Identificación, autenticación y control de acceso

Antoni Martínez-Ballesté
Agustí Solanas
Jordi Castellà-Roca

PID_00177507

UOC
Universitat Oberta de Catalunya

Índice

Introducción

La proliferación de los sistemas informáticos ha ocasionado que la identidad de los usuarios se haya convertido en un factor clave estrechamente relacionado con la seguridad. Ya en los primeros sistemas, que sólo admitían su uso por parte de un operador, se debía controlar el acceso a la máquina y las acciones que se hacían.

El desarrollo de sistemas *mainframe* con múltiples terminales posibilitó el trabajo multiusuario: cada usuario creía disponer de la máquina en su totalidad, cuando en realidad los recursos estaban siendo repartidos en tiempo y espacio entre todos los usuarios. En ese momento se hizo necesario el desarrollo de sistemas de identificación que en la medida de lo posible estuviesen estrechamente ligados con el sistema operativo del ordenador central. Cada usuario debía identificarse en la propia máquina para poder hacer uso de sus recursos. Además, la identificación permitió monitorizar quién usaba cada recurso y en qué momento. Ahora bien, ¿quién aseguraba que el usuario conectado al ordenador era realmente quien decía ser? La autenticación es esencial para que la identificación tenga sentido y sea útil.

La universalización de servicios accesibles remotamente basados en Internet ha hecho crecer la importancia de la autenticación de la identidad. Y aún es más importante en cuanto Internet es una plataforma que permite realizar acciones que implican dinero (comprar por Internet, presentar la declaración de la renta, etc.) o un control de la privacidad del usuario (redes sociales, documentos en la nube, etc.). En este sentido, la firma digital y las tecnologías relacionadas juegan un papel clave en la autenticación segura.

En este módulo estudiaremos los distintos métodos de identificación y su autenticación, desde las ya clásicas contraseñas hasta los certificados digitales, llegando hasta los dispositivos seguros de identificación y apuntando el porvenir de la identificación por medio de la biometría. También se dará una visión de las técnicas de control de acceso y sus metodologías de implantación.

En el módulo estudiaremos las técnicas básicas de autenticación y control de acceso. Sobre la autenticación de la identidad de los usuarios, se estudiará en qué consiste la autenticación y se expondrá una panorámica de los distintos tipos de medios que se pueden utilizar. También estudiaremos conceptos relacionados con el ciclo de vida de la identidad digital, desde la creación de esta hasta su baja, pasando por detalles a tener en cuenta a la hora de utilizar las herramientas. El módulo se centra, en general, en las tecnologías usadas para autenticación en escenarios basados en Internet. Finalmente, se realiza

una panorámica de los sistemas de control de acceso, estudiando las fases de un sistema de control de acceso, las políticas de acceso y los tipos de control de acceso.

Objetivos

Los objetivos que el estudiante habrá alcanzado al finalizar este módulo son:

1. Comprender qué es la autenticación de la identidad.

2. Conocer técnicas y conceptos relacionados con las contraseñas.

3. Comprender el funcionamiento del certificado electrónico.

4. Conocer los dispositivos de usuario que se pueden utilizar en la autenticación.

5. Conocer las técnicas que proporciona la biometría para la identificación y autenticación.

6. Conocer la estructura del DNI electrónico.

7. Comprender los pasos necesarios para dar de alta usuarios de un servicio telemático.

8. Comprender los mecanismos de autenticación en un servicio telemático.

9. Conocer el procedimiento de la baja de usuarios de un servicio telemático.

10. Comprender qué es el control de acceso.

11. Conocer las fases y políticas del control de acceso.

12. Comprender el funcionamiento de los principales tipos de control de acceso.

1. Técnicas de identificación y autenticación

La identificación digital forma parte indisoluble de la mayoría de servicios en Internet y las TIC. Por ejemplo, para colgar un vídeo en un servidor, se pide que el usuario esté dado de alta en el servicio y se identifique para poder llevar a cabo la publicación del contenido. Por otra parte, para utilizar una red social, es preciso que el usuario esté registrado. Asimismo, los contactos de esta red también deben estar convenientemente identificados. Para realizar acciones tan variadas como hacer un pago mediante tarjeta de crédito, utilizamos esta misma tarjeta para identificarnos. O bien para hacer gestiones bancarias a través de Internet, lo primero que haremos es especificar quién somos.

Login

Para hacer referencia al identificador de usuario se utiliza el término *login*.

Esta identificación digital puede ser relativamente sencilla. Basta con disponer de un nombre de usuario, usar como identificador la dirección de correo electrónico o, en el caso de un pago, usar el número de tarjeta de crédito. Ahora bien, para la mayoría de servicios, además de la identificación digital, es necesaria una autenticación de esta identidad.

Mediante la autenticación de la identidad, el servicio se asegura de que el usuario es quien dice ser.

El concepto de "quien dice ser" lo resuelve el identificador de usuario: mediante una cadena de caracteres se denota cuál es la identidad del usuario. Y para demostrar la autenticidad de la identidad del usuario se pueden usar cuatro aproximaciones distintas:

1) El usuario es quien dice ser si demuestra conocer algo que solamente este conoce. Por ejemplo, conoce una palabra secreta de acceso.

2) El usuario es quien dice ser si posee algún objeto, como por ejemplo una tarjeta magnética. Un ejemplo no relacionado con la informática podrían ser las llaves de casa: en principio, es el propietario quien las posee.

3) El usuario es quien dice ser si posee alguna característica física que sólo él tiene: por ejemplo, la huella dactilar.

4) El usuario es quien dice ser si es capaz de hacer algo de forma única: por ejemplo, el patrón de escritura o la forma de andar.

Pese a tratarse de cuatro formas de abordar la autenticación de la identidad, no existe una frontera clara entre algunas de ellas. Además, es perfectamente posible el uso de varias de estas técnicas de forma combinada, para conseguir mayores grados de seguridad.

Por ejemplo, disponer de una tarjeta con códigos de seguridad para permitir operaciones bancarias a través de Internet (tarjeta de coordenadas) podría verse como una mezcla entre los dos primeros casos: el usuario posee la tarjeta, pero podría decirse que es conocedor de una información, aunque en este caso la tenga escrita.

Por otra parte, todas estas formas de abordar la autenticación no están exentas de problemas de seguridad. Por ejemplo, la tarjeta con códigos podría ser robada y usada por otro usuario. Otro caso podría ser la obtención de una clave de acceso mediante un correo electrónico fraudulento.

La usurpación de identidad consiste en que una entidad use con éxito el mecanismo de identificación que identifica a otra identidad.

En este apartado se exponen las técnicas y tecnologías para implantar la identificación y autenticación de usuarios.

1.1. Contraseñas

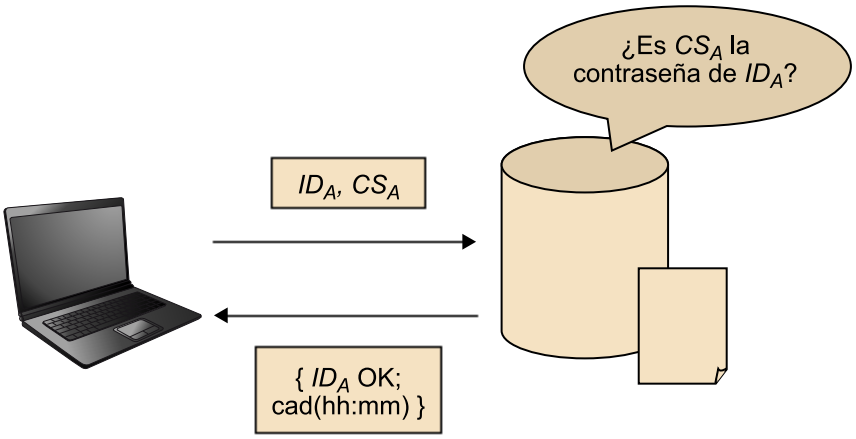
La autenticación por medio de contraseñas es relativamente sencilla: el usuario A envía su identificador ID_A y acto seguido su contraseña CS_A. La implementación del protocolo de identificación puede precisar que ambas informaciones se manden en un mismo mensaje, o bien que primero se pida el nombre de usuario y después la contraseña.

Esta contraseña es usada por el servicio para validar la identidad del usuario. Si A es el único conocedor de CS_A, es altamente probable que el usuario sea realmente A. En la figura siguiente, el usuario utiliza su contraseña para validarse al servicio. Este dispone de una lista de pares con usuarios y contraseñas para comprobar las identidades. Ante una contraseña correcta, el servicio manda un mensaje de autenticación correcta al equipo del usuario. Nótese que este incluye una hora de caducidad de esta autenticación, tras la cual el usuario deberá volver a autenticarse.

Ved también

En el apartado 2 estudiaremos variantes en el uso de las contraseñas, distintas formas de almacenamiento y gestión, así como algunas políticas de seguridad.

Esquema de funcionamiento de autenticación mediante contraseña



La contraseña es una cadena de caracteres de longitud arbitraria. En algunos escenarios, se usa una versión reducida de la contraseña, formada tan sólo por unos pocos números. En este caso la contraseña se conoce con el nombre de PIN[1]. De todos modos, en algunos entornos se utiliza el término PIN cuando en realidad la contraseña está formada tanto por letras como por números.

[1]PIN son las siglas de *personal identification number*.

Así como un mismo servicio o sistema no permite que haya dos usuarios con el mismo identificador, es perfectamente posible que dos o más usuarios tengan la misma contraseña.

1.2. Certificados electrónicos

Los sistemas de **criptografía de clave pública** representan otra forma de autenticación ante un servicio. La firma electrónica o los certificados electrónicos son dos de las utilidades que sirven a modo de autenticación de identidad. A continuación, revisamos algunos conceptos básicos sobre criptografía de clave pública.

1.2.1. Sistemas de clave pública

Los sistemas de criptografía de clave pública (o asimétricos) se basan en el uso de dos claves: la **clave privada** o secreta, que sólo conoce el propietario, y la **clave pública**, la cual, como su nombre indica, puede ser conocida por otras entidades sin que esto tenga consecuencias en la seguridad del sistema. En cambio, la clave privada debe estar perfectamente custodiada. Los matemáticos, criptógrafos y organismos han propuesto sistemas y estándares de sistemas de clave pública. Estos criptosistemas están basados en funciones matemáticas con trampa, que permiten realizar una operación en un sentido fácilmente, pero su inversa no es posible computacionalmente calcularla sin una información extra (trapa). Es decir, con la clave pública se realiza una operación que únicamente se puede invertir con la clave privada, y viceversa.

Protección de las claves privadas

Las claves privadas se guardan protegidas en el software mediante una contraseña, o bien en un dispositivo seguro como una tarjeta inteligente (al que también se accederá a través de una contraseña).

Ved también

El concepto de tarjeta inteligente se estudiará en el subapartado 1.3 de este mismo módulo.

El uso de la clave pública o la clave privada depende de la operación a realizar:

- Para dar **confidencialidad** a un mensaje que *A* (emisor) envía a *B* (receptor), es decir, que ningún otro usuario pueda conocer el contenido del mensaje, *A* usará la clave pública de *B* (que cualquier usuario puede obtener). Cuando *B* reciba el mensaje, usará su correspondiente clave privada para acceder a su contenido. Es recomendable que la clave privada esté protegida y sea necesario proporcionar una contraseña para acceder a ella.

- Para dar **autenticidad** a un mensaje que *A* manda a *B*, lo que se conocería como firma electrónica. *A* usa su clave privada para firmar el mensaje, esta operación solo la puede realizar el poseedor de la clave privada. Cuando *B* reciba el mensaje comprobará la validez de la firma. En primer lugar deberá obtener la clave pública de *A*, y a continuación realizará la operación inversa a la firma (verificación).

La firma electrónica permite alcanzar tres propiedades relacionadas con la seguridad de la información. Mediante la firma electrónica de un mensaje (o fichero) *M* mandado de *A* a *B*:

1) Se asegura que nadie ha modificado el mensaje firmado. Asegura, pues, la **integridad** del mensaje. La modificación de un único bit de la firma o del mensaje firmado dará lugar a una verificación incorrecta.

2) Se asegura que el mensaje lo ha firmado *A*, es decir, **asegura la identidad** del firmante (o **autenticación de origen** de *M*). La operación únicamente la puede realizar el propietario de la clave privada.

3) Ante cualquier tercera parte, al ser *A* el único que conoce su clave privada, se podrá demostrar que fue *A* quien firmó *M*. O dicho al revés, *A* **no podrá repudiar** la firma electrónica realizada del mensaje. Únicamente *A* puede realizar esta operación (firma).

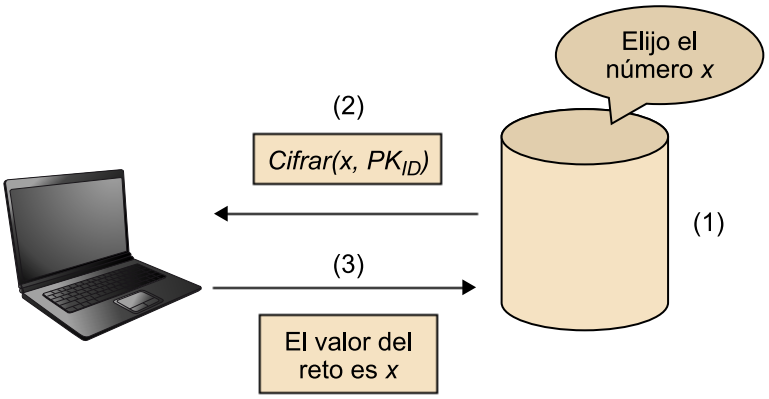
El hecho de no compartir una clave entre quien cifra y quien descifra, cosa que ocurre en los sistemas de clave compartida (o simétricos), proporciona estas dos últimas características de seguridad. Así pues, los sistemas de clave compartida no garantizan completamente la identidad y el no repudio.

