tecnimap 2002
www.tecnimap.com
VII JORNADAS SOBRE TECNOLOGÍAS DE LA INFORMACIÓN PARA LA MODERNIZACIÓN DE LAS ADMINISTRACIONES PÚBLICAS
A CORUÑA
15, 16, 17 y 18 de octubre de 2002
Palacio de Congresos y de la Ópera
Ponencias

Servicios de Criptografía e Infraestructura de Clave Pública (PKI) en el entorno "mainframe" IBM zSeries 900 2064/104 del INEM.

Enrique Sanz Velasco

Licenciado en Ciencias Físicas (Física del Estado Sólido)

por la Universidad Autónoma de Madrid U.A.M. (1981).

Pergentino Arias Prida

Licenciado en Ciencias Físicas (Física Aplicada)

por la Universidad Autónoma de Madrid U.A.M.

1.- Introducción.

La evolución de Internet en los últimos cinco años ha hecho que desde una primera fase de ofrecimiento de información estática consistente en páginas HTML, se haya pasado a una fase en la que sea necesario poner a disposición de los usuarios en general, tanto el acceso a los datos, almacenados en Bases de Datos Corporativas en sistemas de tipo "mainframe", como la posibilidad de efectuar algunos tipos de gestión, directamente por un usuario remoto conectado a Internet.

En el caso del INEM, la aplicación de pre-registro de contratos por Internet (Comunicación de la Contratación), permitirá al usuario, en este caso las empresas; la comunicación de la gestión de altas y bajas sobre el personal de su plantilla.

Un aspecto importante en aplicaciones de este tipo, que usan canales de comunicación no seguros, como es el caso de Internet, hace que los requisitos de seguridad hayan de ser tenidos en cuenta desde el principio. El mecanismo o mecanismos de seguridad que el INEM implante para este tipo de aplicaciones debe garantizar básicamente las siguientes características:

- Integridad de la información: su objetivo será impedir que la información transmitida a través de un canal de información inseguro, como puede ser Internet, no sea modificada durante la transmisión por parte de personas no autorizadas.
- Confidencialidad de la información: asegurando que los datos transmitidos, no pueden ser vistos por personas ajenas a la comunicación.
- Autenticación de la información: garantizando que los participantes en la comunicación son realmente quienes dicen ser.
- No repudio (irrenunciabilidad): proporcionando la prueba ante una tercera parte, de que cada una de las entidades han participado efectivamente en la transmisión.

Conceptos generales de seguridad criptográfica:

Cifrado: Es el proceso por el cual la información contenida en un mensaje, se transforma quedando oculto su significado. Modernamente los algoritmos que se emplean para cifrar mensajes son públicamente conocidos, aceptándose que la seguridad del cifrado ha de residir el uso de claves.

En la actualidad se consideran dos tipos de cifrado: de clave simétrica o única, y de clave asimétrica (llamado también de clave pública, dando lugar al término PKI: Public Key Infrastructure, como contraposición al anterior).

- El DES (Data Encryption Standard) es un cifrado de clave simétrica, la misma clave que se utiliza para cifrar se utiliza para descifrar. Este cifrado es el utilizado para codificar la información propiamente dicha.
- El RSA (Rivest-Shamir-Adleman) es un cifrado de clave asimétrica, es decir se utiliza una clave para cifrar y otra para descifrar. El conocimiento de una de las claves no conlleva el conocimiento de la otra. En 1976 Diffie y Hellman publican la idea de que cada usuario tenga dos claves: una clave pública de todos conocida; y una clave privada sólo de su propiedad. Aunque Diffie y Hellman definieron los principios de la criptografía de clave asimétrica, fueron Ron Rivest, Adi Shamir y Leonard Adleman, investigadores del MIT, los primeros que en 1978 encontraron el algoritmo adecuado.

