



Eusko Jaurlaritzaren Informatika Elkarte
Sociedad Informática del Gobierno Vasco

KeyTool IUI

Manual de usuario

Fecha:

Referencia:

EJIE S.A.
Mediterráneo, 3
Tel. 945 01 73 00*
Fax. 945 01 73 01
01010 Vitoria-Gasteiz
Posta-kutxatila / Apartado: 809
01080 Vitoria-Gasteiz
www.ejie.es

Este documento es propiedad de EJIE, S.A. y su contenido es confidencial. Este documento no puede ser reproducido, en su totalidad o parcialmente, ni mostrado a otros, ni utilizado para otros propósitos que los que han originado su entrega, sin el previo permiso escrito de EJIE, S.A.. En el caso de ser entregado en virtud de un contrato, su utilización estará limitada a lo expresamente autorizado en dicho contrato. EJIE, S.A. no podrá ser considerada responsable de eventuales errores u omisiones en la edición del documento.

Control de documentación

Título de documento: KeyTool IUI. Manual de Usuario

Histórico de versiones

Código:

Versión: 1.0

Fecha:

Resumen de cambios:

Cambios producidos desde la última versión

Primera versión.

Control de difusión

Responsable: Ander Martínez

Aprobado por: Ander Martínez

Firma:

Fecha:

Distribución:

Referencias de archivo

Autor: Consultoría de áreas de conocimiento

Nombre archivo: KeyTool IUI. Manual de instalación v1.0.doc

Localización:

Contenido

Capítulo/sección	Página
1 Introducción	5
2 Conceptos básicos	5
3 Integración con la implementación Java	5
3.1 Conceptos básicos de Criptografía y Java	6
4 Visualización de de certificados y claves	7
4.1 Formatos admitidos para certificados y claves	7
4.2 Visualización de certificados y claves en un almacén	8
4.2.1. Visualización del detalle de un certificado o clave	9
4.3 Visualización del almacén por defecto de la implementación Java	10
4.4 Visualización de certificados almacenados en fichero	11
5 Gestión de los almacenes de certificados y claves	13
5.1 Gestión de los certificados de un almacén	14
5.2 Gestión de las claves (privadas o secretas) de un almacén	15
5.3 Gestión del almacén por defecto de la implementación Java	16
5.4 Cambio de la password de acceso al almacén.	16
6 Funcionalidades	17
6.1 Creación de un almacén de certificados	17
6.2 Creación de una clave secreta y asignación a un almacén	19
6.3 Creación de una clave privada y asignación a un almacén	21
6.4 Creación del jar de un directorio	24
6.5 Firma digital de documentos	25
6.6 Validación de la firma digital de un documento	26
6.7 Cifrado de documentos utilizando la entrada de un almacén	28

6.8	Descifrado de documentos utilizando una clave de un almacén	30
6.9	Importación de claves y certificados a un almacén	30
6.9.1.	Importación al almacén de Autoridades Certificadoras Raíz: cacerts	32
6.9.2.	Importación desde el fichero de respuesta de una CA	32
6.10	Exportación de claves y certificados a fichero	33
6.10.1.	Exportación del certificado con la clave pública	35
7	Acceso a la ayuda de la aplicación	36

1 Introducción

En este manual se describe la funcionalidad de la aplicación KeyTool IUI versión 2.4.

2 Conceptos básicos

KeyTool IUI es una aplicación dotada de un interfaz gráfico de uso muy sencillo, que pretende ayudar en la realización de tareas criptográficas enmarcadas en el entorno de las aplicaciones Java.

El JDK de Sun, incluye algunas herramientas de manejo de claves y del almacén de certificados (keytool, jarsigner), pero su ejecución es desde línea de comandos, no se incluyen herramientas visuales que faciliten el trabajo con certificados. KeyTool IUI pretende dar al menos la misma funcionalidad que las herramientas de Sun pero con un interfaz gráfico muy sencillo. Además proporciona otra funcionalidad como la de cifrado de ficheros

Con KeyTool IUI se podrá crear un nuevo almacén de claves (KeyStore), importar en él certificados, firmar ficheros con dichos certificados, etc.

En el momento de escribir este documento la última versión disponible de la aplicación es la 2.4.