> La firma electrónica garantiza las propiedades de integridad, identidad y no repudio.

La criptografía de clave pública permite un sencillo sistema de autenticación: el **sistema reto-respuesta**. El servicio obtiene la clave pública de la identidad que se autentica (PK_{ID}). Entonces, el servicio elige un número al azar, lo cifra con PK_{ID} y manda este reto a la identidad. Si la identidad es quien dice ser,

debería tener acceso a la clave privada con la que descifrar el reto, con lo cual deberá poder mandar de vuelta el número aleatorio con que el servicio lo ha retado.

Esquema del sistema de autenticación reto-respuesta



Otras formas de reto-respuesta

También se puede realizar un reto-respuesta mediante una clave secreta compartida. Este caso es habitual con el uso de *tokens*, que estudiaremos más adelante.

El uso de la firma electrónica y en general de los sistemas de clave pública plantea dos grandes problemas, que trataremos a continuación. Por una parte, cuando se obtiene la clave pública de una entidad (correo electrónico, servidor de Internet...), ¿cómo se asegura que realmente corresponde a la entidad identificada? Por otra parte y dado que cada día son más los procesos comerciales, administrativos, fiscales, etc. que pueden realizarse telemáticamente, ¿qué amparo legal tienen los sistemas de firma electrónica?

1.2.2. Autenticidad de la clave pública

Asegurar que una clave pública corresponde a una entidad se lleva a cabo mediante un **certificado electrónico**.

El objetivo de un certificado electrónico (o digital) es dar fe de la vinculación de una clave pública a una identidad, como un usuario o un servicio.

Ejemplos de certificado electrónico

Cuando nos conectamos a un sitio web seguro se obtiene un certificado cuya misión es autenticar la identidad del servidor. Por otra parte, cuando realizamos una firma electrónica en un documento, esta firma se acompañará, en general, de un certificado que de fe de la validez de la clave pública que se deberá usar para comprobar la firma.

La información básica que contiene un certificado es:

- La identidad que se certifica, por ejemplo una dirección de correo electrónico o el número de DNI de un ciudadano.
- El período de validez, es la fecha a partir de la cual el certificado no será reconocido como válido.
- La clave pública que se certifica, que se usará directamente para comprobar la firma.

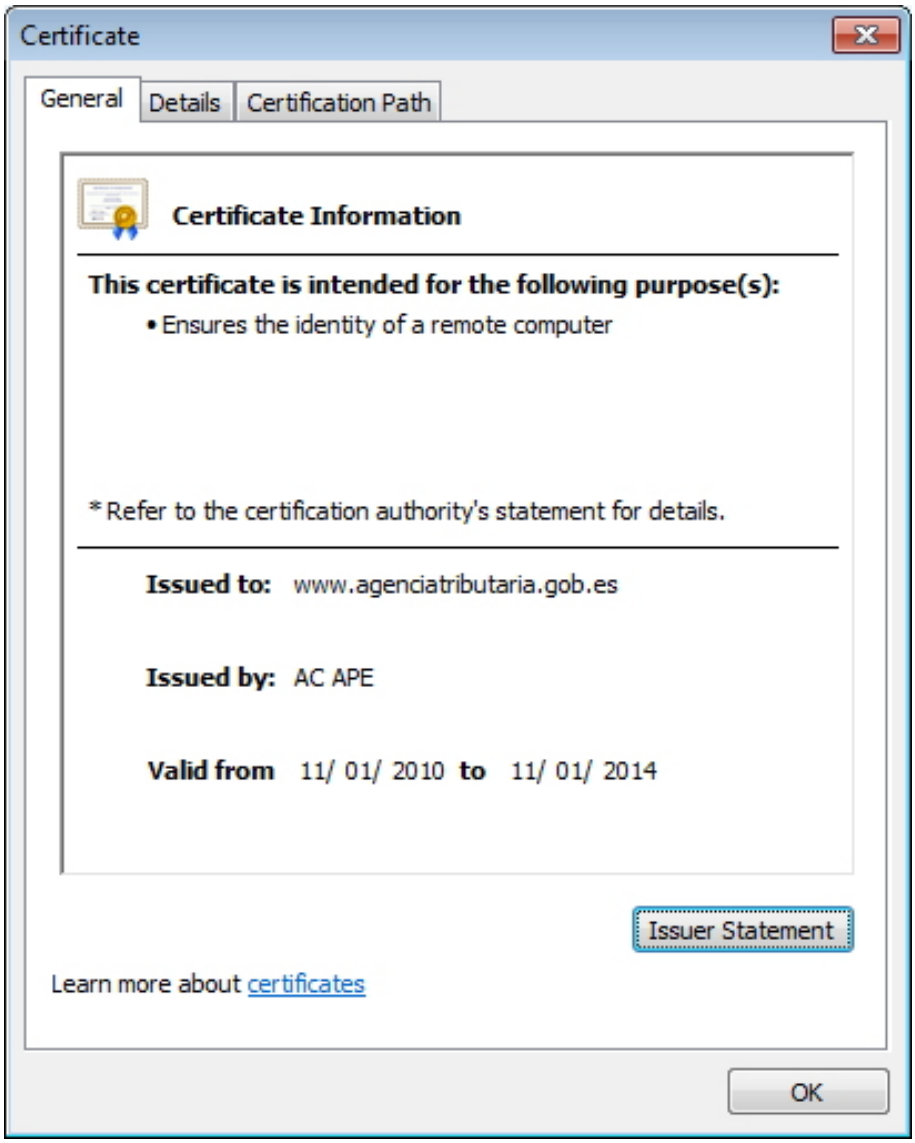
- El nombre del emisor del certificado. En general, se trata de una **autoridad de certificación**, un organismo que puede expedir certificados de clave pública. Esta autoridad da fe de que la clave pública pertenece a la identidad especificada en el certificado.

Las autoridades de certificación disponen de una plataforma telemática (infraestructura de clave pública, PKI) que permite generar y gestionar claves y certificados.

Este último elemento es crucial para el funcionamiento de los sistemas de clave pública. Hemos visto que una clave pública está autenticada mediante un certificado. Pues para confiar en un certificado se debe confiar en su emisor. Esto es, si el emisor no es de confianza, el certificado se deberá rechazar. Si por el contrario el emisor es de confianza, el certificado es aceptado y, en consecuencia, se dará por buena la clave pública que certifica.

La figura siguiente muestra las propiedades del certificado de servidor de la Sede Electrónica de la Agencia Tributaria. El objetivo es asegurar la identidad del servidor www.agenciatributaria.gob.es. Puede observarse su periodo de validez y quien emite el certificado (AC APE, Agencia de Certificación Administración Pública Española).

Certificado de la Agencia Tributaria para sus servicios web



Ahora bien, la discusión puede estar en qué significa confiar en un emisor. Un sistema informático de usuario (ordenador, navegador, etc.) dispone, por defecto, de unas entidades emisoras de certificados de confianza. En este sentido, los sistemas tienen un almacén de certificados en el cual hay una colección de entidades de confianza. Si se recibe un certificado que ha sido emitido por una autoridad de confianza, este se acepta sin más (de hecho, el usuario no es ni consciente de este hecho). La dificultad surge cuando el emisor del certificado no está en la lista de autoridades de confianza del sistema. En este caso, suele mostrarse un mensaje de información y es el usuario quien debe decidir si aceptar al emisor del certificado recibido como una nueva autoridad de confianza (a partir de entonces, los futuros certificados que se reciban y hayan sido emitidos por esta autoridad ya serán aceptados como válidos, sin preguntar al usuario).

Las grandes empresas certificadoras internacionales, así como organismos certificadores dependientes de los gobiernos ya son reconocidas por la mayoría del software criptográfico.

1.2.3. Legalidad de la firma electrónica

Para realizar aplicaciones que permitan una firma electrónica en la Unión Europea, es necesario tener en mente la legislación pertinente, mediante la cual saber con qué firmas se debe trabajar, qué características harán una firma legal en un determinado momento y qué datos se deben verificar.

Los países de la Unión Europea se rigen por la Directiva 1999/93/CE de 13 diciembre, por la que se establece un marco comunitario para la firma electrónica. Posteriormente, cada país miembro realiza la transposición de la Directiva dentro de su legislación de la forma que vea más conveniente.

En particular, dentro de la legislación española, se regula la firma electrónica en la Ley 59/2003, de 19 de diciembre, de firma electrónica, donde define dos tipos de firma electrónica en el título I: "Disposiciones generales, artículo 3. Firma electrónica y documentos firmados electrónicamente":

1) La **firma electrónica** es el conjunto de datos en forma electrónica, consignados junto a otros o asociados con ellos, que pueden ser utilizados como medio de identificación del firmante. Por ejemplo un PIN o una firma electrónica. Este tipo de firma no tiene efectos jurídicos totales. No es suficientemente segura, ya que no se garantiza que haya sido creada por el supuesto firmante; puede tratarse de una firma reproducible por un usuario malintencionado.

2) La **firma electrónica avanzada** permite identificar al firmante y detectar cualquier cambio ulterior de los datos firmados; está vinculada al firmante de manera única y a los datos a que se refiere, siendo creada por medios que el firmante puede mantener bajo su exclusivo control. Puede ser considerada no suficientemente segura si su algoritmo es débil o si ha habido un compromiso de la clave privada –por lo que la firma no habrá sido realizada por el firmante–, o si puede ser reproducible (es decir, se pueden realizar ataques por fuerza bruta para crear firmas). Un tipo específico de firma electrónica avanzada es la reconocida.

Una **firma electrónica reconocida** es una firma electrónica avanzada basada en un certificado reconocido y generada mediante un dispositivo seguro de creación de firma. Se equipara totalmente a la firma manuscrita.

Nota

El DNI electrónico permite una firma electrónica reconocida.

Cabe mencionar también la Decisión de la Comisión de 14 de julio del 2003, relativa a la publicación de los números de referencia de las normas que gozan de reconocimiento general para productos de firma electrónica.

Los diferentes tipos de firma electrónica vienen definidos, dentro de la Unión Europea, según la Norma ETSI TS 101 733 CMS *Advanced Electronic Signatures* (CAdES), o su equivalente en XML ETSI TS 101 903 XML *Advanced Electronic Signatures* (XAdES):

- **Firma electrónica básica**
- **Firma electrónica avanzada**. Su implantación se realiza como ASN.1/ CMS o S/MIME, o XMLDSig.
- **Firma electrónica con sellado de tiempo** (*timestamping*). Se incorpora a la firma la fecha en la que se realiza.
- **Firma electrónica completa**. Incluye datos de verificación. Equivalente a la firma electrónica reconocida dentro de la legislación española.
- **Firma electrónica extendida**. Incluye datos de verificación con sellado de tiempo.

Organizaciones de estandarización

Las organizaciones encargadas de la estandarización de los formatos de firma electrónica y recomendaciones son: el European Telecommunications Standards Institute (ETSI), que es la encargada oficial dentro de la Unión Europea, la Internet Engineering Task Force (IETF), a nivel internacional, y la World Wide Web Consortium (W3C), también a nivel internacional.

Firma de documentos ofimáticos y PDF

Los documentos PDF (*portable document format*) pueden ser firmados electrónicamente, al igual que otros formatos de documento ofimático. Utilizan estructuras propias para guardar la información de la firma.

Los estándares más comunes en firma electrónica son:

- PKCS#7 (Public Key Cryptographic Standard). Definido en la RFC 2315.
- IETF RFC 2630/ RFC 3369 (CMS, Cryptographic Message Syntax). Sintaxis ASN.1. Basada en PKCS#7.
- ETSI TS101733. CadES. Estándar europeo basado en CMS para firmas ASN.1.
- Estándares de firma basados en XML más comunes:
  - Firma XML, XML-Dsig, Dsig XML o XML-Sig. Recomendación del W3C que define una sintaxis XML para la firma electrónica. Basada en PKCS#7. Se utiliza en tecnologías web como SAML. Define un formato para transportar firmas electrónicas.
  - XadES: extensión de XML-Dsig definida por el ETSI TS101903.Es el estándar europeo basado en XML-Dsig para firmas XML.

1.2.4. Firmas XML

Gracias a la facilidad que da XML para estructurar la información, el hecho de ser libre de licencias y su independencia con la plataforma, repercute en que esta tecnología sea una de las más extendidas para firmar documentos generados en web. El estándar de referencia es XML, *advanced electronic signatures*, más conocido por su acrónimo, XAdES, según la Directiva Europea para Firma Electrónica, por lo que define formatos de firma que puedan ser válidos para realizar transacciones.

Existen varios tipos de XAdES que implementan una XML-Dsig, según el nivel de protección ofrecido. A continuación se definen algunos de los niveles de protección de XAdES, del más básico al más extenso, donde cada uno incluye y extiende al previo:

- **XAdES-BES** (*basic electronic signatura*). Define la forma básica de una firma electrónica. Tiene que contener el elemento *SigningCertificate* o *ds:KeyInfo*.

- **XAdES-EPES** (*explicit policy based electronic signatura*). Añade información sobre políticas de firma mediante la propiedad *SignaturePolicyIdentifier*.

- **XAdES-T**. Añade un campo de sellado de tiempo a través de *SignatureTimeStamp* o de un proveedor de tiempo de confianza.