Un usuario A (emisor) genera un mensaje para otro usuario B (receptor), y aplica el algoritmo de cifrado usando la clave pública de B (receptor), que es de dominio general; en el destino el usuario B descifra el mensaje utilizando su clave privada, que solo él conoce. Un intruso que capturase el mensaje mientras está en transito, no podría descifrarlo, al no conocer la clave privada del receptor. Este último método es el que presenta más interés, ya que puede ser usado junto a otras técnicas para proporcionar servicios, no solo de confidencialidad sino también de integridad y de autenticación. Estos mecanismos y técnicas son las llamadas firmas digitales.

Su base matemática es el siguiente teorema:

Sea p, un número primo. Entonces $x^{[p-1]} = 1(\mathrm{mod}\ p)$

Para un número entero k, $x^{[k\&(p-1)]} = x^{(p-1)}.....x^{(p-1)} = 1(\mathrm{mod}\ p).....1(\mathrm{mod}\ p) = 1(\mathrm{mod}\ p)$

Tomando ahora p y q dos números primos, se tendrá:

$$x^{[q-1 I p-1]} = 1(\mathrm{mod}\ p)$$
$$x^{[p-1 I q-1]} = 1(\mathrm{mod}\ q)$$

Entonces $x^{[q-1][p-1]} = 1(\mathrm{mod}(p.q)) = 1(\mathrm{mod}\ n)$, con $n = p.q$, siendo p y q primos.

El algoritmo RSA está descrito en múltiples sitios y es teóricamente simple (aunque computacionalmente costoso):

- Se encuentran dos números primos grandes p y q, (de 100 cifras o más).
- Se define n = p.q, conocido como módulo
- Se define z = (p-1).(q-1).
- Se busca un número e, llamado exponente, que sea menor que el módulo n = p.q, y primo con respecto de z = (p-1).(q-1). Clave pública.
- Se determina un número d, que existe y es único, tal que (ed -1) es divisible entre z. Clave privada.

De esta manera la clave publica esta constituida por el par (n,e), y la clave privada por (n,d).

El cifrado y descifrado de los mensajes se lleva a cabo por exponenciación. Dado un mensaje en claro M, las operaciones que permiten cifrarlo mediante la clave pública, obteniendo C, y descifrar C mediante la clave privada son:

$$C = M^e \pmod{n}$$

$$M = C^d \pmod{n}$$

El algoritmo RSA se basa en el hecho de que factorizar números muy grandes es un problema de difícil resolución. Este hecho que fundamenta su robustez, es también su principal problema, por el coste computacional que supone. En la práctica se fija el exponente de la clave pública: e (el valor comúnmente usado es 1000116), variando el módulo.

El algoritmo RSA, fue precipitadamente publicado, ante el temor de que la Administración U.S.A. lo clasificara dentro de los productos relativos a la seguridad, obteniendo la patente después de la publicación (algo que solo es posible en EE.UU., en el resto del mundo no se puede patentar nada que haya sido previamente publicado), por lo tanto dicha patente (expirada en septiembre de 2000), fue solamente valida en U.S.A., siendo libre en el resto de los países.

Este último método es el que presenta más interés, ya que puede ser usado junto a otras técnicas para proporcionar servicios, no solo de confidencialidad sino también de integridad y de autenticación. Estos mecanismos y técnicas son las llamadas firmas digitales.

Firmas digitales: La firma manuscrita como medio para acreditar la identidad del firmante de un documento ha sido, y sigue siendo, ampliamente usada. Su equivalente en la actual sociedad de la información es lo que se conoce como firma digital.

Además de la capacidad de autentificar al signatario de un mensaje, la firma digital posee otra cualidad interesante, que consiste en mantener la integridad del mensaje firmado. Dado un mensaje, basta calcular su huella digital (resultado de aplicar una función hash o resumen, sobre la totalidad del mensaje o una parte del mismo), y cifrarla con la clave privada del remitente para obtener simultáneamente la seguridad de que el contenido no se manipula (integridad), y de que el firmante es quien dice ser (autenticación). El procedimiento que se utiliza es el siguiente:

