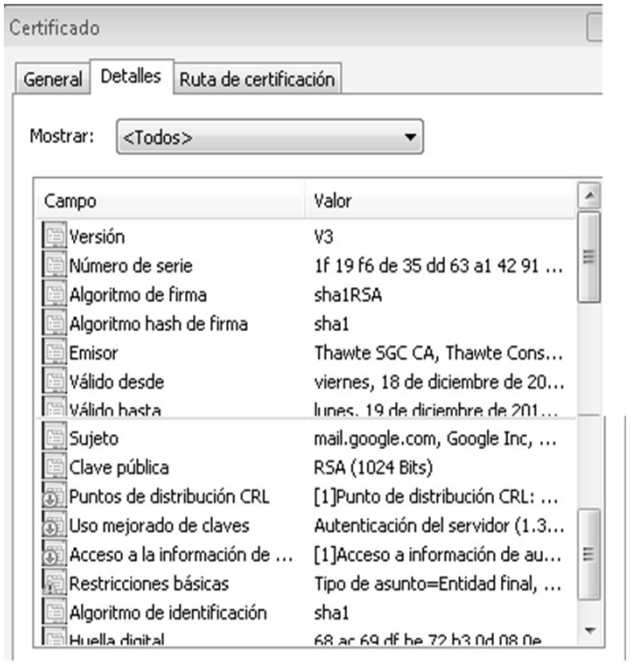
Certificados digitales: Usos más frecuentes

Identificación de servidores

- Hoy en día es la aplicación más frecuente.
- Se usa para asegurarse que la máquina a la que nos conectamos es realmente quien dice ser.
- Su mayor utilidad es para conectarnos a servidores web a través del protocolo https:
  - Nos garantiza que el servidor es quien dice ser.
  - La transmisión de datos se hace de manera cifrada, con lo que cualquier dato confidencial no podrá ser obtenido *esnifando* la red.
- Fundamental para garantizar la privacidad de contraseñas, correo, transacciones electrónicas, …

Identificación de servidores

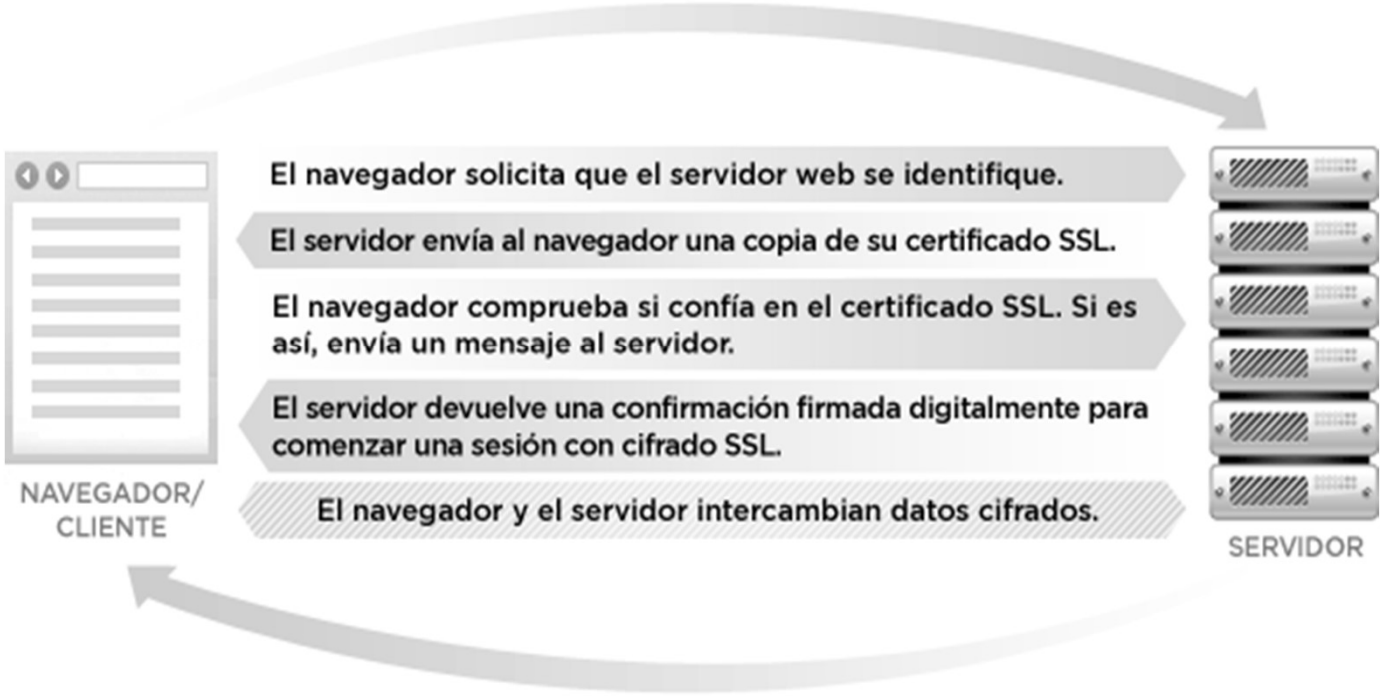




- Hoy en día Google ha cambiado su certificado.
- Comprobad los algoritmos empleados en la actualidad

Certificados digitales. Funcionamiento de https

.



