

# Seguridad, privacidad y aspectos legales

---

Álvaro López García

Grupo de Computación Avanzada y e-Ciencia  
Instituto de Física de Cantabria (IFCA) - CSIC-UC

Máster universitario en ciencia de datos / Master in Data Science



CONSEJO SUPERIOR DE INVESTIGACIONES CIENTÍFICAS

**CSIC**



**UIMP**  
Universidad Internacional  
Menéndez Pelayo

## Parte I

# Tecnologías para la protección de la información y privacidad

## 1. Conceptos Generales

# Fundamentos criptográficos

---

## Definición

«Campo que se encarga del estudio de los algoritmos, protocolos y sistemas que se utilizan para dotar de seguridad a las comunicaciones, a la información y a las entidades que se comunican.» J. Pastor Franco, M. A. Sarasa López, J. L. Salazar Riaño. Criptografía digital: fundamentos y aplicaciones.

**Autenticación** asegurar que alguien es quien dice ser.

**Confidencialidad** asegurar que nadie no autorizado puede acceder a la información.

**Integridad** asegurar que los datos no pueden ser manipulados.

**No-repudia** asegurar que nadie puede denegar que haya generado una información.

- Presente en nuestro día a día, aunque de forma imperceptible a veces.
- Transacciones financieras, tarjetas de crédito, etc.
- Certificados digitales y DNI digital.
- Conexiones seguras (https).
- Firma digital de documentos oficiales.
- Transferencias de ficheros.
- Cifrado de discos



## Your connection is not secure

The owner of **bad.example.com** has configured their website improperly. To protect your information from being stolen, Firefox has not connected to this website.

[Learn more...](#)

Go Back

Advanced

Hoy

🔒 Las llamadas y mensajes enviados a este chat ahora están seguros con cifrado de extremo a extremo. Pulsa para más información.

A grandes rasgos, los sistemas criptográficos se basan en 3 tipos de algoritmos.

**Clave privada** o simétrica, donde se utiliza una sola clave.

**Clave pública** o asimétrica, donde se utilizan un par de claves, una pública y otra privada

**Función hash** o message-digest.

Los sistemas entran dentro de dos categorías.

- Computacionalmente seguros.
  - No hay recursos y/o tiempo suficientes para romperlos (hoy).
  - Ejemplo: tiempo estimado para romper el cifrado de un certificado SSL de 2048-bit: 6.440 trillones de años (<https://www.digicert.com/TimeTravel/>).
  - Son la mayoría de sistemas de cifrado.
- Seguros de manera incondicional.
  - Nunca se pueden romper.
  - Libretas de un solo uso.



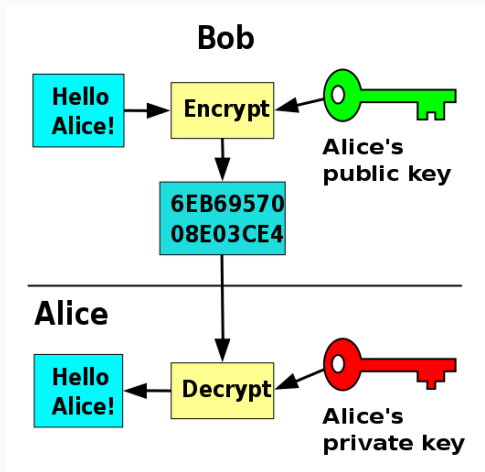
- Clave secreta para cifrar la información.
- Misma clave para descifrar la información.
- La clave tiene que ser conocida por ambos extremos de la comunicación.
- Hay que compartir la clave, debe existir un contacto previo.
- Ventajas:
  - Eficiente.
  - Sencillo de implementar.
- Desventajas:
  - Dificultad para compartir claves.
  - Número de claves elevado.
  - No hay posibilidad de no-repudio.
- DES, Blowfish, AES, etc.

- Se usa un par de claves: pública y privada.
- La clave privada es secreta, la clave pública es pública.
- Ambas claves están relacionadas matemáticamente.
- Una información cifrada con una clave pública solo puede ser descifrada con su clave privada.
- Una información cifrada con una clave privada solo puede ser descifrada con su clave pública (firma).