1. El usuario A (emisor) aplica una función resumen al mensaje. El resultado es la huella digital del mensaje.
2. A continuación, el usuario A (emisor) cifra con su clave privada este resultado. El resultado es el mensaje firmado digitalmente.
3. El usuario A (emisor) envía conjuntamente el mensaje y la firma digital (la huella digital cifrada con su clave privada) al usuario B (receptor).
4. El usuario B (receptor), que conoce la función resumen utilizada y la clave pública del usuario A (emisor), realiza dos operaciones: descifra la firma digital, mediante la clave pública; y aplica la función resumen al mensaje en claro (tal vez haya tenido que descifrarlo antes, si es que viene cifrado). Si ambos resultados coinciden, el usuario B (receptor) puede tener la certeza de dos hechos: que el mensaje no ha sido modificado en su tránsito por la red; y que el mensaje ha sido remitido, efectivamente, por el usuario A (emisor).

Hay que hacer notar que los procesos de cifrado y firmado son totalmente independientes. Un mensaje puede venir de varias formas: solo cifrado -sin firma electrónica-; solo firmado electrónicamente -con su texto en claro-; y cifrado y firmado, en este caso las parejas de claves pública/privada, pueden ser las mismas o independientes para cada una de las funciones de cifra y firma.

La huella digital: Se llama habitualmente huella digital al resultado de aplicar una función resumen (o función hash) sobre un mensaje.

Una función resumen toma como entrada una cadena de bits de longitud variable (mensaje) y genera como salida una cadena de bits de longitud fija (resumen). Para que una función de este tipo sea útil ha de cumplir con los siguientes requisitos:

- La entrada puede tener cualquier valor, es decir acepta todo.
- La salida ha de ser de longitud fija y menor que la longitud de la cadena de entrada.
- Para cualquier entrada su resumen ha de ser fácil de calcular.
- Ha de ser una función que actúe en un único sentido, entendiéndose que dado el resultado de la función de resumen ha de ser, computacionalmente hablando, imposible o extremadamente difícil calcular el mensaje de entrada.
- Debe ser difícil encontrar colisiones (es decir mensajes de entrada distintos que generen idéntica salida).

En la etapa anterior a la existencia de la firma electrónica las funciones resumen se utilizaron para firmar los mensajes, el problema surgió de inmediato debido a la existencia de colisiones:

- La función resumen dependía fuertemente del algoritmo de funcionamiento y por lo tanto en su secreto. Una vez conocido el algoritmo, podría ser relativamente sencillo para un intruso poder modificar los datos adecuados para generar la misma huella digital.
- El mayor problema de las funciones hash o resumen reside en las colisiones, es decir que existan distintos mensajes de entrada que generen la misma función resumen de salida.

En la actualidad se usan las funciones resumen como parte del proceso de firma digital. Es la huella digital de un mensaje, cifrada junto con la clave privada del usuario, mediante un mecanismo de clave asimétrica el procedimiento que a la vez garantiza que los datos de un documento no han sido variados y que proceden, efectivamente, de su remitente.

Funciones resumen más comunes.

Las funciones más comunes usadas son: MD4, MD5, SHA y SHA-1. Las dos primeras son acrónimo de "Message Digest" y fueron diseñadas por Rivest y se usan en pequeñas aplicaciones de correo electrónico; las otras son parte del standard para generación de firmas digitales.

- MD4: Introducida en 1990, su objetivo es ser rápida. En 1995 se demostró que era posible hallar colisiones en menos de un minuto con un simple PC. Ya no es considerada como segura.
- MD5: Versión mejorada (aunque computacionalmente más lenta) de MD4. Aunque se ha demostrado que la parte de compresión que utiliza el algoritmo tiene colisiones, no se ha demostrado para el algoritmo entero. Pero "por lo que pudiera pasar" se recomienda que cualquier producto que utilice el algoritmo se actualice a otros como SHA.
- SHA y SHA-1: (Secure Hash Algorithm), desarrollado en 1993 for NIST (National Institute for Standards and Technology) junto con NSA (National Security Agence) de EE.UU., para ser integrado en la normativa de firma digital DSS. Una versión más actualizada (que corregía un defecto no publicado de SHA), es publicada al año siguiente. Su modo de trabajo es similar a MD5, pero genera una salida de 160 bits (mayor que los 128 de MD4/5), con lo que lo hace más inmune frente a la busca de colisiones por fuerza bruta.