- **XAdES-C**. Añade referencias a datos de verificación (certificados y listas de revocación) a los documentos firmados para permitir verificación y validación *off-line*.

- **XAdES-X**. Firma extendida con sellos de tiempo a las referencias introducidas por XAdES-C para evitar que pueda verse comprometida en el futuro una cadena de certificados, mediante las propiedades *CompleteCertificateRefs* y *CompleteRevocationRefs*.

- **XAdES-X-L**. Firma extendida a largo plazo añadiendo los propios certificados y listas de revocación a los documentos firmados, para permitir la verificación en el futuro incluso si las fuentes originales (de consulta de certificados o de las listas de revocación) no estuvieran ya disponibles a través de las propiedades *CertificateValues* y *RevocationValues*.

- **XAdES-A**. Firma archivada que añade la posibilidad de *timestamping* periódico de documentos archivados por medio de las propiedades *CertificateValues*, *RevcationValues* y al menos un *ArchivedTimeStamp*. Esta firma previene que la firma del documento pueda ser comprometida posteriormente a causa de la debilidad de la firma.

Ved también

En el subapartado 2.3.2 de este módulo estudiaremos en qué consisten las listas de revocación.

El W3C es el encargado de realizar las recomendaciones y definir la sintaxis XML para gestionar firmas electrónicas, permitiendo así que una firma XML pueda firmar cualquier tipo de documento.

Existen varios tipos de firma XML, entre las que remarcamos los tres tipos más comunes:

- ***Enveloping signatures***[2]. La firma engloba el contenido que se encuentra dentro de un elemento Object de la misma firma. El objeto (o su contenido) es identificado a través de una referencia (a través de un identificador de fragmento URI o transformación).

(2)En castellano, firma envolvente.

- ***Enveloped signatures***[3]. La firma engloba el contenido XML que contiene la firma como un elemento. El contenido proporciona el elemento raíz del documento XML. Obviamente, las firmas envueltas deben tener cuidado de no incluir su propio valor en el cálculo de la *SignatureValue*.

(3)En castellano, firma envuelta.

- ***Detached signatures***[4]. Puede ser de dos formas: o bien la firma es sobre un documento externo, o bien la firma es sobre parte del documento.

(4)En castellano, firma separada.

Estructura básica del elemento `<Signature>`

```
<Signature ID?>
    <SignedInfo>
    <CanonicalizationMethod />
    <SignatureMethod />
    (<Reference URI? >
    <Transforms>)?
    <DigestMethod>
    <DigestValue>
    </Reference>)+
    </SignedInfo>
    <SignatureValue />
    (<KeyInfo />)?
    (<Object ID?>)*
</Signature>
```

Para la implementación del firmado de un documento mediante XML se debe tener en cuenta que las representaciones de documentos firmados usados por la aplicación que realiza la firma y la que la verifica deben ser totalmente idénticas. De hecho, es totalmente posible que dos aplicaciones distintas obtengan, cada una, un documento distinto al otro, pero ambos válidos, debido tal vez a la alteración del orden de los atributos, a los posibles espacios en blanco en las etiquetas, a los elementos vacíos, etc. Para asegurar que no haya

problema, W3C define una **forma canónica** para cualquier documento XML, eliminando así posibles ambigüedades en la generación de la representación de un documento. El elemento `<SignedInfo>` y el documento XML deben pasar por este proceso. Con la canonización, aseguramos que el documento se encuentra codificado en UTF-8, no existen saltos de líneas, hay todos los atributos por defecto de cada elemento, etc.

1.3. Dispositivos de usuario

Hasta el momento, la gran mayoría de servicios de Internet usan la contraseña como herramienta de autenticación. Las redes sociales o los servicios de correo electrónico más populares son algunos ejemplos. Los certificados digitales son generalmente usados para autenticar servicios que requieren de un nivel de seguridad elevado, como por ejemplo, la página web segura de un servicio bancario en línea. Sin embargo, en algunos servicios que funcionan sobre Internet ya es habitual que la autenticación de la identidad sea a través de un certificado de cliente.

En general, para usar este sistema de autenticación, se acabará usando una clave privada que estará albergada en el propio sistema o en un dispositivo externo. Esta clave secreta debe estar protegida (cifrada), y sólo se debe poder usar tras introducir una contraseña o PIN. A veces, la propia contraseña introducida por el usuario se usará como clave para descifrar la clave secreta y poder usarla.

En general, pues, los certificados y claves secretas se encuentran guardados en un equipo de trabajo. De todos modos, en determinados ámbitos, es más cómodo o incluso necesario el disponer de un dispositivo extraíble que sea el que proteja esta información.

Los *token* de seguridad son pequeños dispositivos cuyo objetivo es dar soporte al proceso de autenticación de identidad. En general se utilizan para validar la entrada a sistemas y servicios.

Hay gran variedad de *tokens*. Por ejemplo, en cuanto a conexión al equipo, los más habituales son los que se conectan al puerto USB, aunque también los hay que usan una conexión Bluetooth o simplemente no precisan de conexión al equipo. En cuanto a capacidad de memoria y proceso, dependerá del sistema de soporte que permita el *token*. He aquí cuatro casos combinando las anteriores alternativas:

Ejemplo de uso de *token*

Hay sistemas de autenticación que usan *token* entendiendo estos como ficheros mediante los cuales los usuarios pueden acceder a los servicios. Un ejemplo sería el sistema Kerberos, mediante el cual los usuarios obtienen, de forma transparente, ficheros con los que se autenticarán ante servicios.

- El *token* guarda un valor secreto en su memoria y se comunica mediante USB con un controlador instalado en el PC. El controlador sólo permite el acceso al contenido del *token* tras poner un PIN correcto. En el fondo,

la información que contiene la memoria va cifrada, y es preciso que el controlador la descifre.

- El *token* no va conectado al ordenador pero tiene un pequeño teclado numérico mediante el cual el usuario introduce una coordenada proporcionada por el sistema de identificación. El *token* muestra en una pequeña pantalla el valor secreto correspondiente a la coordenada solicitada. Este ejemplo sería análogo a la tarjeta de coordenadas comentada anteriormente.

- Un tercer caso sería un *token* con capacidades criptográficas que pudiera responder a un reto criptográfico mandado por el sistema de identificación, usando el método reto-respuesta. En este caso, se contempla la modalidad de reto-respuesta usando una clave secreta compartida o bien usando un sistema de cifrado de clave: la copia de la clave secreta que necesita el cliente está almacenada en el *token*.

- Un último caso sería el de una etiqueta RFID[5] que contiene un valor que permite identificar la etiqueta. Dado que un lector de RFID interactúa con las etiquetas a través del espectro electromagnético y sin usar conexiones físicas, el hecho de que la identificación sea segura (básicamente esté protegida contra escuchas por parte de atacantes) supone un reto importante inherente al uso de estas etiquetas en servicios y entornos que precisen cierta seguridad.

(5)RFID es la abreviatura de *radio-frequence identification*, identificación mediante radiofrecuencia.

Seguridad de las etiquetas RFID

El reducido rango de lectura fijado en los dispositivos RFID dificulta el ataque, aunque no lo elimina por completo. Un atacante puede acercarse a la víctima para estar dentro del rango de lectura del RFID y a continuación enviar esta información mediante una conexión de alta velocidad al sistema informático para autenticarse. Por este motivo, es necesario fijar medidas de seguridad que requieran la participación del propietario en la autenticación.

Sea como fuere, el *token* es un dispositivo independiente del ordenador y puede llevarse siempre consigo. Esto acarrea un claro problema de seguridad en caso de robo del *token*.

Ejemplos de dispositivos *token*. A la izquierda, *token* sin contacto; a la derecha, *token* USB. Fuente: Wikipedia

Otro elemento que puede considerarse *token* es la tarjeta inteligente. Este tipo de tarjetas se caracterizan por llevar un circuito integrado capaz de almacenar información e incluso hacer operaciones criptográficas. El uso de estas tarjetas

se popularizó gracias a su aceptación como medio de pago (suponen un nivel avanzado en seguridad respecto de las tarjetas bancarias con banda magnética, puesto que para poder usarlas es necesario conocer un PIN), y también por las tarjetas *subscriber identity module* (SIM), que tienen actualmente todos los teléfonos móviles. Además, en multitud de organizaciones e instituciones el uso de los certificados digitales y tarjetas inteligentes se ha convertido en algo habitual.

Claves de un certificado digital

El elevado uso de operaciones criptográficas debilita la seguridad del par de claves que componen un certificado digital. Mediante la creación de dos pares de claves privadas y públicas, se consigue reducir las posibilidades de debilitamiento de las claves. Uno de los pares de claves se utiliza para las operaciones de autenticación, y el otro para las de firma de documentos.

Seguridad de las tarjetas inteligentes

En inglés una tarjeta inteligente se llama *smart card*, mientras que en francés *carte à puce* (tarjeta con chip). Las tarjetas inteligentes son seguras contra manipulaciones. Si un atacante intenta acceder a la información que contiene la tarjeta, esta se bloquea y se elimina la información. Además, incorporan mecanismos de protección para evitar que se pueda obtener información de las claves a partir del consumo de energía o tiempo de respuesta.

Por otra parte, otro uso que ha contribuido a su generalización es que sea la plataforma del documento nacional de identidad electrónico (DNIe).

1.3.1. El DNIe

El DNIe es una tarjeta inteligente capaz de guardar de forma segura información y de procesarla internamente. Esta propiedad permite las acciones siguientes:

- Acreditar electrónicamente y de forma segura la identidad de la persona.
- Firmar digitalmente documentos electrónicos, otorgándoles una validez jurídica equivalente a la que les proporciona la firma manuscrita.

La tarjeta soporte del DNI electrónico contiene los datos de filiación del ciudadano, los datos biométricos (modelo dactilar, foto y firma manuscrita) y los dos pares de claves RSA con sus respectivos certificados digitales, uno de autenticación y otro de firma electrónica.

Para poder usar el DNIe, al igual que con otras tarjetas inteligentes, es necesario un lector de tarjetas, por ejemplo, el que muestra en la figura siguiente.

Imagen de un lector de tarjetas inteligentes con un DNIe introducido

Protección del DNIe

El DNIe mantiene el material criptográfico sensible siempre en su interior y protegiendo su uso gracias a un control de acceso mediante un número de identificación personal que sólo el usuario conoce.

La información en el chip está distribuida en tres zonas con diferentes niveles de seguridad y condiciones de acceso:

- **Zona pública**. La lectura de esta zona es accesible sin restricciones. Contiene, entre otros, el certificado del emisor y unas claves para cifrar el intercambio de información entre el chip y el dispositivo de lectura. Estas claves están certificadas mediante un certificado de componente.

- **Zona privada**. La lectura de esta zona está permitida al ciudadano, mediante la clave personal de acceso o PIN. Contiene un **certificado de autenticación** (que tiene, como finalidad, garantizar electrónicamente la identidad del ciudadano al realizar una transacción telemática) y un **certificado de firma**. Este último es el que se utilizará para la firma de documentos, garantizando la integridad del documento y el no repudio. Este certificado expedido como certificado reconocido y creado en un dispositivo seguro de creación de firma es el que convierte la firma electrónica avanzada en firma electrónica reconocida, permitiendo su equiparación legal con la firma manuscrita.

Certificado de componente

El certificado de componente permite la autenticación mutua de dispositivos tal y como se describe en el estándar CWA-14890.

- **Zona de seguridad**. Los ciudadanos pueden acceder a esta zona en los puntos de actualización del DNI electrónico. Contiene los datos impresos en el soporte físico del DNI, la fotografía y la firma escaneada.

Aunque la Administración pública ofrece multitud de trámites en línea, la realidad es que la gran mayoría de ciudadanos que disponen de DNI (ya en versión electrónica en casi su totalidad) todavía no son conocedores de sus utilidades o bien no disponen de los conocimientos técnicos adecuados para su uso.

1.4. Biometría

La biometría vendría a significar algo así como "medir" los rasgos "biológicos" de un individuo.

> La biometría es la aplicación de las matemáticas y la ciencia de la computación para identificar individuos de acuerdo con sus características o rasgos físicos.

Esta tecnología hace años que aparece como habitual sistema de autenticación en la literatura y películas de fantasía y ciencia ficción. De todos modos, el análisis del iris, la forma de la mano, incluso la manera de andar, etc. ya juegan un papel importante en sistemas de control de acceso reales. Además, es conocido que el uso de las huellas dactilares para la identificación de un individuo es ya un sistema clásico en organismos policiales.

En un sistema biométrico el individuo debe en primer lugar registrarse. Entonces una o más características físicas o de conducta son registradas por un sistema de información. Para la identificación del individuo, el sistema de información recoge las características del individuo que quiere identificarse para hacer una comprobación en su base de dados. Si la información recogida no coincide con ninguno de los individuos registrados, se deniega el acceso.

Si el número de individuos registrados es muy elevado, el sistema no buscará entre todos los individuos de su base de datos, ya que podría llegar a ser costoso y potenciador de errores. El individuo que quiere identificarse lo hace mediante un código (o cualquier otro identificador), mientras que el sistema biométrico comprueba que las características físicas recogidas para el individuo con el identificador concreto coinciden con las almacenadas. En este caso, el sistema biométrico realiza una autenticación de la identificación.

No obstante hay distintos problemas a los que hacer frente. El primero es el de la recogida de características: los sistemas deben obviar ruidos, imperfecciones o distintos matices que los sistemas captadores y sensores recogen a cada acceso de un individuo. En segundo lugar, se deben minimizar las tasas de errores del sistema de identificación: no se debe permitir la autenticación de un individuo no registrado o no válido y, al contrario, no se debe denegar la correcta autenticación de un individuo registrado o válido. En este sentido, la tasa de falso positivo[6] y la de falso negativo[7] mide el buen funcionamiento y rendimiento del sistema biométrico.