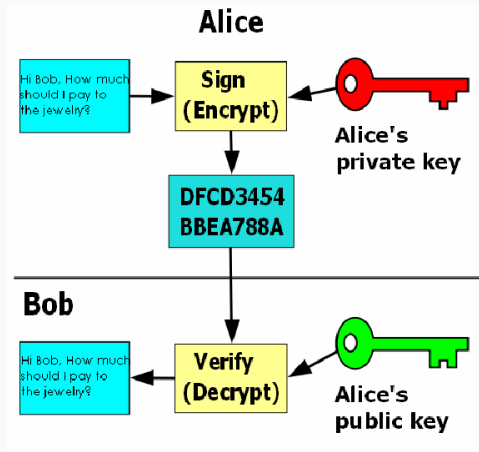
# Clave asimétrica

## Cifrado y firmado

### Cifrado



### Firmado



- Ventajas:
  - No hay que compartir ningún secreto.
  - Las claves públicas se pueden distribuir libremente.
  - No hay por qué establecer un contacto previo.
  - Firma digital.
- Desventajas:
  - No es eficiente, cifrado y descifrado mucho más lento.
  - El tamaño de las llaves tiene que ser más grande.
  - ¿Como asegurar que la clave pública de alguien es realmente suya?
- Sistemas: RSA, Diffie-Hellman, DSA, El-Gamal.

# Combinaciones de sistemas

## Clave simétrica y asimétrica

- Problema: el cifrado asimétrico es mucho más lento.
- ¿Qué sucede con las comunicaciones con un sistema? (p.e. https)
- Solución: usar una combinación de las dos.
  - Se usa la clave asimétrica para comunicar de forma segura una clave simétrica (generada automáticamente).
  - Se usa la clave simétrica para el resto de la comunicación.

- Es necesario establecer la confianza en una clave pública.
- Idealmente, podríamos comunicarla uno a uno (imposible en la práctica).
- Diferentes esquemas:
  - Autoridades de Certificación (CA).
  - Red de confianza.
  - ...

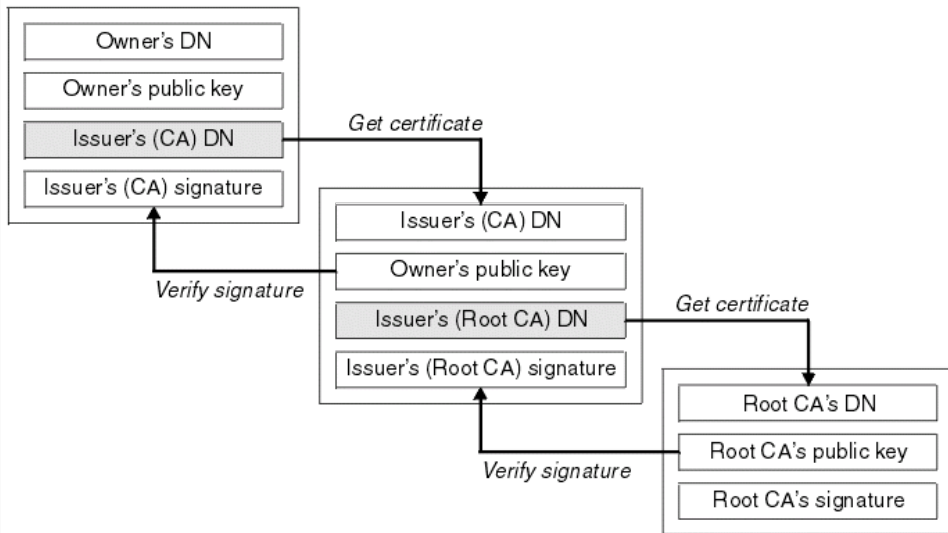
# Métodos propagación de la confianza I

## Autoridades de Certificación (CA)

- Entidades encargadas de certificar (firmar) las claves públicas.
- Después de generar un par de claves, el usuario demuestra su identidad frente a una CA.
- El firmado se hace offline.
- La confianza está basada en la autoridad.
- Hay CAs de confianza, instaladas por defecto (navegador, sistema operativo).
  - Ejemplo: FNMT en Mozilla (Firefox):  
[https://bugzilla.mozilla.org/show\\_bug.cgi?id=435736](https://bugzilla.mozilla.org/show_bug.cgi?id=435736)
- El firmado puede ser jerárquico (root authority).
- Estándar: X.509.