Puede encontrarse información adicional en inglés sobre el producto, en la página web:

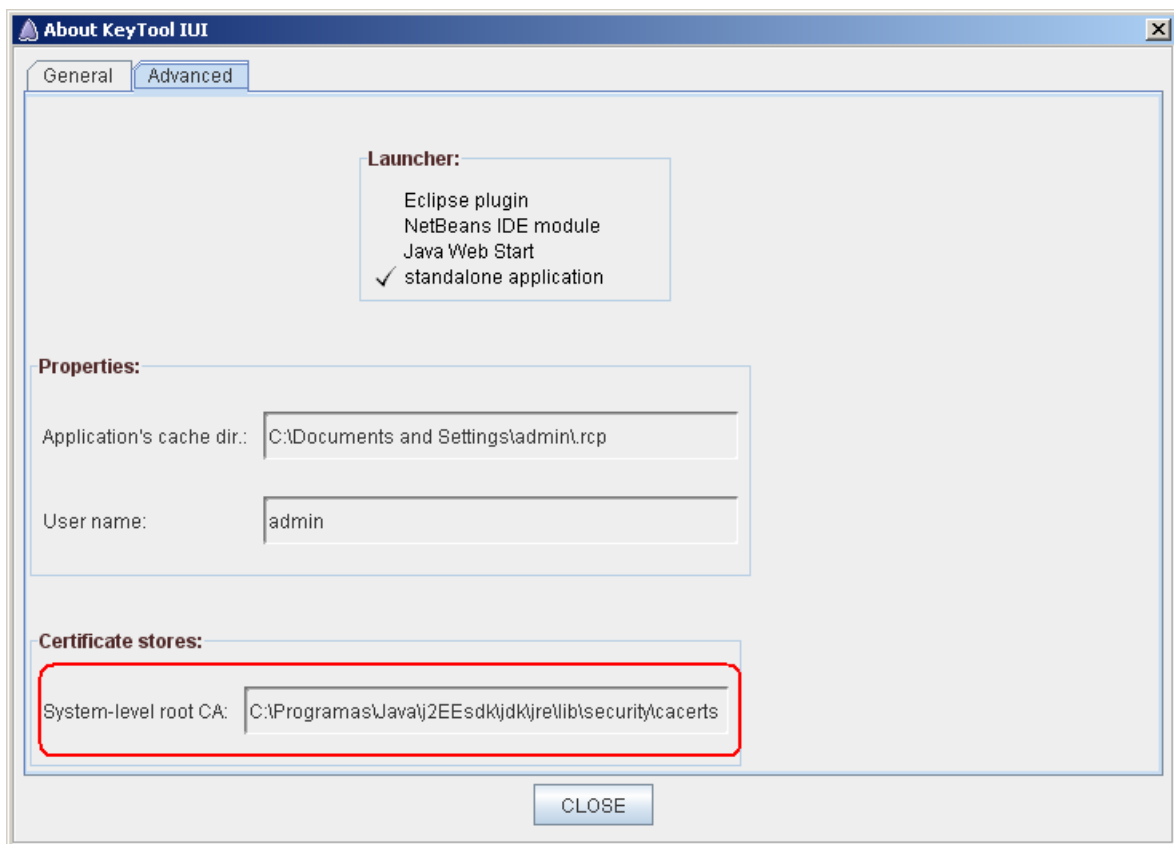
http://yellowcat1.free.fr/index_ktl.html

o en francés, en:

<http://ragingcat.developpez.com/java/outils/keytool/ui/>

3 Integración con la implementación Java

La aplicación KeyTool IUI es una herramienta desarrollada en Java y como tal utiliza por defecto el almacén de certificados que viene con la implementación Java con la que se arranca la aplicación. Puede verse accediendo al menú: "Help → About → KeyTool IUI...", en la pestaña 'Advanced' se indica el almacén de certificados que utiliza el sistema por defecto:



3.1 Conceptos básicos de Criptografía y Java

En Java existen dos APIs utilizadas fundamentalmente para temas criptográficos, son: JCE (Java Cryptography Extension) y JSSE (Java Security Socket Layer).

JCE es un API que da soporte para la realización de servicios criptográficos como: implementación de algoritmos de cifrado (RSA, DSA, AES, Triple DES, SHA, PKCS#5, etc.), cifrado de textos (simétrico, asimétrico, de bloque, etc...), firmas digitales, generación de claves y certificados, etc.

JSSE se utiliza fundamentalmente para comunicaciones seguras, dando soporte a protocolos como SSL, TLS, Kerberos (utilizando GSS-API). Ofrece una completa implementación de HTTPS sobre TLS y SSL.