(6)En inglés, *false acceptance rate.*

(7)En inglés, *false nonmatch rate.*

A continuación enumeramos los rasgos biométricos más utilizados, comentando aspectos sobre su calidad y uso:

- **Huellas dactilares**. El uso de las huellas dactilares como medio de identificación es de alta fiabilidad. Tiene buena aceptación y popularidad.

- **Ojo**. La identificación de un individuo a través del análisis del iris tiene una fiabilidad muy alta. El problema es la facilidad de uso. El análisis de la retina para identificar al individuo es más complejo todavía.

- **Forma de la mano**. Este sistema está bastante extendido, pero quizás presenta un poco menos de fiabilidad que los anteriores. Este sistema es susceptible de padecer ataques, puesto que no resultaría difícil recrear la forma de una mano usando un molde.

- **Cara**. El análisis de la cara, ya sea en 2D o 3D, es un buen medio para identificar a un individuo. Aun así, siempre es preferible un estudio 3D, ya que se mejoran los resultados con respecto el mero estudio de una imagen 2D, y se dificulta el éxito de ataques. La desventaja de los equipos 2D es que el sistema no distingue si lo que está capturando es realmente un rostro o una fotografía de un rostro.

- **Venas del dedo o la mano**. El estudio vascular de los dedos o de la mano proporciona alta fiabilidad, además no es muy complejo su análisis.

- **Voz**. Este sistema presenta bastante fiabilidad, pero también es susceptible a los ataques. Además este sistema padece de poca estabilidad, con lo cual deberían ser varias las tomas de voz analizadas para reducir la tasa de errores.

Claramente, son varias las características biométricas que pueden usarse para identificar o autenticar. Estas pueden usarse individualmente o bien en combinación entre ellas, o bien procediendo a la autenticación mediante información adicional, como podría ser una contraseña. La figura siguiente muestra un lector de huella digital para ayudar a la identificación en un ordenador portátil.

Lector de huellas dactilares de un ordenador portátil

2. Ciclo de vida de la identidad digital

Una identidad digital pasa por tres pasos básicos en su ciclo de vida: la creación, el uso y la desaparición o baja. En este apartado estudiaremos aspectos relacionados con el alta, el uso y la baja de las identidades digitales, en concreto, las contraseñas y los certificados.

2.1. Alta de usuarios

El alta de un usuario es el procedimiento previo a que este pueda usar un servicio o bien acceder a un sistema. Existen dos formas mediante las cuales un usuario puede darse de alta en un sistema:

- **De forma presencial**. El individuo acude al gestor de identidades del sistema o del servicio (por ejemplo, al administrador de los equipos informáticos de una empresa) y este le proporciona la información necesaria para usar el servicio. En este caso el gestor de identidades ha identificado y autenticado al usuario de forma presencial. La presencialidad es importante en aquellos sistemas o servicios con potenciales riesgos.

Importancia de la presencialidad

Obtener un certificado digital o una tarjeta de coordenadas para operar con una entidad financiera pueden ser un par de situaciones en las que se requiera presencialidad.

- **De forma no presencial**. En este caso el individuo elige (o se le asigna) un nombre de usuario, y también una contraseña u otra información para su autenticación. En este caso, útil y habitual en aquellas situaciones que no haya grandes riesgos, será necesario realizar una serie de comprobaciones para mejorar la seguridad del proceso de alta de usuario.

2.1.1. Confirmación no presencial de la identidad

En la forma presencial, quien da de alta el identificador y demás información de autenticación precisará de la verificación visual de que el solicitante es quien dice ser. Sin embargo, tal y como se ha apuntado, en los sistemas no presenciales deben tomarse una serie de medidas para evitar situaciones como la suplantación de identidad o el acceso indebido a los sistemas.

Supongamos que un usuario quiere darse de alta en un servicio telemático para el cual será necesario introducir una contraseña como sistema de autenticación. El usuario introduce su identificador (del cual se comprobará que no haya otro igual en el sistema) y una contraseña. Se pueden usar estas herramientas para mejorar el alta de usuario:

- **Correo electrónico**. Se puede pedir al usuario que introduzca una dirección de correo electrónico. En esta dirección, se recibirá un correo que contendrá un enlace a una página que activa automáticamente la cuenta

de usuario recién creada. Ahora bien, supongamos el caso de que el servicio es una red social. Al no haber comprobación presencial, se podría usar una dirección de correo electrónico propia (con lo cual la activación tendría éxito), pero al mismo tiempo, estar creando un identificador para rellenar el perfil de red social de otro individuo.

- **Teléfono móvil**. Otro sistema podría consistir en pedir un número de teléfono móvil al cual el servicio mandará un código de activación. Claramente, como en el caso del correo electrónico, no se evita la posibilidad de fraude. No obstante, con la legislación actual se puede identificar al propietario del número del teléfono móvil.

Autenticación por móvil

El medio de autenticación a través del teléfono móvil es habitual para la confirmación de operaciones de compra a través de Internet.

- **Confirmación por datos**. Si el sistema telemático se corresponde con una entidad que ha mantenido relación con el usuario previamente, se pueden pedir datos para comprobar que la identidad del solicitante se corresponde con el individuo. Podría ser el caso de crear un acceso a la oficina virtual de una compañía eléctrica: se podría pedir, por ejemplo, una cantidad determinada de alguna de las facturas emitidas con anterioridad.

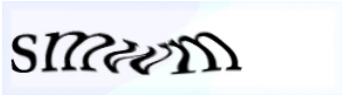
Finalmente, en muchos casos se pretende evitar el uso indebido del servicio (por ejemplo, un servicio de correo electrónico, un blog o un foro) o bien se quiere proteger el sistema ante bloqueo del servicio. En el primer caso, un programa podría automáticamente darse de alta en múltiples servicios para inundarlos de publicidad. En el segundo, el programa iría realizando peticiones automáticas y creando usuarios hasta colapsar el sistema. Para evitar que un programa use formularios pensados para que los rellene un humano, se usan los **captcha**.

Captcha

Captcha es el acrónimo de *completely automated public Turing test to tell computers and humans apart*, en castellano, prueba de Turing pública y automática para diferenciar máquinas y humanos.

Un captcha es una prueba que en principio sólo puede ser resuelta por humanos y no por programas, y cuyo objetivo es diferenciar a un humano de un programa.

La típica prueba es que el usuario debe introducir por teclado una palabra o conjunto de caracteres que se exponen por medio de una imagen distorsionada, borrosa, con ruido, etc., es decir, con cualquiera de las características que pueda dificultar la lectura automática por parte de un programa pero no para un humano.



Ejemplo de captcha

2.1.2. Contraseñas, códigos y recomendaciones

Si creamos un usuario, necesitaremos una contraseña. Si recibimos un certificado electrónico, es muy recomendable utilizar una contraseña o un código para proteger la clave privada correspondiente al certificado. Así pues, en este subapartado trataremos algunos conceptos sobre las contraseñas y demás códigos de autenticación.

En primer lugar, veamos qué estrategias tiene un atacante para intentar adivinar un código o contraseña que le permitan suplantar una identidad:

- **Fuerza bruta**. Consiste en probar todas las posibles combinaciones de símbolos válidos hasta dar con el valor correcto.

- **Búsqueda inteligente**. Se trata de buscar por un espacio de nombres restringido (por ejemplo, probar el número de teléfono del individuo al que se quiere suplantar). Uno de estos casos es el ataque de diccionario, consistente en buscar contraseñas a partir de un diccionario.

Afortunadamente, se dispone de una gran cantidad de consejos y técnicas para mejorar la seguridad de la contraseña:

- **Cambiar la contraseña por defecto**. Cuando en el proceso de alta el usuario no decide cuál es su contraseña, el sistema le proporciona una. Es altamente recomendable que el usuario la cambie a una que le sea más fácil de recordar, y que sea "nueva" tan pronto como le sea posible.

- **Aspecto de la contraseña**. Es aconsejable una longitud mínima, así como la combinación de letras (mayúsculas y minúsculas), números y a poder ser otros caracteres. Desafortunadamente, algunos sistemas limitan el conjunto de caracteres posible, o incluso limitan la longitud máxima de la contraseña. El objetivo de esta medida es aumentar el espacio de búsqueda y dificultar las posibles repeticiones de contraseñas.

- **No usar contraseñas obvias**. Es evidente, sobre todo para evitar los ataques de búsqueda inteligente.

Contraseñas obvias en Hotmail

Tras un ataque masivo, se revelaron cuáles son las contraseñas más usadas en Hotmail. La clave que apareció con más frecuencia fue "123456". La noticia puede leerse en la web de RTVE.

- **Usar medidores de seguridad de contraseña**. Muchos sistemas informan de la calidad de la contraseña elegida, con relación a la seguridad que ofrece. Estos sistemas se basan en las recomendaciones anteriores para decidir sobre la seguridad de la contraseña escogida.

- **Forzar periódicamente el cambio de contraseña**. Si la contraseña del usuario ha sido obtenida por algún atacante, el cambio periódico hace que la contraseña usurpada ya no tenga validez.

- **Permitir un número máximo de intentos fallidos**. Mediante esta técnica se limita claramente el éxito de los ataques de fuerza bruta. Cuando se llega al máximo de intentos se bloquea la cuenta.

- **Solicitar códigos de autorización en aquellas operaciones que precisen de más seguridad**. Estos códigos suelen encontrarse en una tarjeta de coordenadas específica del usuario, aunque también podría usarse un *token*. En este caso es recomendable que el código de autorización sea de un solo uso.

Los consejos anteriores son habituales en la mayoría de sistemas de validación por medio de contraseña. El administrador del sistema debe diseñar un proceso de gestión de contraseñas que tenga en cuenta estos aspectos. Sin embargo, aun tomando precauciones técnicas, el robo de contraseñas tiene otro problema de seguridad: el robo a través de un correo electrónico o llamada telefónica pretendiendo ser un administrador del sistema y pidiendo la contraseña a un usuario incauto. Para ser inmune a los ataques de *phishing*, se debe concienciar al usuario, porque no basta con tomar medidas tecnológicas. Además de todo esto, es esencial no apuntar contraseñas en lugares visibles. Estos temas son necesarios en cualquier política de seguridad informática de cualquier organismo o empresa.

2.2. Procedimiento de autenticación

Cuando un usuario se ha dado de alta en el servicio, ya está en disposición de usarlo. Es entonces cuando, concretamente, ya puede autenticarse en el servicio. Por ejemplo, ya puede usar su perfil de red social, o bien puede acceder a hacer un trámite con la Administración.

El sistema deberá tener en cuenta que el usuario se ha autenticado y se encuentra en una **sesión activa**. Los distintos sistemas operativos tienen implementado el control de los usuarios que están activos, pero esto no ocurre en las aplicaciones basadas en la web.

Trasladar directamente a las páginas web el concepto de sesión no es posible, puesto que el protocolo HTTP[8] trata las peticiones que hace el usuario de forma independiente. Para poder implementar el concepto de sesión en la tecnología de páginas web es necesario usar una *cookie* de sesión.

Una *cookie* es un fichero de texto que el servidor web guarda en la máquina cliente. Son los únicos ficheros que, por defecto, se pueden depositar en el equipo cliente, sin que su usuario sea consciente.

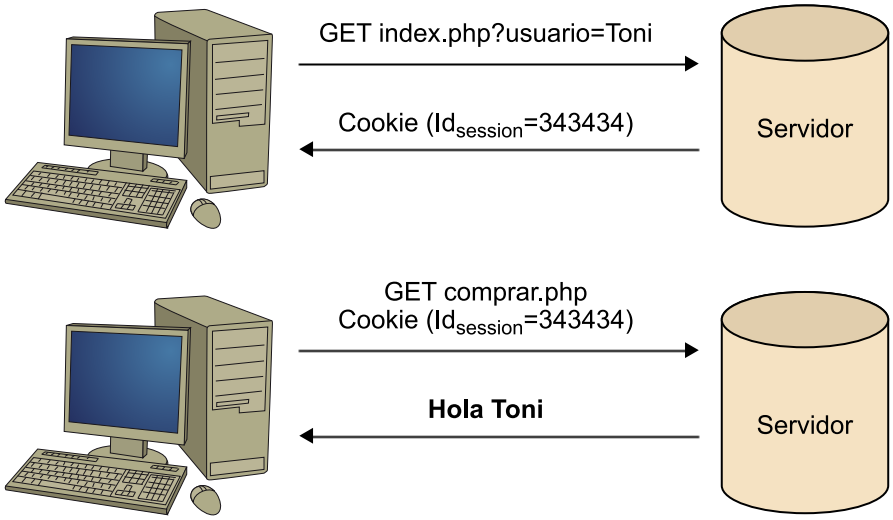
[8]HTTP es la sigla de *hypertext transfer protocol*.

Sesiones sin *cookies*

Pueden implementarse sesiones sin usar *cookies*: en este caso el identificador de sesión es un parámetro más de la dirección web.

Las *cookies* se usan para guardar información variada sobre el cliente, por ejemplo, se pueden guardar ciertas preferencias de aspecto de un sitio web que tiene el usuario. A través de las ***cookies* de sesión**, se pueden manejar sesiones de usuarios autenticados en páginas web: cuando el usuario se identifica correctamente, se le asigna un número de sesión (una cadena de bits larga y aleatoria). Este número de sesión suele guardarse en una *cookie* en el ordenador cliente. Así pues, el servidor es capaz de asociar al cliente con una determinada sesión.