# Métodos propagación de la confianza II

## Autoridades de Certificación (CA)





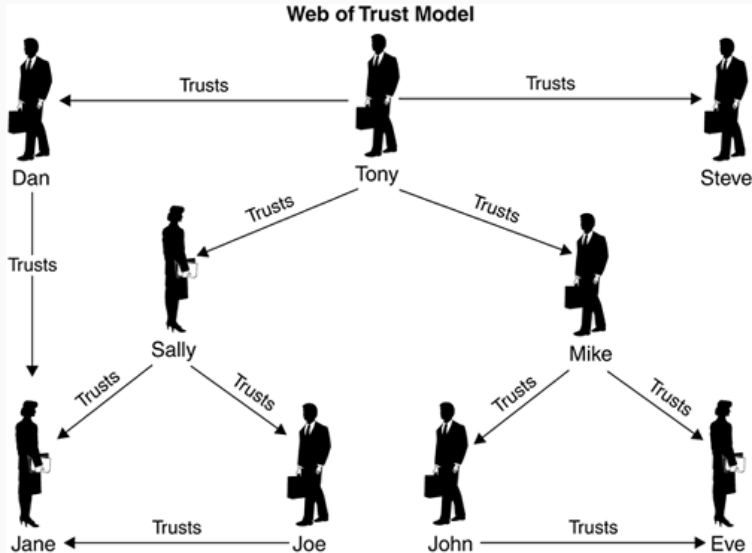
# Métodos propagación de la confianza I

## Red de confianza

- Utilizado en PGP (Pretty Good Privacy).
  - Programa de cifrado que sigue el estándar OpenPGP (RFC 4880).
  - Desarrollado por Phil Zimmermann en 1991.
  - GnuPG: implementación libre de OpenPGP.
- No existen autoridades de certificación, sino que son los usuarios los que firman.
- La confianza es transitiva.

# Métodos propagación de la confianza II

Red de confianza



# Hash criptográficos

- Message-digest.
- A partir de una información de tamaño arbitrario, se produce una salida de un tamaño fijo (digest o hash).
- Cifrado de un solo sentido.
- Propiedades de los hash criptográficos
  - No se puede conocer la información a partir de hash.
  - No se puede encontrar una información arbitraria que genere un hash determinado.
  - No debe haber colisiones.
  - Sensible a los cambios en la entrada.
- Usos: Firmas digitales, control errores, checksum ficheros, cifrado de contraseñas, etc.
- MD1-5, SHA1, CRC, etc.

```
alvaro@torio:~/w/talks/master/seguridad $ sha1sum talk.pdf
55aade398b78da7091b2d8737b1e199afda5ee4e  talk.pdf
alvaro@torio:~/w/talks/master/seguridad $ md5sum talk.pdf
be3a89d83682f06e31d342caa967e9c7  talk.pdf
```