Mecanismos híbridos: El alto coste de computación que se deriva de los sistemas basados en PKI, ha hecho que se consideren algunas soluciones híbridas, de las que señalaremos únicamente la técnica del ensobrado electrónico:

- Ensobrado: Es una técnica muy usada para reducir el coste de cálculo de RSA. Consiste en generar aleatoriamente una clave DES, que se cifra con la clave pública RSA del destinatario; los datos significativos se

cifran con dicha clave DES generada. En destino, se descifra primero la clave DES con la clave privada RSA del destinatario, -como vemos la claves DES tiene validez para cada transmisión-; y por último, se descifra con la clave DES los datos que se han encriptado.

2.- Arquitectura de criptografía en IBM zSeries.

Las funciones de criptografía, dentro de los entornos IBM, se desarrollan según la arquitectura IBM CCA (Cryptographic Common Architecture), que define el conjunto de funciones criptográficas, interfaces externas y reglas de manejo de claves, que permiten la utilización de los algoritmos de encriptación standard, tanto simétricos: DES; como asimétricos o de clave pública: PKI (PKA -Public Key Architecture, en las referencias de los manuales IBM).

2.1.- Conceptos de Master Key y de separación de claves.

Master Key: Se llaman Master Keys al conjunto de claves simétricas, residentes en un coprocesador criptográfico, protegidas en una zona de memoria interna alimentada por batería de alta duración (aproximadamente 10 años), que sirven para encriptar el resto de las claves usadas por cada aplicación (claves operacionales). Este mecanismo permite la protección de muchas claves mediante la protección física de una sola.

El concepto de Master Key es aplicable tanto a claves DES como PKI.

Separación de claves: Dentro de la variedad de aplicaciones, cada clave criptográfica, puede ser usada para un tipo exclusivo de trabajo. Con objeto de proporcionar Master Keys para cada tipo de función determinada se usa un Control Vector.

Control Vector CV: Control Vector CV, es una cadena prefijada definida para cada función del coprocesador criptográfico, operando mediante XOR entre cada clave maestra y cada componente del vector, lo que obtenemos son claves maestras para cada tipo de necesidad.

Master Keys en un sistema S/390 (o zSeries) - Cryptographic Coprocessor Facility CCF.

En la arquitectura OS/390 con Cryptographic Coprocessor Facility (CCF), existen 3 tipos de Master Keys:

- DES Master Key: Es la clave maestra de las claves simétricas.
- PKA Signature Master Key (PKA-SMK): Clave maestra para firma electrónica.
- PKA Key Management Master Key (PKA-KMMK): Clave maestra de las claves asimétricas.

El hecho de existir una separación diferenciada para las claves maestras relativas a firmado y cifrado asimétrico (PKI), proviene de que mientras que un documento para ser cifrado requiere la clave pública del destinatario, en un documento a ser firmado se usa la clave privada del remitente.

2.2.- Hardware.

Las funciones de criptografía dentro del mainframe IBM zSeries 900 2064/104 (INEM), se implementan mediante el uso de dos coprocesadores criptográficos, separados e independientes de los procesadores de uso general. De esta manera se proporciona una separación entre el consumo de CPU en funciones propias de criptografía (de muy alto coste computacional), y el uso de la CPU por parte de las aplicaciones de gestión del sistema, y de las aplicaciones corporativas. Dichos procesadores criptográficos forman parte de el equipamiento de serie del sistema zSeries 900, en los equipos CMOS (desde la serie G5).

Dentro de dichas funciones ofrecidas se encuentran implementadas en hardware las siguientes funciones: DES, Triple DES, DSS, RSA, generación de números pseudo-aleatorios, y algoritmos de hash.

Según el anterior esquema hay que destacar los siguientes elementos:

- Buffers FIFO (cola) de entrada/salida.
- Procesador con funciones SHA y DES.
- Memoria RAM de 4 MB, donde residen las claves maestras (Master Keys).
- Generador de números aleatorios.
- Reloj interno.
- Batería de alimentación de memoria interna (RAM).
- Exponenciador modular para RSA de 1024 bits.
- Sensores de manipulación (Tamper Sensors).