Ambas APIs utilizan los almacenes de claves y certificados (Keystores), que son ficheros que almacenan de forma segura los certificados de las entidades certificadoras, los pares de clave pública y privada de un usuario y cualquier otra clave privada utilizada para cifrar un texto.

El almacén que utiliza Java por defecto es el que se encuentra en el directorio lib\security\ de la implementación JRE utilizada, y se llama cacerts, pero puede cambiarse, bien en el arranque de la aplicación, bien por programa.

Cuando en el presente manual se hace referencia a una clave privada, se está haciendo referencia al conjunto de clave privada – clave pública utilizado en el cifrado asimétrico. Cuando se hace referencia a una clave secreta, se está haciendo referencia a la utilizada en el cifrado simétrico.

4 Visualización de de certificados y claves

Con la aplicación KeyTool IUI se pueden ver todos los certificados de las autoridades certificadoras (CAs) y claves (pública y privada) que incluya un almacén.

4.1 Formatos admitidos para certificados y claves

Los formatos que admite la aplicación para almacén de claves y certificados son:

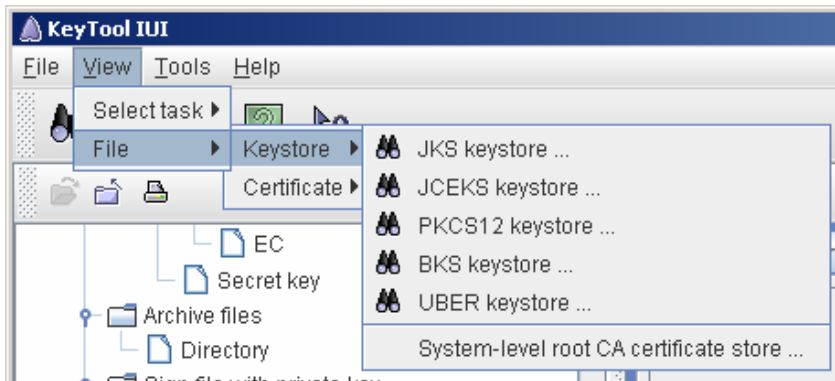
Formato	Nombre		Extensión del Fichero
JKS	Java Keystore	Tipo de formato ideado inicialmente para el Java de Sun	.jks
			.ks
JCEKS	Java Cryptography Extension Keystore	Definido en el API JCE	.jce
PKCS12	Public Key Cryptography Standards #12		.p12
			.pfx
BKS	Bouncycastle Keystore	Definido por el grupo BouncyCastle	.bks
UBER	Bouncycastle UBER Keystore	Versión más segura definida por el grupo BouncyCastle	.ubr

También pueden visualizarse los certificados guardados en ficheros. Los formatos de fichero aceptados por la aplicación son:

Formato	Nombre		Extensión del Fichero
DER	Codificación DER: Distinguished Encoding Rules	Definido en el estándar X.690 de ITU-T	.crt
			.cer
CMS	Cryptographic Message Syntax	Derivado de la sintaxis definida en PKCS#7	.p7c
PEM	Codificación PEM: Privacy-Enhanced Electronic Mail	También llamada: "printable PKCS#7", por generar un fichero ASCII.	.pem
PKCS10	Public Key Cryptography Standards #10	Estandar para la solicitud de certificación de una clave pública a una CA.	.p10 .csr
PKCS7	Public Key Cryptography Standards #7	Utilizado en las respuestas de las CAs a una solicitud de certificación (PKCS#10).	.p7b