# Fundamentos criptográficos

---

Ejercicios prácticos

```
ubuntu@gpgnode:~$ gpg --list-keys
gpg: directory '/home/ubuntu/.gnupg' created
gpg: keybox '/home/ubuntu/.gnupg/pubring.kbx' created
gpg: /home/ubuntu/.gnupg/trustdb.gpg: trustdb created

ubuntu@gpgtest:/practicass$ gpg --list-keys
```

```
ubuntu@gpgnode:~$ gpg --search-keys aloga@ifca.unican.es
gpg: data source: https://51.38.91.189:443
(1)      Alvaro Lopez Garcia <aloga@ifca.unican.es>
        4096 bit RSA key 3D9857115570F717, created: 2018-05-08
(2)      Alvaro Lopez Garcia <aloga@ifca.unican.es>
        Alvaro Lopez Garcia <alvaro.lopez@csic.es>
        Alvaro Lopez Garcia <alvaro@perseverantia.com>
        Alvaro Lopez Garcia <alvaro.lopezgarcia@unican.es>
        1024 bit DSA key C663E40450738AE3, created: 2009-08-05
Keys 1-2 of 2 for "aloga@ifca.unican.es".  Enter number(s), N)ext, or Q)uit > 1
gpg: key 3D9857115570F717: 1 signature not checked due to a missing key
gpg: key 3D9857115570F717: public key "Alvaro Lopez Garcia <aloga@ifca.unican.es>" imported
gpg: no ultimately trusted keys found
gpg: Total number processed: 1
gpg:          imported: 1

ubuntu@gpgnode:~$ gpg --list-keys
/home/ubuntu/.gnupg/pubring.kbx
-----
pub      rsa4096 2018-05-08 [SC]
         DD1E52B5DACBF2B6FFFC98D23D9857115570F717
uid          [ unknown] Alvaro Lopez Garcia <aloga@ifca.unican.es>
sub      rsa4096 2018-05-08 [E]
sub      rsa4096 2018-05-08 [S]
```

```
ubuntu@gpgnode:/practicass$ gpg --import aloga.pub
gpg: key 3D9857115570F717: 1 signature not checked due to a missing key
gpg: key 3D9857115570F717: "Alvaro Lopez Garcia <aloga@ifca.unican.es>" not changed
gpg: Total number processed: 1
gpg:             unchanged: 1

ubuntu@gpgnode:~$ gpg --list-keys
/home/ubuntu/.gnupg/pubring.kbx
-----
pub      rsa4096 2018-05-08 [SC]
          DD1E52B5DACBF2B6FFFC98D23D9857115570F717
uid          [ unknown] Alvaro Lopez Garcia <aloga@ifca.unican.es>
sub      rsa4096 2018-05-08 [E]
sub      rsa4096 2018-05-08 [S]
```

```
ubuntu@gpgnode:/practicass$ gpg --verify mensaje1.txt.gpg
gpg: Signature made Fri May 3 11:40:07 2019 UTC
gpg:                using RSA key ECD3FC684570B68020C69E04030935F8354CFC5A
gpg: Good signature from "Alvaro Lopez Garcia <aloga@ifca.unican.es>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg:                There is no indication that the signature belongs to the owner.
Primary key fingerprint: DD1E 52B5 DACB F2B6 FFFC 98D2 3D98 5711 5570 F717
Subkey fingerprint: ECD3 FC68 4570 B680 20C6 9E04 0309 35F8 354C FC5A
```



```
ubuntu@gpgtest:/practicass$ gpg --edit-key aloga@ifca.unican.es
gpg (GnuPG) 2.2.4; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

(...)

gpg> trust

(...)

Please decide how far you trust this user to correctly verify other users' keys
(by looking at passports, checking fingerprints from different sources, etc.)

 1 = I don't know or won't say
 2 = I do NOT trust
 3 = I trust marginally
 4 = I trust fully
 5 = I trust ultimately
 m = back to the main menu

Your decision? 5
Do you really want to set this key to ultimate trust? (y/N) y

(...)

Please note that the shown key validity is not necessarily correct
unless you restart the program.
```

```
ubuntu@gpgnode:/practicass$ gpg --verify mensaje1.txt.asc
gpg: Signature made Fri May 3 11:40:07 2019 UTC
gpg:                using RSA key ECD3FC684570B68020C69E04030935F8354CFC5A
gpg: checking the trustdb
gpg: marginals needed: 3  completes needed: 1  trust model: pgp
gpg: depth: 0  valid: 1  signed: 0  trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: Good signature from "Alvaro Lopez Garcia <aloga@ifca.unican.es>" [ultimate]

ubuntu@gpgnode:/practicass$ gpg --verify mensaje2.txt.asc
gpg: Signature made Fri May 3 11:53:43 2019 UTC
gpg:                using RSA key ECD3FC684570B68020C69E04030935F8354CFC5A
gpg: Good signature from "Alvaro Lopez Garcia <aloga@ifca.unican.es>" [ultimate]
gpg: WARNING: not a detached signature; file 'mensaje2.txt' was NOT verified!
```