Certificados digitales. Conexión por ssh

- De manera similar a https, se puede usar ssh para encriptar las conexiones a una máquina remota.
- En el principio de la conexión el servidor envía su clave pública al cliente.
- Si el cliente acepta esta clave, genera una clave de sesión aleatoria, la encripta con la clave del servidor y se la envía.
- Esa clave se utilizará para la transmisión encriptada (con un algoritmo simétrico) de los datos.

Certificados digitales.

Autentificación de clientes en transacciones web

- A la hora de realizar trámites con empresas o instituciones, el cliente también debe identificarse, para que la empresa o institución pueda asegurarse que quien realiza la operación es realmente quien dice ser.
- Esto es especialmente importante en trámites oficiales (Agencia Tributaria, Ayuntamientos, …) y para transacciones comerciales.
- Tradicionalmente muchas empresas utilizaban contraseñas, tarjetas de coordenadas, etc. Hoy muchas soportan certificados personales (como el DNI electrónico).
- Las instituciones oficiales suelen reconocer certificados emitidos por el Ceres (de la Fábrica Nacional de Moneda y Timbre).

Certificados digitales.

Autentificación de mensajes de correo electrónico

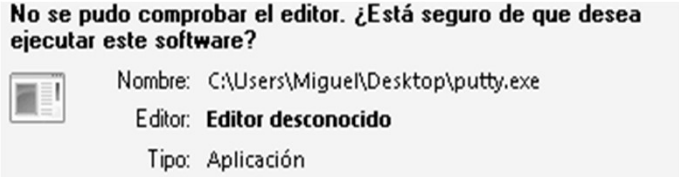
- Determinados clientes de correo permiten llevar a cabo la gestión (envío y recepción de correos) firmados y/o encriptados.
- Existen sistemas basados en certificados o en pares de claves públicas/privadas RSA o GnuPG, por ejemplo.
- También existen sistemas de webmail que hacen lo mismo (hushmail, por ejemplo).
- Los navegadores tienen soporte para gestionar certificados digitales, de tal manera que el servidor solicita al navegador la identidad del usuario.

Certificados digitales. Autentificación de ficheros

- Existen aplicaciones que permiten firmar y/o encriptar ficheros con claves RSA/GnuPG o similar o con certificados.
- Estos sistemas permiten la utilización de documentos electrónicos con la misma validez legal que documentos físicos firmados.
- Al existir muchos sistemas, en cada caso habrá que utilizar los que las instituciones/empresas a los que nos dirijamos requieran (eCofirma, Adobe, …)

Certificados digitales. Autentificación de código (I)