4.2 Visualización de certificados y claves en un almacén

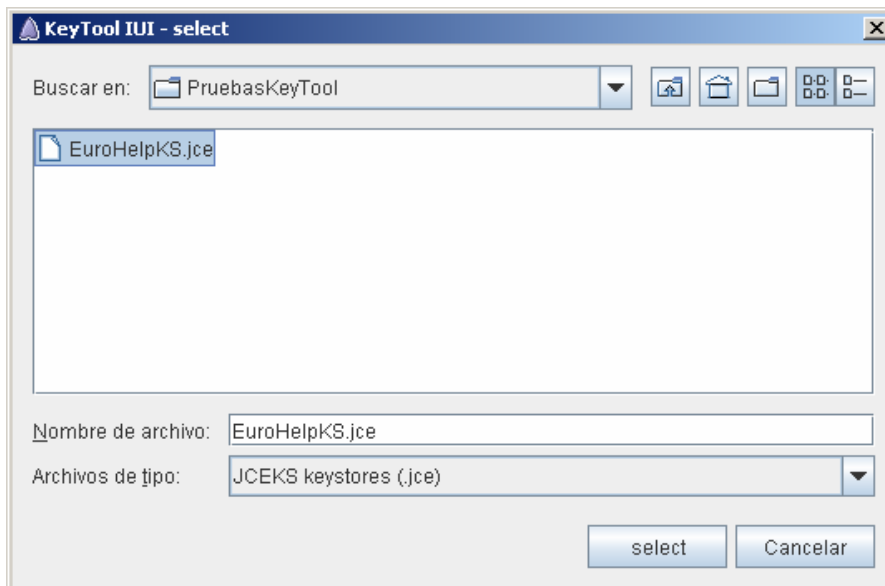
Para ver los certificados y claves en un almacén, acceder al menú: “View → File → Keystore” y seleccionar la opción deseada en función del formato del almacén.



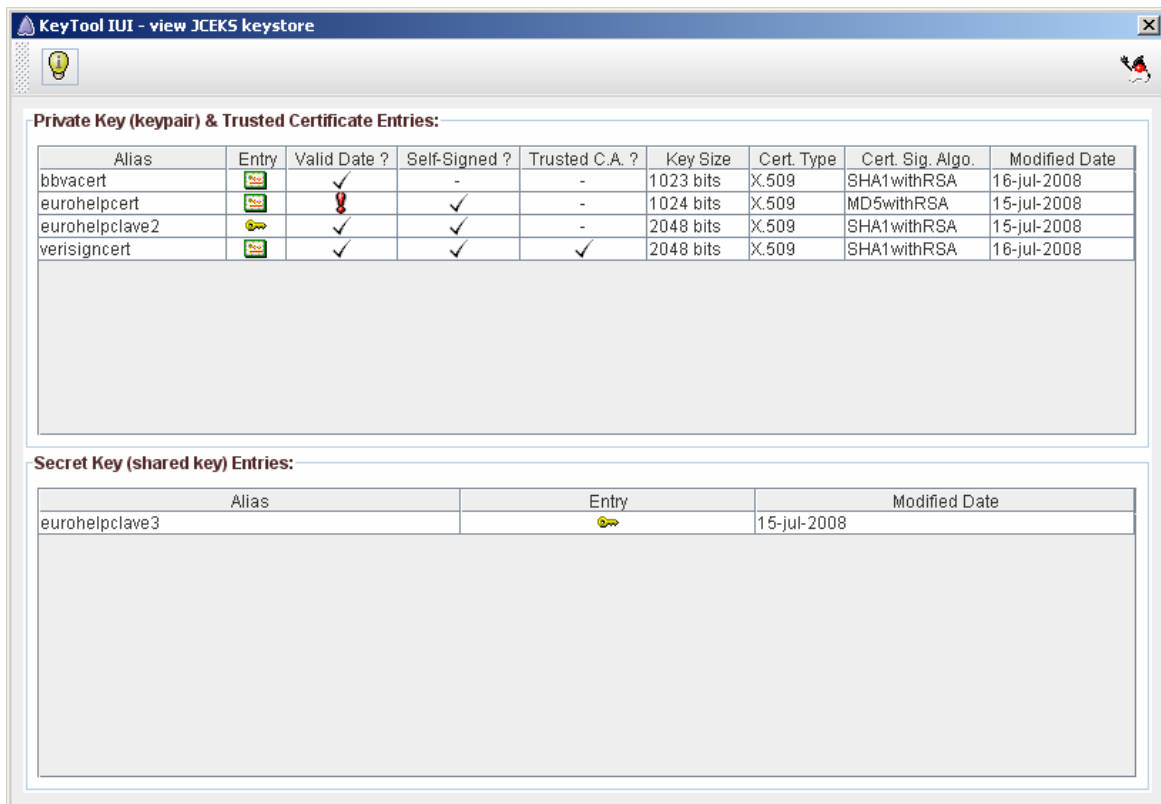
Se puede acceder también desde el botón de la barra de herramientas:



Se mostrará un cuadro de diálogo para seleccionar el almacén:

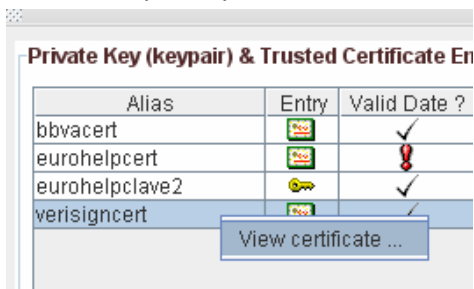


Aparecerá la siguiente ventana, con todos los certificados y claves que contiene el almacén.