<https://www.kernel.org/signature.html>

```
ubuntu@gpgnode:/practicass$ gpg --verify kernel/linux-4.16.7.tar.sign kernel/linux-4.16.7.tar
gpg: Signature made Tue May 1 19:48:10 2018 UTC
gpg:
gpg: using RSA key 647F28654894E3BD457199BE38DBBDC86092693E
gpg: Can't check signature: No public key

ubuntu@gpgnode:/practicass$ gpg --search-keys gregkh@kernel.org
gpg: data source: https://51.38.91.189:443
(1)
  Greg Kroah-Hartman <gregkh@kernel.org>
  Greg Kroah-Hartman <gregkh@linuxfoundation.org>
  Greg Kroah-Hartman (Linux kernel stable release signing key) <greg@kro
    4096 bit RSA key 38DBBDC86092693E, created: 2011-09-23
Keys 1-1 of 1 for "gregkh@kernel.org". Enter number(s), N)ext, or Q)uit > 1
gpg: key 38DBBDC86092693E: 2 duplicate signatures removed
gpg: key 38DBBDC86092693E: 160 signatures not checked due to missing keys
gpg: key 38DBBDC86092693E: public key "Greg Kroah-Hartman <gregkh@linuxfoundation.org>" imported
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: Total number processed: 1
gpg:
gpg: imported: 1

ubuntu@gpgnode:/practicass$ gpg --verify kernel/linux-4.16.7.tar.sign kernel/linux-4.16.7.tar
gpg: Signature made Tue May 1 19:48:10 2018 UTC
gpg:
gpg: using RSA key 647F28654894E3BD457199BE38DBBDC86092693E
gpg: Good signature from "Greg Kroah-Hartman <gregkh@linuxfoundation.org>" [unknown]
gpg:
gpg: aka "Greg Kroah-Hartman <gregkh@kernel.org>" [unknown]
gpg:
gpg: aka "Greg Kroah-Hartman (Linux kernel stable release signing key) <greg@kroah.com>"
[unknown]
```

```
ubuntu@gpgtest:/practicass$ nano archivo.txt

ubuntu@gpgtest:/practicass$ gpg --encrypt --armor --recipient aloga@ifca.unican.es archivo.txt

ubuntu@gpgtest:/practicass$ cat archivo.txt.asc
-----BEGIN PGP MESSAGE-----
Version: GnuPG v1

hQQ0AzI6t9NYmoVnEA//Sejzdocsf9zYP+5T/PsRUps41lz97X8PXJBkmZHv1W0H5
sLsLQVRa0MFJ1alv0S1K/FDpdbBi+mUZThPBZW+Ko+cw9aaUj40et45Wp4wpXvZX
uYcuoBADov1Qzbj0/ybXR3n14xYuAjekrLeJaBvYkVaeC8wC/up7Jm5CPTsvSQIV
uMmaxruCNOW4S0aL3bgAuXLW70B0zrR2KlquxArVzdPQCum4NkPJakKHf4iaQfOT
XjzNL1fr6c4sQAom/3ZbEhftKly44nYJ6R/6fZCQz4xXbvWdSwUdWMNCurlh/pwz
EOHzqObIK9fr2T++JYOSxJ2nG/Vf5wMP2eV8I+Pij04hDJaR+Zg2oCiJ6kjj0dWT
WfaxlsygDl/9tt/rRPd0Y5K55rvlu1ljwfXqTXUCNU1kE6bHuMHgOXn7XGE01Cso
RPhD1A5+xYlxiZyGslRk+Yvwsypy3Bc+SUY0CCmEFi3S3S5+9x03r2Jy2z3xs8A
YuHcQf9x17eUCIAwvdQ3khcb+RIQSPfqEh9di5/2+6bAIEd11lB5bX52yHPGzg/L
o30s5LwyAS1TRE8QVKEam4tqvhtR5cQYeLMz6+ZC7Y9FOPxJZc4HjXN5/12MiadC
7DlgrAQFrLYuLqk/towimwW4XKqY3vSK09PPBmLX1bD0Y6Dkcho3AWV/ukIAA2QP
/36KoeopcSKmkAGLdn7qL+RUt05Ze+NMvdHYHQbzhs/rXVKNosmGkk6969yBHGQt
KExuvCYegz9Z01tXu30aC3aeb6QcipH86ywZXim6T+U3P51LW4ul4N80ZG5Kg104
pNp4CPcpt3isvapCkkdCidZNIH3NBEjp68AaWz1maH+XpZ8TQ7tbFr597xVAX449
ufBg9RWPcGhU38p/RiadBW1L0ocjKZCtRHMs8HoHv1HNT7eTqbnKGJOVrtawFD0u
2190daPR/8Uv0jGX0HH0Gef1dtVljOpYYYEu6FthevvNsE6qvbi19bbZWNUXZNVK
OF3+5i7NpvAgVdqrkkrAb7vaVitFXNRbX+Xn4xo0gb2536xM6AcRQushVapewMy
J2NBLRINf3P3Hvm03M+8xh4nk651T8pJzSSLZZp1idfZuHb/g/OWpbq42X4Q02TS
RaRUwZm0LATIv0ZaJdjtkXDWkxcvR0ZliJ8fS6fp0xqShXDAKDB2fMp4F+LenizH
UezPKdSfMqgRXi3UTX9VgfskUMaYFLjpfEqFHIQ0p8eSWbTbQXx2qTR0VRb5iujS
PkiJOZIIy/xxX62fi5resJNw3pMadGL57r/2UolNPbaF5vScAbCKWHBx9c6YTwAV
```

```
ubuntu@gpgnode:/practicass$ gpg --gen-key
gpg (GnuPG) 2.2.4; Copyright (C) 2017 Free Software Foundation, Inc.
This is free software: you are free to change and redistribute it.
There is NO WARRANTY, to the extent permitted by law.

Note: Use "gpg --full-generate-key" for a full featured key generation dialog.

GnuPG needs to construct a user ID to identify your key.

Real name:

(...)
```

```
ubuntu@gpgtest:/practicass/centos$ ls
CentOS-7-x86_64-GenericCloud-1802.qcow2  CentOS-7-x86_64-GenericCloud-1802.qcow2.xz  sha256sum.txt
sha256sum.txt.asc
ubuntu@gpgtest:/practicass/centos$ sha256sum -c sha256sum.txt --ignore-missing
CentOS-7-x86_64-GenericCloud-1802.qcow2: OK
CentOS-7-x86_64-GenericCloud-1802.qcow2.xz: FAILED
sha256sum: WARNING: 1 computed checksum did NOT match
```

```
ubuntu@gpgnode:/practicass/centos$ gpg --verify sha256sum.txt.asc
gpg: Signature made Thu Jan 31 15:28:30 2019 UTC
gpg: using RSA key 24C6A8A7F4A80EB5
gpg: Can't check signature: No public key

ubuntu@gpgnode:/practicass/centos$ gpg --search-keys 24C6A8A7F4A80EB5
gpg: data source: https://51.38.91.189:443
(1) CentOS-7 Key (CentOS 7 Official Signing Key) <security@centos.org>
    4096 bit RSA key 24C6A8A7F4A80EB5, created: 2014-06-23
Keys 1-1 of 1 for "24C6A8A7F4A80EB5". Enter number(s), N)ext, or Q)uit > 1
gpg: key 24C6A8A7F4A80EB5: 15 signatures not checked due to missing keys
gpg: key 24C6A8A7F4A80EB5: public key "CentOS-7 Key (CentOS 7 Official Signing Key) <security@centos.org>" imported
gpg: marginals needed: 3 completes needed: 1 trust model: pgp
gpg: depth: 0 valid: 1 signed: 0 trust: 0-, 0q, 0n, 0m, 0f, 1u
gpg: Total number processed: 1
gpg: imported: 1

ubuntu@gpgnode:/practicass/centos$ gpg --verify sha256sum.txt.asc
gpg: Signature made Thu Jan 31 15:28:30 2019 UTC
gpg: using RSA key 24C6A8A7F4A80EB5
gpg: Good signature from "CentOS-7 Key (CentOS 7 Official Signing Key) <security@centos.org>" [unknown]
gpg: WARNING: This key is not certified with a trusted signature!
gpg: There is no indication that the signature belongs to the owner.
Primary key fingerprint: 6341 AB27 53D7 8A78 A7C2 7BB1 24C6 A8A7 F4A8 0EB5
gpg: WARNING: not a detached signature; file 'sha256sum.txt' was NOT verified!
```