Recordatorio de usuario mediante un identificador de sesión guardado en el ordenador cliente



Dentro de una sesión hay datos cuyo valor debe estar disponible durante toda la sesión: son las llamadas variables de sesión. Un lenguaje como PHP dispone de herramientas para empezar una sesión (`session_start()`), terminar una sesión (`session_destroy()`), así como herramientas para controlar las variables de sesión.

2.2.1. Autenticación mediante contraseña

Al introducir el concepto de las contraseñas hemos supuesto que el servidor guarda una lista de contraseñas en claro, es decir, sin codificar. Está claro que estas contraseñas deben guardarse en un fichero inaccesible para los usuarios no administradores. Y a su vez, el administrador, en cuya responsabilidad recae parte de la seguridad en este caso, debe ser muy cauto para que no haya problemas de robo de contraseñas.

Las contraseñas deben estar guardadas en un sitio seguro dentro del sistema de información (base de datos) de quien identifica. Así pues, una opción muy extendida es que, junto con la información del usuario no se guarda la contraseña sino un resumen de ésta. Por lo que, cuando un usuario introduce una contraseña, no se comprueba que la contraseña coincida con la almacenada, sino que se comprueba que el resumen de la contraseña coincide con el resumen almacenado de la contraseña. Para realizar el resumen se suele usar una función criptográfica unidireccional de resumen. En algunos sistemas, el valor

de contraseña que se guarda no es un resumen, sino el resultado de cifrar un vector de bits fijo y conocido por el sistema, usando la contraseña como clave de cifrado.

Función criptográfica de resumen

Una función criptográfica de resumen (*hash*) devuelve, a partir de una entrada de longitud arbitraria, una secuencia de bits de una longitud definida. Para que sea útil en seguridad computacional, a partir de los bits obtenidos no se debe poder deducir el valor de entrada (es unidireccional). Además, un pequeño cambio en la entrada debería reflejarse como un valor de salida completamente distinto. Estas funciones también ponen de relieve que hallar dos entradas que sean diferentes y den lugar a una misma salida es muy complejo computacionalmente. Algunos ejemplos de funciones de resumen son SHA-1 o MD5.

Claramente, es probable que los sistemas que permiten recordar una contraseña guarden la contraseña sin resumir o sin cifrar.

Durante el procedimiento de validación mediante contraseña, el usuario manda una contraseña al servicio de autenticación. En los primeros sistemas, la autenticación mediante servicios como Telnet propiciaba el envío de la contraseña sin cifrar. Es decir, que cualquier atacante, usando una herramienta de escucha de red (*sniffer*), podía usurpar la contraseña del usuario. Para evitar esto, se usa la protección a nivel de red o a nivel de sesión, mediante técnicas criptográficas. Ahora bien, tal y como se ha apuntado anteriormente, en la actualidad el robo de contraseñas suele tener lugar mediante ataques de *phishing*.

Pero hay un último tipo de problema que se debe tener en cuenta a la hora de autenticarse mediante contraseña: un atacante podría tener instalado un programa en el equipo del cliente que registrara todas las pulsaciones de teclado para luego mandarlas al atacante y proceder a su análisis.

Un *keylogger* es un sistema, generalmente software, cuyo objetivo es registrar todo lo que teclea el usuario del sistema donde está instalado, con fines de usurpación de contraseñas.

En general se trata de software, aunque también los hay en forma de dispositivo hardware, los cuales se pueden identificar fácilmente mediante una inspección visual. Para defenderse de los *keyloggers*, va siendo habitual pedir al usuario que utilice el ratón en lugar del teclado para introducir una contraseña, tal como se ve en la figura siguiente. En este caso, los caracteres para escribir la contraseña irán cambiando de posición a cada acceso de autenticación, con lo cual es más complejo pensar en un *mouselogger*. En este caso el atacante necesita capturar la pantalla del usuario para obtener la información (*screenlogger*).

Entrada de PIN mediante el ratón para evitar posibles *keyloggers*

2.2.2. Autenticación mediante certificados electrónicos

Los protocolos y especificaciones SSL/TLS/WTLS[9] protegen la capa del transporte de la información entre clientes y servidores web. Su uso, además, está extendido al envío y recepción de correo electrónico y, en general, a todas las aplicaciones que precisen de seguridad. Básicamente proporcionan:

(9)SSL es la sigla de *secure socket layer*; TLS, de *transport layer security*; y WTLS, de *wireless transport layer security*.

- Confidencialidad entre cliente y servidor (por ejemplo, para enviar información como números de tarjeta de crédito o contraseñas).
- Autenticidad del servidor, usando un certificado electrónico de servidor como método de autenticación.
- Integridad de la información.

Además, estos sistemas permiten la identificación de un cliente mediante su certificado. Al tratarse de especificaciones y protocolos muy parecidos, se tiende a hablar de ellos como los "protocolos SSL/TLS".

Cuando nos conectamos a un sitio web seguro (identificado mediante HTTPS[10]) tiene lugar una comunicación entre cliente y servidor destinada a crear un canal de comunicación seguro.

(10)HTTPS es la sigla de *hypertext transfer protocol over secure socket layer*.

En este ejemplo se resumen los mensajes que se transmiten cuando un cliente se conecta a un servicio HTTPS autenticado mediante un certificado de servidor:

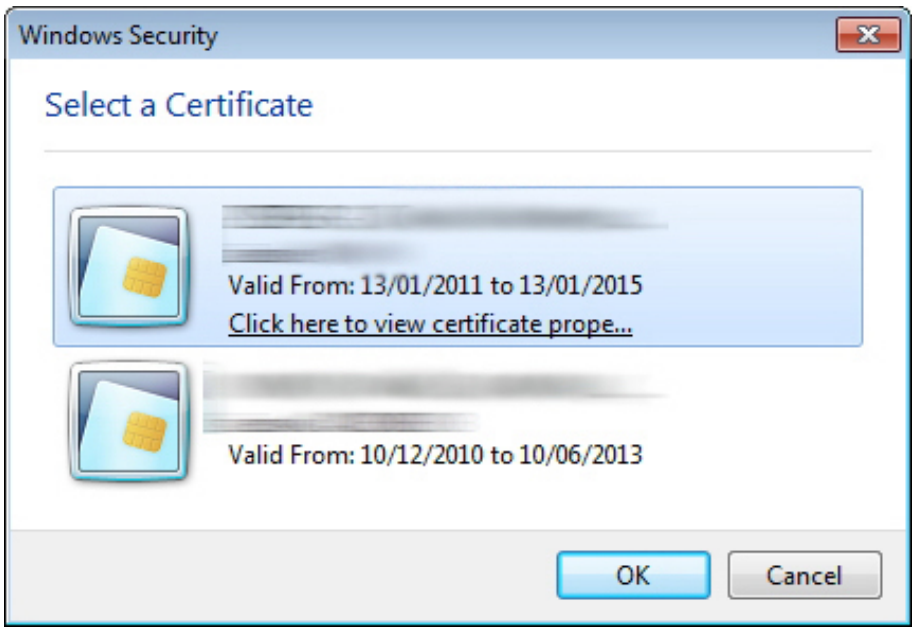
- *Hello request*. El servidor manda este mensaje al cliente (el navegador web) para iniciar la securización.
- *Client hello*. El cliente responde, dando una serie de información, como por ejemplo qué algoritmos criptográficos soporta (el servidor tendrá que acomodarse ante una variedad de software cliente con distintas versiones y sistemas operativos).
- *Server hello*. El servidor responde, y especifica qué combinación de algoritmos criptográficos se usará.
- *Certificate*. El servidor manda el certificado de servidor que incluye una referencia al emisor, con la cual el cliente podrá decidir si confía o no directamente en el certificado.
- *Server hello done*. A partir de este mensaje se intercambian las claves criptográficas y empieza la comunicación segura.

Un caso un tanto distinto es cuando el servidor requiere que el cliente se valide usando un **certificado de cliente** (el caso, por ejemplo, de la validación mediante el DNIe). En el caso de SSL, el servidor mandará, antes del *Server hello done*, el mensaje *Certificate request*. Este mensaje contiene una lista de los posibles tipos de certificado que el servidor puede admitir, por orden de preferencia. También manda una lista de autoridades de certificación. Así pues, el navegador carga el certificado correspondiente (en caso de haber varios certificados de cliente, el sistema pide elegir uno de ellos), tal como se ve en la figura siguiente.

Directiva *SSLVerifyClient require*

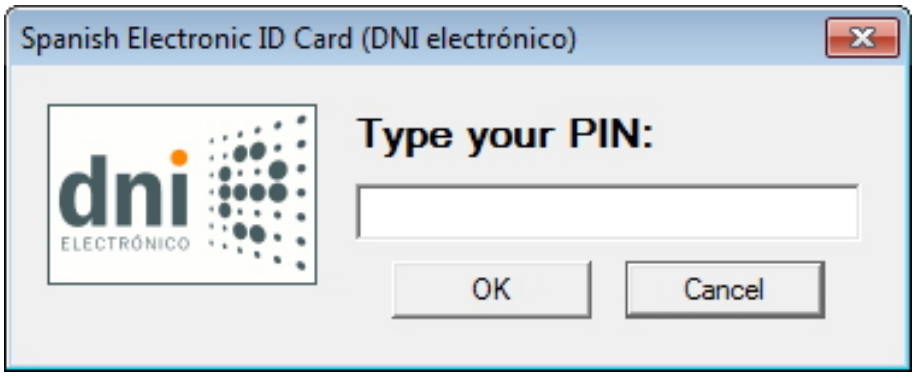
Se puede obligar el servidor Apache a la autenticación del cliente mediante certificado mediante la directiva *SSLVerifyClient require*.

Selección de certificado electrónico en Windows 7



Para poder usar el certificado, el usuario deberá demostrar que es el poseedor mediante la introducción del código o contraseña correspondiente.

Cuadro de diálogo para introducir la clave que permite usar el DNIe



Si la validación ha tenido éxito, se procederá a mandar el certificado mediante el mensaje *Certificate*.

2.2.3. Autenticación Single Sign On

Al principio de la llegada de Internet, los usuarios disponían de pocos servicios ante los que autenticarse. Hoy en día, son millares los sitios web que requieren de una autenticación del usuario para que éste los pueda usar. Desde la cuenta de correo, la red social, el servidor de vídeos, o el espacio de comentarios de la prensa, etc. Por otra parte, muchas empresas y organizaciones disponen de múltiples servicios propios o internos basados en la web.

Está claro que en ambos escenarios los usuarios deben recordar múltiples nombres y sus respectivas contraseñas para poder utilizar todos los servicios a los que están suscritos. Evidentemente, usar siempre la misma contraseña para todos los servicios plantea un problema: si un atacante obtiene la contraseña puede usarla de forma deshonesta en todos los servicios.

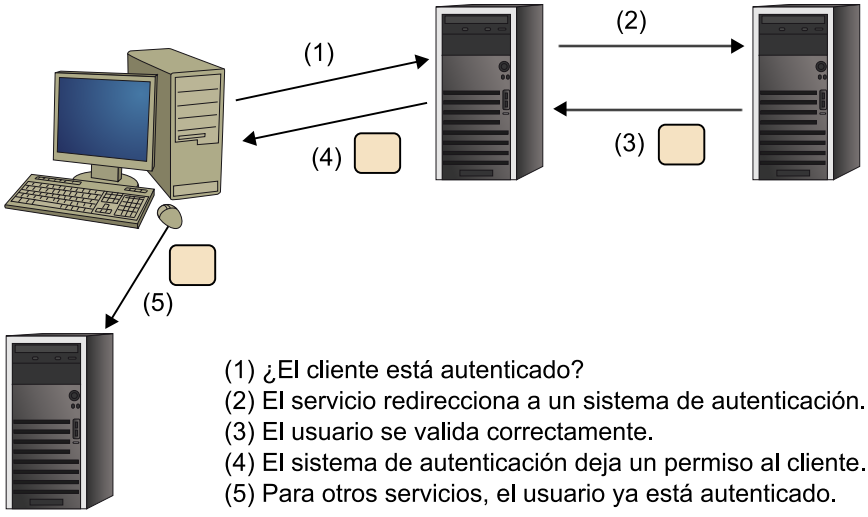
Para solucionar el hecho de que, dentro de una organización o un grupo de aplicaciones basados en la web, se tengan que usar múltiples nombres de usuario y contraseñas, existe el procedimiento *Single Sign On*.

Los sistemas *Single Sign On* permiten el acceso a varios servicios a partir de un único acto de autenticación inicial.

Ejemplo de sistema *Single Sign On*

Una compañía ofrece un correo electrónico, un disco virtual, un calendario y un sistema de elaboración cooperativa de documentos a través de la web. Cuando el usuario quiere usar, por ejemplo, el calendario, debe autenticarse ante el sistema. Con esto, al no estar autenticado, es redireccionado a un sistema autenticador. Si el usuario se valida correctamente, ya pude usar el calendario. Si el usuario ahora quiere trabajar con un documento en línea, no debe volver a autenticarse. Probablemente, el sistema de autenticación le habrá dejado en el equipo una *cookie* como prueba de que el usuario se ha validado. Esta *cookie* tendrá un periodo de validez a partir del cual la sesión habrá caducado y el usuario tendrá que autenticarse de nuevo. La figura siguiente muestra este procedimiento.

Procedimiento resumido de autenticación *Single Sign On*



(1) ¿El cliente está autenticado?
(2) El servicio redirecciona a un sistema de autenticación.
(3) El usuario se valida correctamente.
(4) El sistema de autenticación deja un permiso al cliente.
(5) Para otros servicios, el usuario ya está autenticado.