Como parte importante relativa a la seguridad, hay que destacar que es en la memoria RAM de los coprocesadores criptográficos donde residen las claves maestras (Master Keys), por lo tanto quedan expuestos a la posibilidad de manipulación; para evitarlo, existen los sensores de manipulación (Tamper Sensors), encargados de borrar la memoria RAM de los coprocesadores al detectar alguna de las siguientes circunstancias:

- Condiciones extremas: Implican el borrado de la RAM, y la perdida de las Master Keys.
  - Intento de extracción física de la tarjeta del coprocesador criptográfico.
  - Exposición a radiación (Rayos X).

- Bajada de tensión de la batería de alimentación de la RAM.
- Temperatura por debajo de -20ºC +/- 5ºC, o por encima de 95ºC +/- 5ºC.

- Condiciones de aviso: No implican borrado de las Master Keys, pero impiden el uso de los coprocesadores hasta que las condiciones de normalidad se hallan alcanzado:
  - Variaciones de tensión por debajo de los límites de operación (3,3 V, (2,9+/-0,1 V) o 12 V, (10,5 V +/-0,15 V).
  - Variaciones de Temperatura que lleven fuera de la ventana de operación (0ºC+/-2ºC - 75ºC+/-3ºC).

2.3.- Definición de los perfiles de imagen de cada partición LPAR.

La máquina está dividida en 3 particiones lógicas PR/SM, que constituyen entornos de trabajo diferenciados : EXPLOTACIÓN, DESARROLLO y TEST.

EXPLOTACIÓN: Desde esta partición se da servicio en tiempo real a todas las aplicaciones corporativas del INEM, en sus diversas actividades (Prestaciones, REASS, Contratos, etc.), así como a las aplicaciones de gestión interna (Estadística, Gestión del Presupuesto, etc.).

La carga de trabajo se distribuye entre: el horario de servicio en tiempo real, de 8:00 a 18:00 horas; y los trabajos batch de mantenimiento, que tradicionalmente se ejecutan fuera del horario de conexión.

DESARROLLO: En este entorno se realizan los trabajos de diseño, desarrollo, programación y pruebas de aplicaciones, poniéndolas a punto antes de propagarlas hacia la partición de EXPLOTACIÓN.

TEST: Partición en la que se instalan y prueban nuevas versiones de sistema operativo, mantenimiento de productos y software de base. Los recursos asignados a esta partición son mínimos.

El coprocesador criptográfico contiene 16 conjuntos físicos de registros de Master Keys, cada uno constituye un Dominio. Cada dominio se asocia con una partición lógica LPAR mediante su Índice de dominio ("Usage Domain Index"). El mismo número de índice ha de ser definido en los parámetros de arranque del ICSF ("Integrated Cryptographic Support Facility"). Mediante el uso de índices de dominio lo que proporcionamos es independencia y aislamiento total (si se desea) entre las posibles particiones existentes.

3.- Integrated Cryptographic Service Facility (ICSF).

El ICSF (Integrated Cryptographic Service Facility), es la tarea encargada de proporcionar las funciones criptográficas, esta liberado como parte del software de base de OS/390 desde la versión 2.6. ICSF corre como una Started Task (STC), y tiene su propio espacio de direcciones y su espacio de datos (para residencia de copias en memoria de sus ficheros de claves).

Las funciones de ICSF son:

- Introducción de claves, tanto en los coprocesadores criptográficos (hardware) como en los ficheros de claves (tanto simétricas: CKDS (Cryptographic Key Data Set); como asimétricas: PKDS (Public Key Data Set)).
- Creación y manejo de claves criptográficas para uso de las aplicaciones.
- Servir de interface con el usuario de los programas que solicitan servicios de criptografía. ICSF es el único interface que soporta API,s :

- Protección de Datos.
- Protección de otras claves.
- Verificación de integridad de los mensajes (cifrado, descifrado, hash, etc.).
- Distribución de claves DES.
- Generación y verificación de firmas digitales.