**Tarea** Escribir un mensaje para `aloga@ifca.unican.es`, **cifrarlo** y enviarlo por correo electrónico.

**Bonus** **Cifrar y firmar** el mensaje con el par de claves generadas en clase.



# Identidad digital: Autenticación y Autorización

---

- Conjunto de rasgos que hace ser a una persona quien es.
- Conjunto de rasgos que distinguen a un individuo del resto.
- Diferentes ámbitos, diferentes identidades, diferentes características.
  - Ámbito biológico, identidad física: rasgos y características físicas.
  - Ámbito legal, identidad oficial (legal): DNI.
  - Ámbito familiar.
  - Ámbito digital.
- Es un factor clave cuando se trata de establecer una relación entre dos partes.
- Hasta hace algunos años solo era necesario gestionar nuestra identidad con respecto a nuestro entorno físico más cercano.
- Hoy en día no es así: compras por Internet, acceso a recursos on-line, aplicaciones móviles, etc.

- Existen diferentes representaciones de un individuo en el mundo digital.
- Según el ámbito o dominio una identidad tendrá unas características u otras.
- No tiene por qué existir una correspondencia (ni validez) de identidades entre dominios.
- Tipos de identidad, según el dominio donde se va a utilizar.
  - Sector público.
    - Utilización de credenciales que representen a un individuo real unívocamente.
    - Validación de la identidad real, física, del individuo.
    - Objetivo: Validación identidad física e identidad digital. Correspondencia 1:1.
    - Credenciales generadas por autoridades oficiales, con validez legal.
    - Ejemplos: Fábrica Nacional de Moneda y Timbre, Ministerio del Interior.

- Identidad corporativa.
  - Credenciales generadas por la corporación (empresa, organismo) de forma interna, ligadas a una identidad física.
  - Pueden incorporar atributos de otras identidades (validados o no) e incluso añadidos por el usuario (sin validar).
  - Objetivo: Validar un usuario en el dominio de la aplicación.
  - No hay correspondencia 1:1. Depende de que se utilice para identificar al usuario (por ejemplo, cuenta bancaria).
  - Validez legal o no.
- Identidad social.
  - Identidad sin correspondencia con una identidad física.
  - Validación no necesaria.

- No solo las personas tienen identidad digital, también hay diferentes agentes que puede tener identidad.
  - Empresas, organismos: Seguridad Social, Agencia Tributaria, Google.
  - Servicios: Gmail.
  - Clientes, servidores: `host01.example.org`
- La identidad digital con respecto a un dominio (por ejemplo Google) no tiene por que ser válida en otro dominio (por ejemplo, el IFCA).
- Diferentes soluciones para solucionar este problema.