Problema de los sistemas *Single Sign On*

Obviamente, los sistemas *Single Sign On* también permiten a un atacante acceder a múltiples servicios conociendo una única contraseña del usuario.

2.3. Baja de usuarios

El ciclo de vida de una identidad digital termina con un procedimiento de baja. Todo proceso de gestión de identidades debe contemplar cómo se realiza este trámite. A continuación comentamos algunos aspectos referentes a la baja de contraseñas y certificados electrónicos.

2.3.1. Baja de contraseñas

Cuando un individuo ya no puede (o no quiere) usar un servicio telemático, debe (o puede) darse de baja. En los sistemas gestionados por administradores, este procedimiento debe hacerse cuando se notifique el cese del individuo. Claramente, un individuo que, por ejemplo, ya no trabaje en una empresa, no debería poder acceder a sus servicios telemáticos, como por ejemplo el correo electrónico o la intranet. Normalmente, puede definirse un breve periodo durante el cual el usuario todavía estará activo en todos o algunos de los servicios. Por ejemplo, no puede acceder a la intranet, pero sí seguir recibiendo correos electrónicos a la cuenta institucional (y quizás ya no pueda mandar correos desde esta cuenta).

En servicios gestionados automáticamente, en general servicios con una ingente cantidad de usuarios, el procedimiento de baja puede llegar a ser automático: cuando un usuario se registra, se le advierte de que, por ejemplo, su cuenta de usuario caducará si no ha realizado ninguna actividad por un determinado periodo de tiempo. No está de más que, en un sistema de este tipo, se manden algunos mensajes de recordatorio al usuario antes de que la cuenta caduque.

2.3.2. Baja de certificados electrónicos

En cuanto a certificados electrónicos, la baja de la identidad digital implica otros conceptos y escenarios:

- **Un certificado puede caducar**. Si la fecha en que se usa un certificado está fuera del periodo de validez, se avisará al propietario del certificado de un modo u otro. El administrador de un servicio debe prever la caducidad de los certificados de servidor de sus sistemas y renovarlos antes que caduquen para no perjudicar a los usuarios.

- **Un certificado ve comprometida su seguridad**. Por ejemplo, si se pierde un DNIe o bien el usuario cree que ha habido una usurpación de la clave privada o del código de acceso, se debe proceder con una **revocación** del certificado.

Un **certificado revocado** es un certificado considerado no válido aunque sea empleado dentro de su período de validez y su emisor sea de confianza.

Hay distintas técnicas para tratar con certificados revocados. En primer lugar, el usuario deberá pedir la revocación del certificado, mediante un trámite con el emisor del certificado.

Ahora bien, también es importante que la revocación sea notificada a los potenciales usuarios del certificado. Para resolver este problema existe el concepto de **lista de revocación** (CRL[11]). Una CRL contiene una lista de **números de serie** de certificados revocados por su autoridad de certificación. Si se desea comprobar la validez de un certificado, se debe descargar una CRL actualizada. La dirección web de donde descargar esta lista se indica en uno de los campos del certificado digital. La gestión de las CRL forma parte de las plataformas PKI. Aunque las CRL son un mecanismo de comprobación ampliamente usado, la obtención e interpretación de la lista de revocación acarrea cierta complejidad. El protocolo OCSP[12] es un protocolo que informa de la validez de un certificado en concreto: se realiza una petición a un servidor OCSP *responder* y este nos informará de la validez de un certificado. Así pues, se evita la descarga e interpretación (*parsing*) de una lista de certificados revocados.

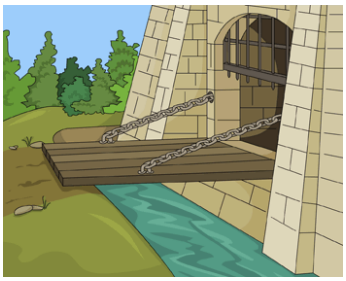
(11)CRL es la sigla de *certificate revocation list*.

(12)OCSP es la sigla de *online certificate status protocol*.

3. Control de acceso

Desde tiempos antiguos los humanos se han protegido a sí mismos y a sus recursos (alimentos, herramientas, etc.) mediante una gran variedad de sistemas. Probablemente, una de las formas más primitivas de proteger sus vidas y sus recursos fue el uso de cuevas. Estas les permitían guarecerse del frío y de los elementos, a la vez que dificultaban los ataques de depredadores. El tamaño de las cuevas y, en especial, el de sus entradas resultaba de vital importancia para los primeros humanos. Diversos estudios muestran que las cuevas con entradas pequeñas (más fáciles de controlar) eran usadas para realizar actividades secretas (Holderness y otros.).

El control de acceso ha sido históricamente un problema eminentemente físico. Murallas, fosos, puentes destruibles, puentes levadizos, puertas y verjas son solo algunos ejemplos de elementos arquitectónicos que a lo largo de la historia han sido usados para controlar el acceso a espacios y recursos. El control de acceso entendido desde un punto de vista físico se ha implementado en gran medida mediante el uso de llaves y puertas. Los recursos se almacenan en áreas solo accesibles a través de puertas que pueden ser abiertas únicamente con determinadas llaves. Así, solo los poseedores de las llaves adecuadas podrán acceder a los recursos.



Durante la Edad Media resultaba frecuente la protección de castillos y ciudades mediante el uso de puentes destruibles o levadizos que permitían cruzar un foso y acceder a un recinto a menudo amurallado.

Aunque el control de acceso físico es aún una realidad ineludible. ¿Quién no se ha olvidado las llaves alguna vez? Tras el advenimiento y la posterior generalización de las tecnologías de la información y de las comunicaciones, el control de acceso físico ha perdido protagonismo en favor del control de acceso lógico[13] o virtual, relacionado con los sistemas de seguridad informática. En el texto que sigue nos centraremos en el estudio del control de acceso entendido desde este punto de vista.

(13) *Lógico* en el sentido informático del término en contraposición a *físico*.

El **control de acceso** es el proceso por el cual, dada una petición de recursos, se permite o niega el acceso a los mismos en base a la aplicación de unas políticas de acceso.

Ved también

El concepto de políticas de acceso como conjunto de reglas que rigen el control de acceso se explica detalladamente en el subapartado 3.2.

El control de acceso comprende mecanismos de autenticación, autorización y auditoría. Sus principales objetivos son proteger datos y recursos frente al acceso no autorizado (proteger el secreto) y frente a una modificación no autorizada (proteger la integridad) a la vez que garantizar el acceso de los usuarios legítimos a los recursos (no denegación de servicio). Con el fin de conseguir estos objetivos, se controlan todos los accesos al sistema y sus recursos, y solo se permite que tengan lugar aquellos autorizados. Haciendo un símil con los sistemas físicos, en concreto los puentes levadizos y los puentes destruibles,

podemos observar que ambos puentes cumplen con los dos primeros objetivos (proteger el secreto y la integridad). Sin embargo, los puentes destruibles no satisfacen el tercer objetivo puesto que al destruirse no pueden ser cruzados por nadie, provocando así una denegación de servicio para las peticiones autorizadas.

Los sistemas de control de acceso deben ser entendidos como mecanismos de monitorización capaces de interceptar todas las peticiones de recursos que llegan al sistema. Estos sistemas de control deben cumplir con los siguientes requisitos:

- **Resistencia a manipulaciones**. El sistema no puede ser alterado o manipulado y, si lo es, dicha alteración debe ser detectable. En el caso de que no sea así, el sistema no sería seguro, ya que podría permitir el acceso no autorizado a recursos de forma inadvertida.

- **No eludible**. El sistema no puede ser saltado, es decir, todo acceso debe producirse a través de él. En caso de no cumplirse esta condición, el sistema no sería seguro dado que habría peticiones de recursos que no serían analizadas dando lugar, así, a posibles accesos no autorizados.

- **Seguridad nuclear**. La seguridad del sistema debe concentrarse en un núcleo y no distribuirse por el sistema informático. De no ser así, todo el código del sistema informático debería ser validado por todos los puntos de acceso creando así una sobrecarga innecesaria.

- **Tamaño pequeño**. El sistema debe ser lo suficientemente pequeño para permitir la prueba formal de su seguridad.

Demostraciones formales

Gracias a las demostraciones formales podemos probar las propiedades de seguridad de un modelo sin necesidad de implementarlo. Así, si el modelo es fiel a la realidad, el sistema resultante, tras su implementación, mantendrá las mismas propiedades de seguridad.

3.1. Fases del desarrollo de un sistema de control de acceso

El desarrollo de un sistema de control de acceso suele tener las tres fases siguientes:

1) la definición de las políticas de seguridad,

2) la representación mediante un modelo formal, y

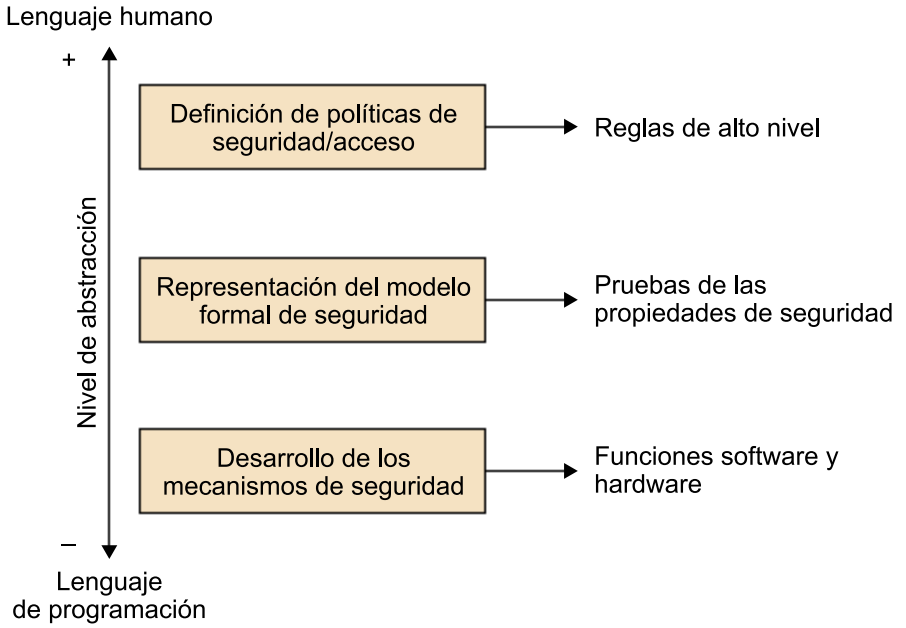
3) la implementación de los mecanismos de seguridad.

Políticas de seguridad

Las políticas de seguridad son el conjunto de reglas que regulan el acceso a los recursos del sistema.

Las tres fases, que se asemejan a las del proceso de desarrollo de software, se muestran gráficamente en la figura siguiente.

Abstracción y fases de desarrollo de un sistema de control de acceso



Durante la primera fase del proceso (**definición de políticas de seguridad**) se establecen mediante lenguaje natural el conjunto de reglas que regulan el acceso a los recursos del sistema de forma abstracta. En esta fase del proceso, las reglas pueden ser vagas e incluso ambiguas, ya que a menudo hacen referencia a leyes, procesos de funcionamiento propios, y procedimientos organizacionales. Esta ambigüedad hace que las reglas, en esta fase, deban ser interpretadas y no sean aptas para un sistema informático. Por ejemplo, podríamos considerar la regla "Todos los ingenieros de nivel avanzado tienen acceso al ordenador central", en este caso "nivel avanzado" es un concepto vago que posteriormente deberá ser formalizado.

En la segunda fase del proceso (**representación del modelo formal de seguridad**) se representa formalmente el conjunto de reglas y su funcionamiento. Esta formalización permite demostrar que un sistema cumple con un determinado conjunto de propiedades de seguridad. Uno de los primeros modelos de seguridad que se propusieron fue el de Bell-LaPadula (Bell; LaPadula, 1973). Posteriormente apareció el modelo HRU de Harrison, Ruzzo, y Ullmann en 1976 que formalizó, entre otros, el concepto de matriz de acceso (Harrison y otros, 1976).

Ved también

El concepto de concepto de matriz de acceso se explica detalladamente en el subapartado 3.3.

Finalmente, la tercera fase del proceso (**desarrollo de los mecanismos de seguridad**) consiste en la implementación del modelo mediante el uso de lenguajes de programación que son interpretados de forma determinista y sin ambigüedad por un sistema informático.

3.2. Políticas de acceso: concepto y elementos básicos

Todo sistema de control de acceso considera los siguientes elementos básicos:

- **Objetos** (también llamados objetivos). Son todas aquellas entidades de un sistema susceptibles de ser protegidas. En el caso de un sistema operativo pueden ser: archivos, directorios, programas, dispositivos, terminales, puertos, etc. En el caso de una base de datos, tenemos: tablas, relaciones, vistas, procedimientos, etc.

- **Acciones**. Todo aquello que se puede realizar sobre un objeto. Las acciones típicas que podemos realizar sobre un fichero son: lectura, escritura, creación, eliminación. En el caso de que el objeto sobre el que se realiza la acción sea un programa, cabría añadir la opción de ejecución.