ICSF, interacciona por una parte con el hardware de los coprocesadores criptográficos, proporcionando seguridad para las claves, y con la seguridad de OS/390 , para proporcionar seguridad a nivel de API,s, mediante RACF.

3.1.- Fichero CKDS (Cryptographic Key Data Set).

Las claves protegidas con la clave maestra DES, se encuentran almacenadas en un fichero VSAM, llamado CKDS (Cryptographic Key Data Set). Este fichero es un VSAM KSDS con un registro de longitud fija de 252 Bytes. El CKDS contiene una entrada única para cada clave de aplicación de tipo simétrico, cada registro del fichero contiene el valor de la clave y otra información relacionada.

Hay que hacer notar en este punto que las claves almacenadas en el CKDS, no lo están en claro, sino cifradas mediante la clave maestra (Master Key), por lo que una modificación de clave maestra, implica la necesidad de volver a cifrar todas las claves de aplicación .

3.2.- Fichero PKDS (Public Key data Set). Infraestructura de PKI.

Al igual que las claves simétricas de aplicación, son cifradas por la Master Key, las claves privadas de aplicación, en un sistema asimétrico, son encriptadas por su correspondiente clave maestra: KMMK (Key Management Master Key), para cifrado y/o firmado; y SMK (Signature Master Key), sólo para firma.

En la criptografía de clave asimétrica (PKI), existen dos tipos de parejas de claves pública /privada:

- Claves RSA, la clave RSA privada es usada para generar firmas digitales y para descifrar mensajes, la clave RSA pública es usada para verificar firmas digitales y para cifrar mensajes.
- Claves DSS (Digital Signature Standard), cuya parte privada es usada para generar firmas y la pública para verificarlas.

Cryptographic Coprocessor Feature, no permite generar claves RSA por sí mismo, hay varios caminos alternativos:

- TKE Workstation (versión 2): Estación de trabajo remota para administración de claves de seguridad.
- Generación desde una Workstation con adaptador criptográfico 4755, usando posteriormente un servicio de import (aplicación de la clave maestra).
- Generar la clave RSA en claro, desde otra plataforma, usando cualquier programa de software adecuado. Es necesario construir la estructura del registro de clave (External Key Token), y usar después el servicio de importación.

En nuestro caso hemos usado el último método, mediante un programa de generación de claves externo en PC (Open SSL).

4.- Interfaces de programación.

Las API,s invocables desde ICSF, pueden ser llamadas desde programas de aplicación, escritos tanto en determinados lenguajes de alto nivel, como en ensamblador:

- C.
- COBOL.
- FORTRAN.
- PL/I.
- Assembler OS/390.

Cada llamada invocada tiene una serie de parámetros, todos ellos obligatorios y posicionales. Pueden ser de entrada, de salida y de entrada/salida.

Los registros de los ficheros, tanto CKDS (claves DES), como PKDS (claves PKI), tienen dos partes:

- Key Label: Es la clave de acceso al fichero, considerado como un VSAM KSDS.
- Key Token: Es el resto del registro, que contiene el valor de la clave DES o RSA, entre otras informaciones.

A continuación, enumeramos las acciones más comunes que se realizan en criptografía, y sus API,s correspondientes (otras muchas pueden ser encontradas en las referencias):

- Claves simétricas DES.
- Utilidades.
- Claves asimétricas PKI.
- Firmado digital.

4.1.- Claves simétricas DES.

Definición y almacenamiento		
	Generación de clave DES (números aleatorios)	CSNBRNG
	Creación del Key Token	CSNBCKI
	Creación del Key Label	CSNBKRC
	Escritura del registro	CSNBKRW
Cifrado de un texto		CSNBENC
Descifrado de un texto		CSNBDEC

4.2.- Utilidades.