## Definición

Proceso mediante el cual se declara que alguien (o algo) es alguien (o algo).

- La identificación no es igual a autenticación.
- Ejemplo: llamada de teléfono: "Hola, soy Álvaro".
- No hay pruebas de que quien habla sea realmente Álvaro.

## Definición

Proceso mediante el cual se establece que alguien (o algo) es quién (o qué) realmente afirma ser.

- Implica una identificación previa.
- Implica un intercambio de credenciales de forma que se pueda asegurar que la identificación es válida.
- Normalmente basado en alguna de estas tres credenciales (o en una combinación)
  - Algo que se sabe (password).
  - Algo que se tiene (tarjeta de coordenadas, tarjeta criptográfica, certificado digital, etc.).
  - Algo que se es (biométrica).

## Definición

Proceso mediante el cual se establece qué nivel de acceso a un sistema tiene un usuario (identificado, autenticado o no) determinado.

- Puede existir autorización sin autenticación (por ejemplo usuarios anónimos).
- Ejemplo: usuarios autenticados en Moodle reciben diferentes roles (alumno, profesor, administrador).



# Autenticación biométrica

¿Una mala idea?

- Más fácil, no hay que recordar contraseñas. Más rápido, no hay que teclear nada.

# Autenticación biométrica

¿Una mala idea?

- Más fácil, no hay que recordar contraseñas. Más rápido, no hay que teclear nada.
- ¿Qué sucede con el lugar donde se almacena esa información? (por ejemplo un teléfono móvil?)

# Autenticación biométrica

¿Una mala idea?

- Más fácil, no hay que recordar contraseñas. Más rápido, no hay que teclear nada.
- ¿Qué sucede con el lugar donde se almacena esa información? (por ejemplo un teléfono móvil?)
- ¿Qué sucede si esa información se ve comprometida?

# Autenticación biométrica

¿Una mala idea?

- Más fácil, no hay que recordar contraseñas. Más rápido, no hay que teclear nada.
- ¿Qué sucede con el lugar donde se almacena esa información? (por ejemplo un teléfono móvil?)
- ¿Qué sucede si esa información se ve comprometida?
- ¿Qué sucede con nuestra información biométrica ya almacenada en otros sistemas?

# Autenticación biométrica

¿Una mala idea?

- Más fácil, no hay que recordar contraseñas. Más rápido, no hay que teclear nada.
- ¿Qué sucede con el lugar donde se almacena esa información? (por ejemplo un teléfono móvil?)
- ¿Qué sucede si esa información se ve comprometida?
- ¿Qué sucede con nuestra información biométrica ya almacenada en otros sistemas?
- La información biométrica se puede robar de infinitas maneras, y es **imposible** de cambiar.

# Autenticación biométrica

¿Una mala idea?

- Más fácil, no hay que recordar contraseñas. Más rápido, no hay que teclear nada.
- ¿Qué sucede con el lugar donde se almacena esa información? (por ejemplo un teléfono móvil?)
- ¿Qué sucede si esa información se ve comprometida?
- ¿Qué sucede con nuestra información biométrica ya almacenada en otros sistemas?
- La información biométrica se puede robar de infinitas maneras, y es **imposible** de cambiar.
- Lecturas:
  - Wired, Biometrics Are Coming, Along With Serious Security Concerns: <https://www.wired.com/2016/03/biometrics-coming-along-serious-security-concerns/>
  - CSO, The dark side of biometric identification: <https://www.csoonline.com/article/3060856/data-protection/the-dark-side-of-biometric-identification.html>
  - Digicert, Biometric Authentication: An Added Layer of Security or Security Risk? <https://www.digicert.com/blog/biometric-authentication-methods/>

# Identidad basada en certificados X.509

- Basado en criptografía de clave pública privada (PKI).
- Todos los agentes que necesitan identificarse tienen un certificado.
- Un certificado certifica la posesión de una llave pública asignada a un determinado subject (ya sea persona, host, servicio, etc.).

```
subject=DC = org, DC = terena, DC = tcs, C = ES, O = Consejo Superior de Investigaciones Cientificas
, CN = ALVARO LOPEZ GARCIA 594796@csic.es
subject=/businessCategory=Government Entity/1.3.6.1.4.1.311.60.2.1.3=ES/serialNumber=Government
Entity/street=Serrano, 117/postalCode=28071/C=ES/L=Madrid/O=Consejo Superior de Investigaciones
Cientificas/OU=IFCA/CN=portal.cloud.ifca.es
```