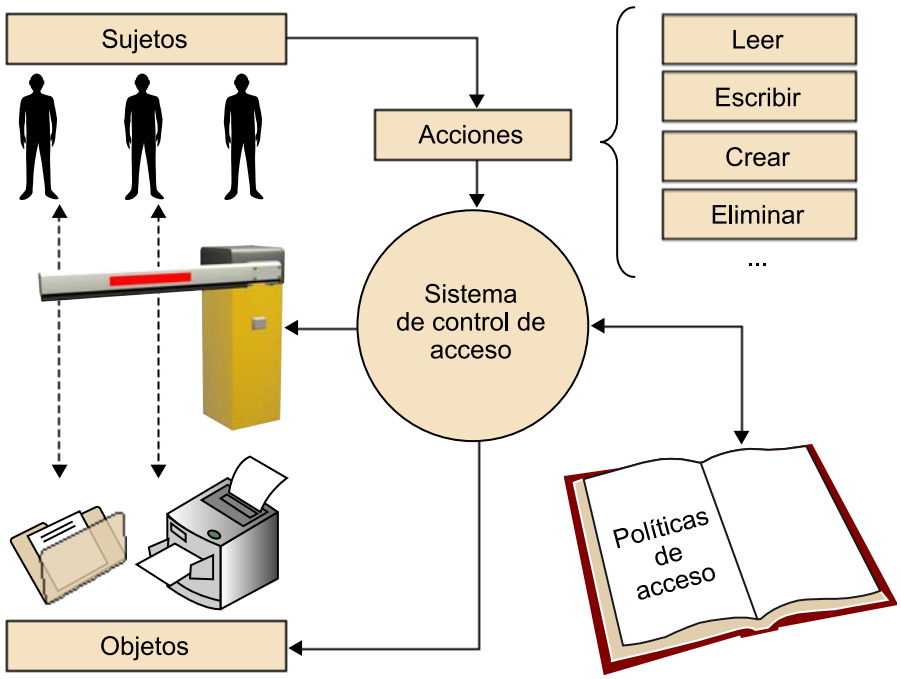
- **Sujetos** (también llamados iniciadores). Es cualquier entidad con capacidad para requerir el acceso a objetos del sistema. Los sujetos típicos de un sistema son sus usuarios y los procesos del sistema (por ejemplo, un navegador web, un procesador de textos, etc.).

En todo sistema informático, los sujetos realizan acciones sobre los objetos. El sistema de control de acceso es el encargado de decidir si un determinado sujeto tiene permiso para ejecutar una determinada acción sobre un determinado objeto. La decisión de permitir o denegar el acceso a los recursos se realiza en base a las políticas de acceso.

Las **políticas de acceso** son el conjunto de reglas que permiten determinar si un sujeto puede realizar una determinada acción (lectura, escritura, modificación, eliminación o ejecución) sobre un objeto.

La figura siguiente muestra un esquema de los componentes principales de un sistema y su interacción con el sistema de control de acceso. Obsérvese cómo el sistema de control de acceso (en el centro de la figura) recibe peticiones de los sujetos para realizar acciones. Evalúa estas peticiones mediante el uso de una política de acceso y actúa en consecuencia permitiendo o denegando el acceso a los objetos.

Elementos básicos de un sistema de control de acceso y su interacción



3.3. Tipos de control de acceso

En función de cómo se aplican y gestionan las políticas de acceso podemos distinguir tres tipos fundamentales de control de este:

1) Control de acceso obligatorio. Las políticas son evaluadas por el sistema entendido como un único ente central. Los sujetos del sistema no pueden rehacer/redefinir las políticas. Por ejemplo, no pueden dar permisos de acceso a otros usuarios. Los objetos y sujetos del sistema pertenecen a diversas clases de acceso. Así, para acceder a un objeto de una determinada clase hace falta que el sujeto pertenezca a una clase igual o superior (en términos de privilegios). Este tipo de acceso se inspira en el funcionamiento militar en el que la información (por ejemplo, planes de ataque) solo puede ser vista por aquellas personas que gozan de un nivel de seguridad suficiente (por ejemplo, nivel de coronel).

2) Control de acceso discrecional. Las políticas son gestionadas por los propietarios (sujetos) de los recursos (objetos). Los sujetos pueden modificar las políticas asociadas a los objetos del sistema. Por ejemplo, un propietario de un determinado objeto del sistema puede dar privilegios de acceso sobre ese objeto a otro sujeto. Este tipo de control de acceso es usado generalmente por los sistemas operativos.

3) Control de acceso basado en roles. Las políticas son definidas por el sistema pero, a diferencia de las políticas de acceso obligatorio, el acceso no se evalúa en función de permisos individuales sino mediante permisos de clase (o de rol). Cada rol tiene asignados ciertos privilegios y cada sujeto del sistema tiene asignado un rol, así el sujeto adquiere los privilegios del rol al que pertenece. Este tipo de control de acceso es un caso general de los dos anteriores. Suele usarse para la gestión de acceso a recursos de bases de datos.

Nota

El control de acceso obligatorio y el control de acceso discrecional pueden verse como casos particulares de un modelo de control de acceso basado en roles.

En los siguientes subapartados describiremos con más detalle las características de estos sistemas de control de acceso y algunos otros menos habituales.

3.3.1. Control de acceso obligatorio

En un sistema control de acceso obligatorio (MAC[14]), las políticas son evaluadas de forma centralizada por una autoridad (en el caso de un sistema operativo, el encargado estaría en el núcleo del mismo). Típicamente se usan sistemas de seguridad multinivel basados en clasificaciones de los sujetos y los objetos del sistema. Todos los sujetos y objetos tienen asignada una **clase de acceso**.

(14)MAC son las siglas en inglés de *mandatory access control*.

Una **clase de acceso** es un elemento de un conjunto de clases parcialmente ordenado. Donde el orden viene dado por una relación de dominancia representada por ≥.

En su forma más simple un conjunto de clases de acceso se podría definir mediante un conjunto de etiquetas ordenadas. Sin embargo, en general, las clases de acceso vienen caracterizadas por dos componentes:

- **Nivel de seguridad (*N*)**. Es un elemento de un conjunto jerárquico ordenado. Por ejemplo, podríamos considerar los elementos: seguridad alta (*SA*), seguridad media (*SM*) y seguridad baja (*SB*), donde $SA > SM > SB$.

- **Conjunto de categorías (*C*)**. Es un subconjunto de un conjunto no ordenado cuyos elementos hacen referencia a áreas funcionales o de competencia. Por ejemplo, podríamos considerar los elementos gestión, investigación, transferencia y docencia. En este caso el elemento Gestión podría asociarse a objetos relacionados con la administración, contabilidad y recursos humanos, mientras que el elemento Investigación o transferencia podría asociarse con objetos relacionados con laboratorios o equipos informáticos.

A partir de estas dos componentes (nivel de seguridad y conjunto de categorías) podemos definir la relación de dominancia como sigue.

Clases incomparables

Dadas dos clases c_1, c_2, si no se cumple que $c_1 \geq c_2$ ni que $c_2 \geq c_1$ se dice que las clases son **incomparables**.

La clase $c_1 \geq$ la classe c_2 si el nivel de seguridad de c_1 es mayor que el nivel de seguridad de c_2 y el conjunto de categorías de c_1 incluye al conjunto de categorías de c_2. Formalmente:

$$\forall\ c_1 = (N_1, C_1),\ c_2 = (N_2, C_2):\ c_1 \geq c_2 \Leftrightarrow N_1 \geq N_2 \wedge C_1 \supseteq C_2$$

La relación de dominancia definida anteriormente cumple con las propiedades: reflexiva, transitiva, antisimétrica, existencia de cota superior, y existencia de cota inferior.

Lectura recomendada

Si estáis interesados en profundizar sobre las propiedades de la relación de dominancia, podéis consultar la obra de Samarati y otros (2001).

La clase de acceso asociada a un objeto indica la **sensibilidad** de la información contenida en el objeto.

La **sensibilidad** de una determina información puede entenderse como el daño potencial que puede causar la revelación no autorizada de dicha información.

Por otro lado, la clase de acceso asociada a un sujeto indica su nivel de autorización formal (en inglés, *clearance*).

La **autorización formal** hace referencia al nivel de confianza en un sujeto. Puede verse como la confianza que se tiene en que el sujeto no revelará información sensible a sujetos sin autorización.

Así pues, el control de acceso obligatorio equivaldría a la restricción de acceso a objetos basada en la sensibilidad de la información contenida en los objetos y la autorización formal (*clearance*) de los sujetos para acceder a una información de ese nivel de sensibilidad.

3.3.2. Control de acceso discrecional

El **control de acceso discrecional** (DAC[15]) fue definido por el Trusted Computer System Evaluation Criteria en 1985. Se basa en evaluar qué sujeto realiza la petición de acceso a los recursos y en un conjunto de reglas de acceso explícitas definidas por los propietarios (*owners*) de los objetos. En un sistema DAC todo objeto tiene un propietario u *owner*, que es quien gestiona el acceso al mismo.

(15)DAC son las siglas en inglés de *discretionary access control*.

Recibe el nombre de discrecional debido a que los usuarios del sistema tienen la capacidad de transferir sus privilegios de acceso a otros usuarios. A diferencia del control de acceso obligatorio, el modelo DAC no usa un sistema cen-

tralizado en el que solo una única autoridad otorga y revoca privilegios de acceso a los recursos. En consecuencia resulta necesario el uso de políticas de administración.

En los sistemas DAC las **políticas de administración** regulan los procesos de gestión de privilegios (por ejemplo, transmisión, revocación...) entre los sujetos del sistema.

Modelo HRU

El modelo de matriz de acceso fue formalizado por Harrison, Ruzzo y Ullmann y se conoce como modelo HRU.

Una de las formas más comunes de definir los permisos que tiene cada sujeto sobre cada objeto es el modelo de matriz de acceso (propuesto en el modelo HRU). Este modelo, que fue inicialmente descrito por Lampson en el contexto de los sistemas operativos, y más tarde fue refinado por Graham y Denning, permite describir el control de acceso discrecional. Recibe el nombre de modelo de matriz de acceso porque usa una matriz para codificar el estado del acceso a los recursos en todo momento. Concretamente, cada celda $M(i,j)$ de la matriz, contiene las acciones que puede realizar el sujeto (i) sobre el objeto (j).

Ejemplo de matriz de acceso

La tabla siguiente muestra un ejemplo de matriz de acceso en la que pueden identificarse tres sujetos (Ana, Bernardo y Carlos), que podrían ser usuarios del sistema, y cuatro objetos (*Archivo1*, *Archivo2*, *Ejecutable1* y *Ejecutable2*). Si observamos la celda de la matriz correspondiente a la intersección entre el sujeto (Ana) y el objeto (*Archivo2*), podemos ver que las acciones permitidas son lectura y escritura. Además también podemos observar que Ana es la propietaria del *Archivo2*, lo cual le confiere la potestad de otorgar privilegios sobre este objeto a otros sujetos del sistema.

Ejemplo de matriz de acceso

	Archivo1	***Archivo2***	***Ejecutable1***	***Ejecutable2***
Ana	Lectura Escritura	Lectura Escritura Propiedad		Ejecución
Bernardo		Lectura	Ejecución	
Carlos				Ejecución

El estado de la matriz de acceso puede ser modificado mediante el uso de comandos que ejecutan funciones primitivas sobre el estado de las autorizaciones. Concretamente, el modelo HRU considera seis funciones primitivas:

- Adición y supresión de sujetos
- Adición y supresión de objetos
- Adición y supresión de privilegios

La representación de la matriz de acceso en su forma más simple, es decir, mediante una tabla bidimensional, resulta altamente ineficiente cuando el número de sujetos y objetos del sistema crece.

Obsérvese que si el número de sujetos y objetos es grande, es probable que la mayoría de las celdas de la matriz estén vacías (puesto que no todos los sujetos tienen acceso a todos los objetos). Por ello se usan formas de representación alternativas espacialmente más eficientes como:

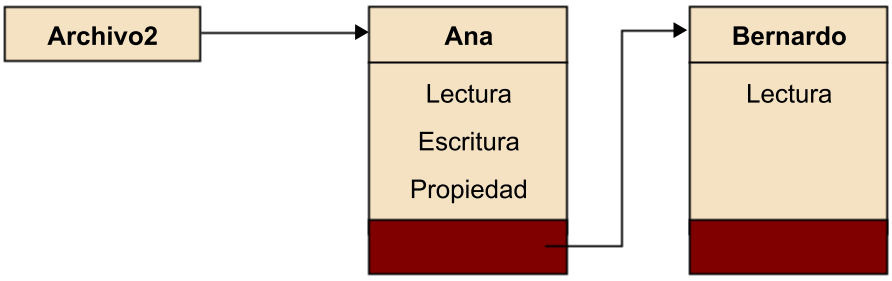
1) Tablas de autorización. Son tablas con tres columnas (sujeto, acción, objeto) que incluyen todas las celdas no vacías de la matriz de acceso. Así, cada tupla de la tabla representa la capacidad que tiene un sujeto de realizar una acción sobre un objeto. La tabla siguiente muestra la representación mediante una tabla de autorización de la matriz de acceso de la tabla anterior.

Representación mediante tabla de autorización

Sujeto	Acción	Objeto
Ana	Lectura	*Archivo1*
Ana	Escritura	*Archivo1*
Ana	Lectura	*Archivo2*
Ana	Escritura	*Archivo2*
Ana	Propietaria	*Archivo2*
Ana	Ejecución	*Ejecutable2*
Bernardo	Lectura	*Archivo2*
Bernardo	Ejecución	*Ejecutable1*
Carlos	Ejecución	*Ejecutable2*

2) Listas de control de acceso. Son listas enlazadas que representan la matriz de acceso por columnas. De este modo, dado un determinado objeto (por ejemplo, el *Archivo2*) se puede determinar si un sujeto (por ejemplo, Bernardo) tiene permiso para realizar alguna acción sobre el (por ejemplo, leerlo). El siguiente gráfico muestra la lista de control de acceso para el *Archivo2* según la matriz de acceso de ejemplo de la primera tabla de nuestro ejemplo.