Generación de números aleatorios	CSNBRNG
Cálculo de Hash (MD5, SHA-1, etc.)	CSNBOWH
Conversión de códigos (*)	CSNBXEA/CSNBXAE

(*) - Una de las aplicaciones propuestas para utilización de la criptografía en mainframe, es el cálculo de las validaciones de tablas, usando de una función hash SHA-1. El valor obtenido ha de ser comparado con el obtenido con la misma función SHA-1 en los equipos de las Comunidades Autónomas. Tenemos que tener en cuenta que los equipos UNIX de la Autonomías utilizan ASCII 8859-1 Latin-1, mientras que el mainframe usa EBCDIC. Por ello es necesario un paso previo de conversión antes de invocar a la función resumen.

4.3.- Claves asimétricas PKI.

Definición y almacenamiento		
	Generación de las claves	Programa externo (Open SSL) (*)
	Creación del Key Token (**)	CSNDPKB
	Creación del Key Label y escritura del registro	CSNDKRC
Encriptación de un texto		CSNDPKE
Desencriptación de un texto		CSNDPKD

(*) Cryptographic Coprocessor Feature, no permite generar claves RSA por sí mismo, hay varios caminos alternativos:

- TKE Workstation (versión 2): Estación de trabajo remota para administración de claves de seguridad.
- Generar la clave RSA en claro, desde otra plataforma, usando cualquier programa de software adecuado.

(**) Una vez obtenida la clave RSA, podemos tener dos casos:

- Nosotros somos los propietarios de la clave, por lo que tenemos en este caso tanto una parte de clave pública, como una parte de clave privada.
- La clave pertenece a otra entidad, por lo que sólo tendremos la parte pública en el Key Token.

4.4.- Firmado digital.

Generación de firma de un texto		
	Generación del Hash	CSNBOWH
	Firma del texto con nuestra clave privada	CSNDDSG
Verificación de la	1.- Obtención del hash del texto	CSNDDSV
firma de un texto	2.- Obtención del hash a partir de	
	la firma (desencriptándola con la clave pública)	
	3.- Comparación	

5.- Conclusiones:

Si bien no es raro encontrar, desde el último año, los servicios de de criptografía y de infraestructura de clave pública (PKI) implantados en algunas aplicaciones de red que usan INTERNET, la potencia de criptografía de los ordenadores de tipo "mainframe" está poco explotada en el ámbito de la Administración Pública; no así en otros entornos de aplicaciones que requieren de ella (Banca Privada, Seguros, etc.).

La incorporación de estos servicios en hardware, usando en el caso del INEM dos co-procesadores criptográficos independientes, hace particularmente interesante su uso por parte de las aplicaciones; ya que los costosos cálculos derivados de la computación con claves públicas/privadas (PKI), son asumidas sin interferir en el consumo de CPU debido a las tareas propias de aplicación, con la evidente ventaja que ello supone.

Los métodos utilizados son standard: cifrado/descifrado usando claves DES (simétricas) o RSA de infraestructura de clave pública (PKI); generación de números aleatorios, translación de código (ASCII/EBCDIC), funciones hash usando los métodos standard: MD5, SHA-1, etc.; firmado y verificación de texto firmados, etc.

Como consecuencia de lo anterior, podemos estar en posición de proporcionar servicios basados en estos standards a las aplicaciones corporativas del INEM que así lo requieran, y tener la seguridad de que podemos establecer relaciones basadas en la arquitectura PKI con cualquier otro sistema (Comunidades Autónomas, Ayuntamientos, Empresas, etc.), que utilice los mismos standards.

Referencias:

- S/390 Crypto PCI Implementation Guide, SG24-5942-00, junio 2000.
- Exploiting S/390 Hardware Cryptography whit Trusted Key Entry, SG24-5455-00, noviembre 1999.
- OS/390 Integrated Cryptographic Service Facility - Administrator´s Guide SC23-3975-08, diciembre 2000.
- OS/390 Integrated Cryptographic Service Facility - System Programmer´s Guide SC23-3974-08, diciembre 2000.
- z/OS Version 1 Release 2 Implementation SG24-6235, mayo 2002.
- OS/390 Integrated Cryptographic Service Facility - Application Programmer´s Guide SC23-3976-08, diciembre 2000.