- Un certificado viene firmado por una Autoridad de Certificación (CA).
- La autenticación consiste en:
  1. Comprobar si el certificado ha sido expedido o firmado por una CA de confianza.

```
issuer=DC = es, DC = irisgrid, CN = IRISGridCA
```

2. Comprobar si el certificado ha expirado.

```
notBefore=May 3 11:19:23 2016 GMT
notAfter=May 3 11:19:23 2017 GMT
```

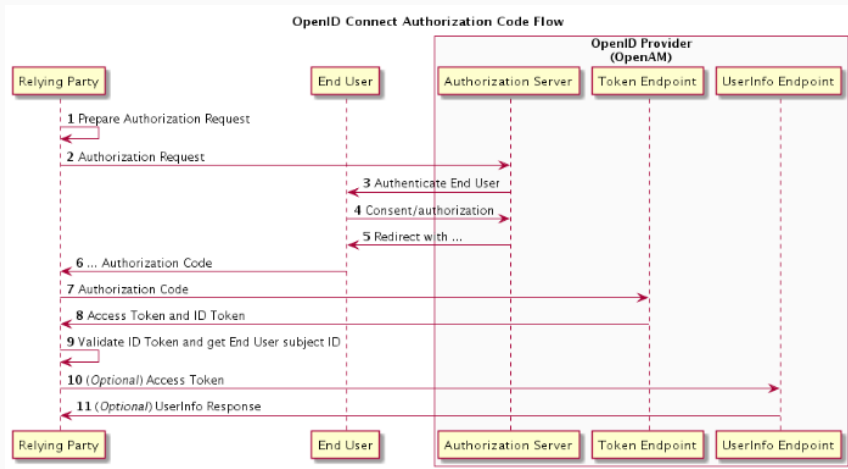
3. Comprobar si el certificado ha sido revocado (CRL, OCSP).
4. Comprobar si la otra parte tiene una prueba de posesión.

- Un usuario puede tener diferentes identidades, en diferentes sistemas, aislados entre si.
- Hay que manejar varias identidades.
- ¿Cómo hacer que un usuario pueda utilizar y linkar las mismas credenciales en diferentes sistemas?
- Diferente a Single Sign On: utilizar las mismas credenciales en diferentes sistemas (solo autenticación).
- La federación es un problema más amplio.
  - Gestión independiente, pero interoperable, de identidades en diferentes sistemas.
  - Manejo de atributos, traducción de credenciales, etc.



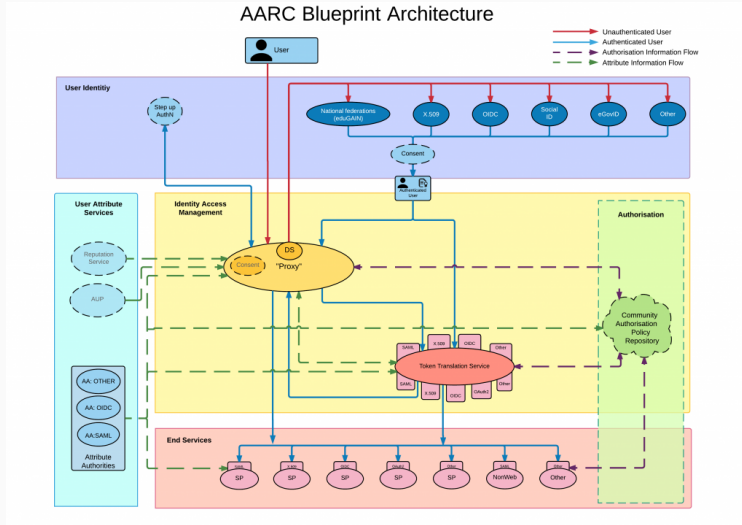
- SAML: Security Assertion Markup Language
  - Lanzado en 2005. Basado en XML. Creado por OASIS.
  - Protocolo de autenticación y autorización.
  - Ejemplos: EduGAIN.
- OAuth 2.0
  - Lanzado en 2006. Basado en JSON. Creado por Google y Twitter.
  - Protocolo de autorización, enfocado a autorización entre aplicaciones.
  - Ejemplos: Google OAuth 2.0, Twitter, etc.
- OpenID Connect
  - Lanzado en 2014, basado en JSON, API REST.
  - Protocolo de autenticación, construido sobre OAuth 2.0 (autorización).
  - Un usuario puede utilizar sus credenciales, manejadas por un IdP, para autenticarse en diferentes servicios.
  - Ejemplos: Google OpenID Connect.

# OpenID Connect



# Federación de identidades

AARC: Authentication and Authorisation for Research and Collaboration



Preguntas?