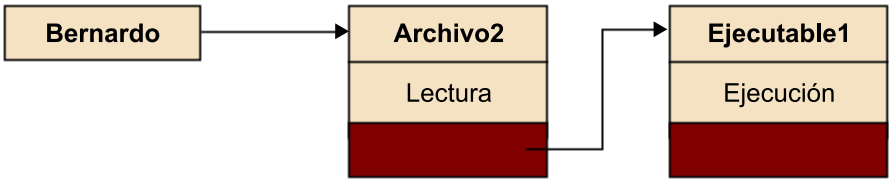
Ejemplo de listas de control de acceso



3) Listas de capacidades. Son listas enlazadas que representan la matriz de acceso por filas. De este modo, dado un determinado sujeto (por ejemplo, Bernardo) se puede determinar si puede realizar una acción (por ejemplo, Ejecu-

ción) sobre un determinado objeto (por ejemplo, el *Ejecutable1*). El siguiente gráfico muestra la lista de capacidades de **Bernardo** según la matriz de acceso de ejemplo de la primera tabla.

Lista de capacidades



3.3.3. Control de acceso basado en roles

Cuando se trabaja con organizaciones grandes resulta poco práctico tener que definir los privilegios de acceso de forma individualizada.

Ejemplo

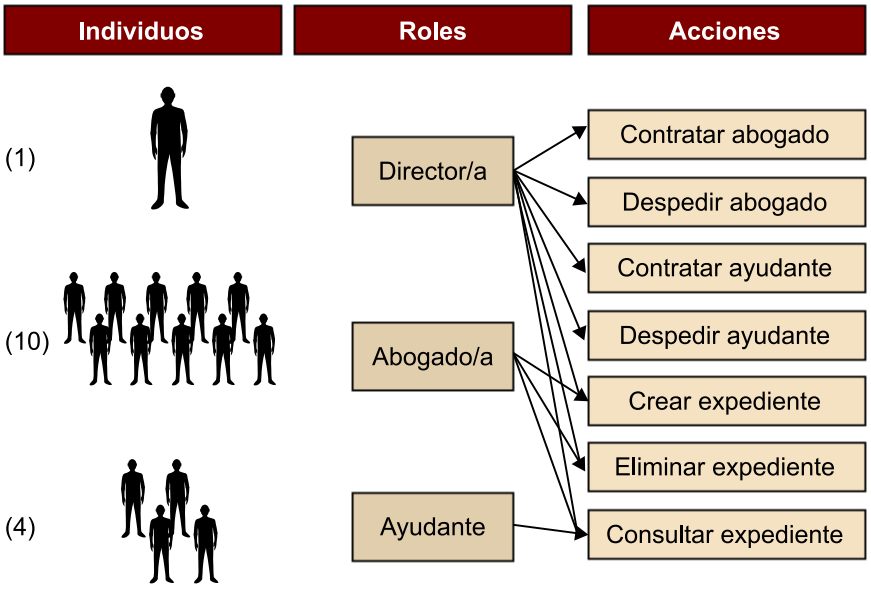
Tomemos como ejemplo un bufete de abogados formado por quince trabajadores de los cuales uno es el director, diez son abogados y cuatro son ayudantes. Supongamos que las acciones que se realizan en el bufete son:

- Contratar/Despedir abogados
- Contratar/Despedir ayudantes
- Crear/Eliminar expedientes
- Consultar expedientes

En el bufete el director puede realizar todas las acciones, los abogados pueden crear, eliminar y consultar expedientes y los ayudantes solo pueden consultar los expedientes.

La figura siguiente muestra gráficamente los usuarios (trabajadores) del bufete, sus roles, y las acciones que pueden realizar. Es fácil observar que la definición de las acciones que puede realizar cada trabajador de la empresa es más cómoda si se considera el rol que desempeña en la empresa (y por consiguiente, las tareas que realiza). Por ejemplo, no es necesario definir las acciones que puede realizar cada abogado de forma individual, sino que simplemente hace falta definir qué acciones puede hacer un abogado "genérico".

Usuarios, roles y acciones para el ejemplo del bufete de abogados



El **control de acceso basado en roles** (RBAC[16]) se fundamenta en la idea de asignar permisos/privilegios para realizar acciones a roles en vez de a sujetos del sistema. Así, cada sujeto del sistema tiene un rol asignado y puede realizar todas aquellas acciones para las que su rol tiene privilegios.

(16)RBAC son las siglas en inglés de *role-based access control*.

Un **rol** es la función que alguien o algo cumple. Por ejemplo, en los sistemas operativos suele hacerse una clara distinción entre el rol de administrador y el rol de usuario básico.

Obsérvese que la gestión de los permisos se simplifica claramente respecto a los anteriores modelos puesto que una vez asignados los permisos a los roles, la única tarea consiste en asignar correctamente los roles a los sujetos del sistema.

El modelo de control de acceso basado en roles es lo suficientemente flexible como para funcionar como el control de acceso obligatorio y el control de acceso discrecional. En realidad podemos considerar que tanto el modelo MAC como el DAC son casos particulares del modelo RBAC.

Resumen

En este módulo hemos estudiado el concepto de autenticación de la identidad y hemos revisado cuáles son las técnicas básicas para implementarla, el ciclo de vida de la identidad digital, y el control de acceso.

En primer lugar hemos visto el uso de contraseñas para la autenticación de los usuarios. Hemos estudiado el concepto de certificado electrónico, revisando de forma resumida el concepto de clave pública y las propiedades de seguridad de la información que permite alcanzar. El concepto de certificado electrónico ha servido para presentar el concepto de firma electrónica, de la cual hemos estudiado aspectos legales y de implementación con XML. También hemos realizado un repaso de los dispositivos de usuario para el soporte a los procesos de autenticación, viendo distintos tipos de *token*. Hemos presentado el concepto de tarjeta inteligente y detallado su uso en el DNI electrónico. Finalmente, hemos presentado la biometría como un conjunto de técnicas capaces de identificar a individuos o bien ayudar a su autenticación.

También se ha expuesto el tema del ciclo de vida de la identidad digital, estudiando conceptos como la gestión y creación de contraseñas, los problemas que esto conlleva y sus posibles soluciones. Hemos visto en detalle los procesos de autenticación mediante certificados electrónicos e introducido el concepto de *single sign on*, que se verá en más detalle en el módulo "*Single sign-on* y federación de identidades". Finalmente, hemos tratado la baja de usuarios, desde el punto de vista de usuarios autenticados con contraseña y desde el punto de vista de revocar certificados electrónicos.

A continuación hemos estudiado los sistemas de control de acceso. Tras presentar el concepto desde un punto de vista histórico y mostrar su vertiente más física, hemos descrito cómo la aparición de las TIC nos obliga a estudiar los sistemas de control de acceso desde un punto de vista de seguridad informática.

Finalmente, se ha analizado el concepto de política de acceso y descrito los principales actores que forman parte del sistema, es decir: objetos, sujetos y acciones. Hemos presentado los tres tipos fundamentales de control de acceso, a saber: control de acceso obligatorio (MAC), control de acceso discrecional (DAC) y control de acceso basado en roles (RBAC).

Actividades

1. Buscad por Internet algunos de los países que hayan integrado los certificados digitales en sus sistemas de identificación ciudadana. Elaborad un listado de qué servicios ofrecen en general y reúne datos sobre su aceptación.

2. Reunid información sobre cómo se procede a firmar electrónicamente algunos de los formatos de documento ofimático más comunes.

3. Buscad alguna aplicación biométrica de software libre o de demostración que podáis probar. Instaladla y haced varias pruebas para determinar su tasa de aciertos y fallos.

4. Visitad varias webs financieras, de comercio electrónico, etc. y enumerad los sistemas de autenticación que admiten. En caso de usar contraseñas, tened en cuenta si se usan algunas de las técnicas detalladas en el módulo.

5. Obtened un certificado de servidor de alguna web que conozcáis que use SSL. Observando los detalles (o propiedades, según el software que uséis) obtened la URL de la lista de revocación. Mediante el navegador, introducid esta URL. Analizad la información que contiene la lista de revocación.

6. Escoged un sistema operativo actual (Linux, Windows o MacOS) y determinad qué tipo de control de acceso utiliza.

7. A partir de la siguiente matriz referencia de acceso, identificad los objetos del sistema y los sujetos del sistema, y describid qué acciones puede realizar cada sujeto sobre cada objeto.

	Archivo1	***Archivo2***	***Ejecutable1***	***Ejecutable2***
Proceso1	Lectura Escritura Borrado	Lectura Escritura Borrado Propiedad		Ejecución
John	Lectura	Lectura		Ejecución
Julia	Lectura Escritura Borrado Propiedad		Ejecución	Ejecución

8. Representad la matriz de acceso anterior mediante la técnica de tabla de autorización y de lista de capacidades.

9. Mostrad la matriz de acceso para el ejemplo del bufete de abogados del subapartado 3.3.3. Argumentad por qué el uso de roles simplifica la gestión de los privilegios de acceso.

10. Además de los tres modelos de control de acceso descritos en este módulo (MAC, DAC y RBAC) existen otros modelos como el ABAC (*Attribute-based Access Control*). Buscad información sobre este modelo de control de acceso y proporcionad ejemplos de cómo usarlo mediante el lenguaje XACML.

Glosario

autoridad de certificación *f* Emisor de certificados electrónicos que goza de reconocimiento sobre su confianza.

autorización formal *f* Véase *clearance*

biometría *f* Aplicación de las matemáticas y la ciencia de la computación para identificar individuos de acuerdo con sus características o rasgos físicos.

captcha *f* Prueba basada en reconocer el texto en una imagen, cuyo objetivo es diferenciar si quien la resuelve es un humano o un programa informático.

clase de acceso *f* Elemento de un conjunto de clases parcialmente ordenado, donde el orden viene dado por una relación de dominancia representada por ≥.

clearance *m* Nivel de autorización o confianza en un sujeto. Puede verse como la confianza que se tiene en que el sujeto no revelará información sensible a sujetos sin autorización.

contraseña *f* Cadena de caracteres alfa8éricos de longitud arbitraria, usada como herramienta básica de autenticación de identidad.

control de acceso *m* Proceso por el cual, dada una petición de recursos, se permite o niega el acceso a los mismos en base a la aplicación de políticas de acceso.

firma electrónica reconocida *f* Firma electrónica creada por medios que el firmante puede mantener bajo su exclusivo control, basada en un certificado reconocido. Se equipara totalmente a la firma manuscrita.

HRU *m* Modelo de Harrison, Ruzzo y Ullman.

infraestructura de clave pública *f* En inglés *public key infraestructure*. Plataforma informática/telemática que permite la emisión y gestión de claves criptográficas y sus correspondientes certificados.

política de acceso *f* Conjunto de reglas que permiten determinar si un recurso puede ser visto, leído, modificado, eliminado o ejecutado por un sujeto del sistema.

política de administración *f* Conjunto de reglas que regulan los procesos de gestión de privilegios (transmisión, revocación, etc.) entre los sujetos de un sistema de control de acceso discrecional.

revocación *f* Proceso mediante el cual un certificado electrónico pierde su validez, a pesar de no estar caducado por fecha y estar emitido por una autoridad de confianza.

rol *m* Función que alguien o algo cumple.

sensibilidad *f* (de una determina información) Daño potencial que puede causar la revelación no autorizada de dicha información.

sesión *f* Periodo durante el cual un usuario está autorizado para realizar acciones en una aplicación basada en el web.

tarjeta inteligente *f* Tarjeta de plástico que lleva un chip incorporado, en general con capacidades de microprocesador con funciones criptográficas. El dispositivo es seguro contra manipulaciones.

token *m* Dispositivo cuyo objetivo es dar soporte al proceso de autenticación del usuario. Puede llevarse consigo.

XAdES *m* Sistema estándar de firmas electrónicas mediante tecnología XML. Reconocido como estándar en la Directiva Europea para la Firma Electrónica.

Bibliografía

Bell, D.E.; LaPadula, L. J. (1973). MITRE Documento Técnico 2547 (vol. 1), *Secure Computer Systems: Mathematical Foundations*. (Versión digital actualizada por Len LaPadula en 1996)

Departamento de Defensa de los EE. UU. (1985). *Trusted Computer System Evaluation Criteria*. DoD Standard 5200.28-STD.

Gollmann, D. (2005). *Computer Security* (2.ª ed.). Chichester (West Sussex): John Wiley & Sons.

Harrison, M. A.; Ruzzo, W. L.; Ullman, J. D. (1976). "Protection in Operating Systems". *Communications of the ACM* (vol. 8, núm. 19, págs. 461–471).

Herrera Joancomartí, J. (2006). *Aspectos avanzados de seguridad en redes*. Barcelona: Editorial UOC.

Holderness, H.; Davies, G.; Chamberlain, A.; Donahue, R. (2006). *A Conservation Audit of Archaeological Cave Resources in the Peak District and Yorkshire Dales*. Documento Técnico. CAPRA.

ISO (2009). *Common Criteria for Information Technology Security Evaluation* (*ISO-IEC-15408*).

Samarati, P.; De Capitani di Vimercati, S. (2001). "Access Control: Policies, Models, and Mechanisms". *FOSAD. Lecture Notes in Computer Science* (vol. 2171/2001, págs. 137-196).

Stallings, W. (2008). *Computer security: principles and practice*. Upper Saddle River (New Jersey): Pearson/Prentice Hall.

Windley, P. (2005). *Digital Identity*. Sebastopol (California): O'Reilly.