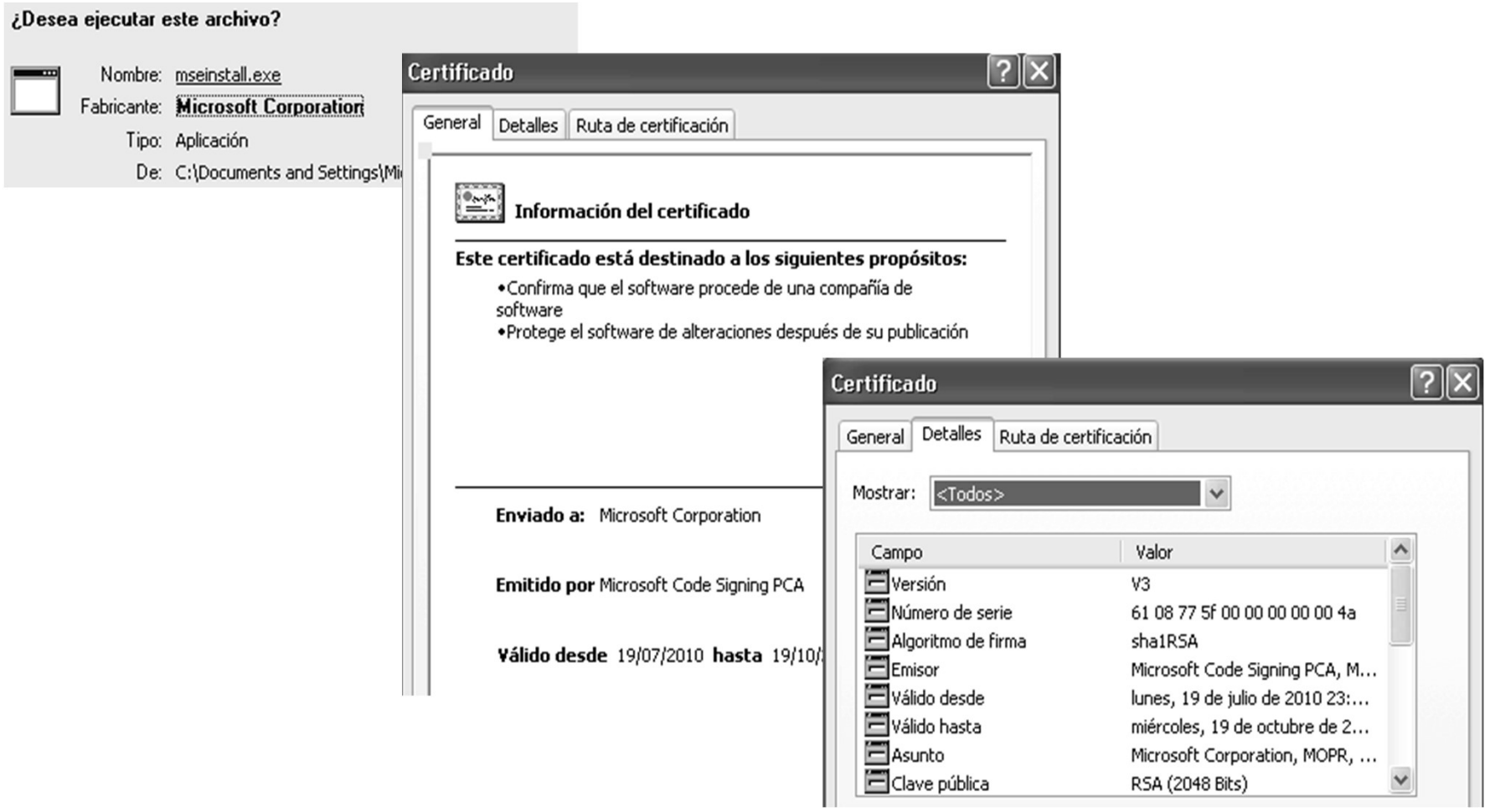
- Los programas también pueden firmarse para garantizar la autoría e integridad del mismo.
- Un programa, en este aspecto, puede aparecer en tres estados:
  - Sin firmar. No se sabe quién es su autor.



  - Firmado pero sin certificado de confianza.

Certificados digitales. Autentificación de código (II)

- Firmado con certificado de confianza.



Anillos de confianza (GnuPG)

- Cuando nos hacemos con una clave pública de un usuario, podemos asociar a esa clave un nivel de confianza:
  - Desconocido. No sabemos qué nivel de confianza nos ofrece esa clave.
  - Ninguno. Sabemos que no nos ofrece ningún nivel de confianza.
  - Marginal. No estamos muy seguros, conocemos los riesgos pero aceptamos la clave.
  - Absoluto. La firma con esa clave es tan buena como la nuestra.
- Cuando nos hagamos con la clave pública de otro usuario, puede venir firmada por otros usuarios. En función de la confianza que tengamos en esas firmas, podremos confiar más o menos en el nuevo usuario (incluso sin conocerlo).
- Con las relaciones de confianza entre un conjunto de usuarios se forma lo que se llama un *anillo de confianza*.

Cifrado Negable (I)

- Hemos visto que cuando queremos ocultar información, la esteganografía y la criptografía ofrecen niveles de seguridad muy altos.
- Pero a veces puede haber problemas simplemente por su uso:
  - En algunos países puede ser ilegal.
  - El simple hecho de usar criptografía puede mostrar el hecho de que tenemos algo que ocultar.
  - Puede también que, por ejemplo, un usuario se vea obligado a proporcionar una clave de desencriptado.
- Surge entonces (desarrollado, entre otros, por Julian Asange) el concepto de Cifrado Negable (Deniable Encryption).

Cifrado Negable (II)

- Se denomina cifrado negable el que permite al usuario negar, de manera convincente, que está ocultando un mensaje. Esto puede hacerlo por dos vías:
  - "Demostrando" que no existe ese mensaje oculto.
  - "Desencriptando" el mensaje oculto, pero dando como resultado un texto diferente al que realmente quiere ocultar.
- Existen diversas herramientas con este propósito:
  - OpenPuff Steganography and Watermarking
  - FreeOTFE (On The Fly Encryption).
  - TrueCrypt

Cifrado Negable (III)

- Se parte del texto a proteger y de un texto alternativo que se *sacrificará* en caso de necesidad. Se encriptan ambos textos en un único documento y se obtienen dos claves.
- Si es necesario desencriptar se puede utilizar una u otra clave obteniendo el texto a *proteger* o el que se puede *sacrificar.*
- Tiene detractores ya que si saben que están ante un cifrado de este tipo aunque se les dé el texto a proteger pueden pensar que se les sigue engañando y sigan *presionando* para obtenerlo.