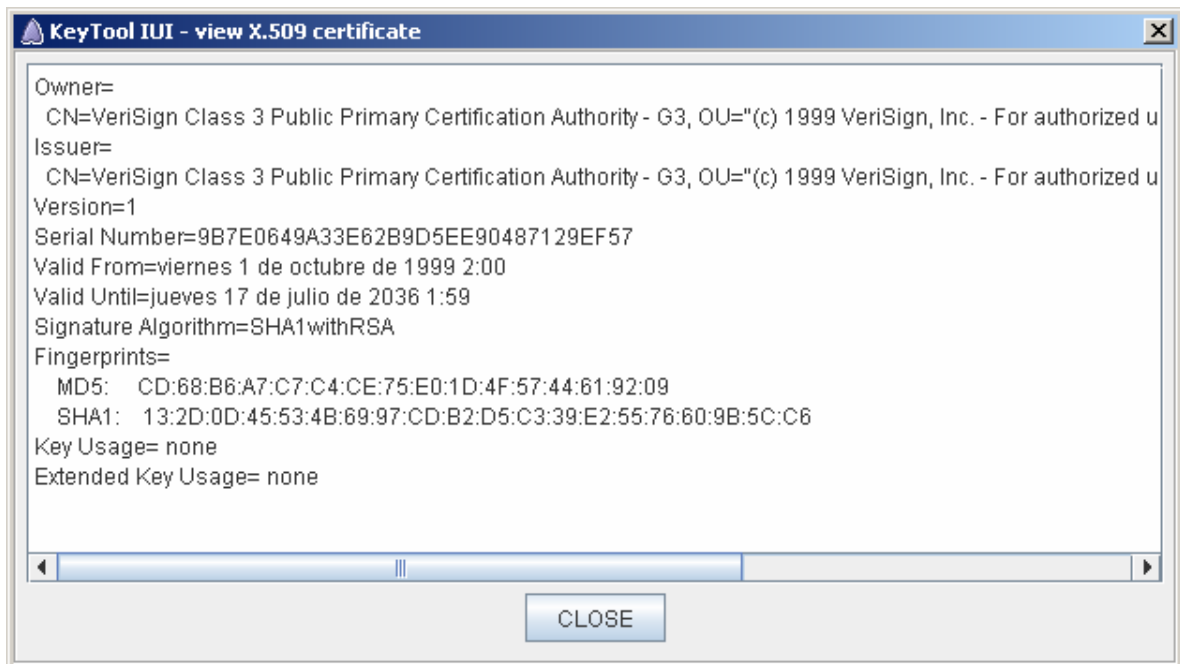


4.2.1. Visualización del detalle de un certificado o clave

Pinchando con el botón derecho del ratón sobre una entrada (certificado o clave), aparecerá un menú contextual con el que se puede obtener más información sobre dicha entrada.

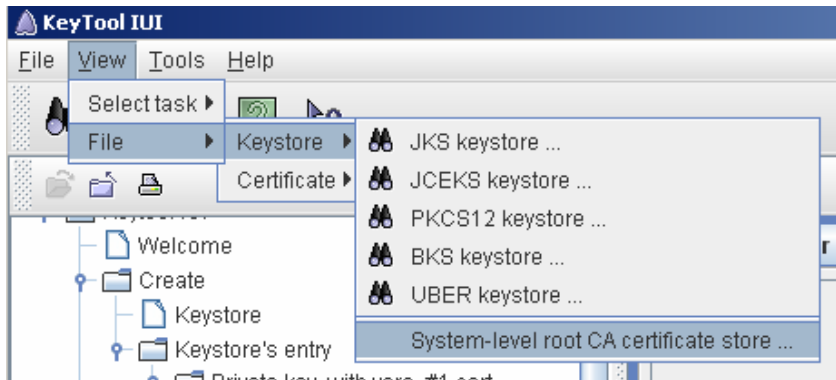


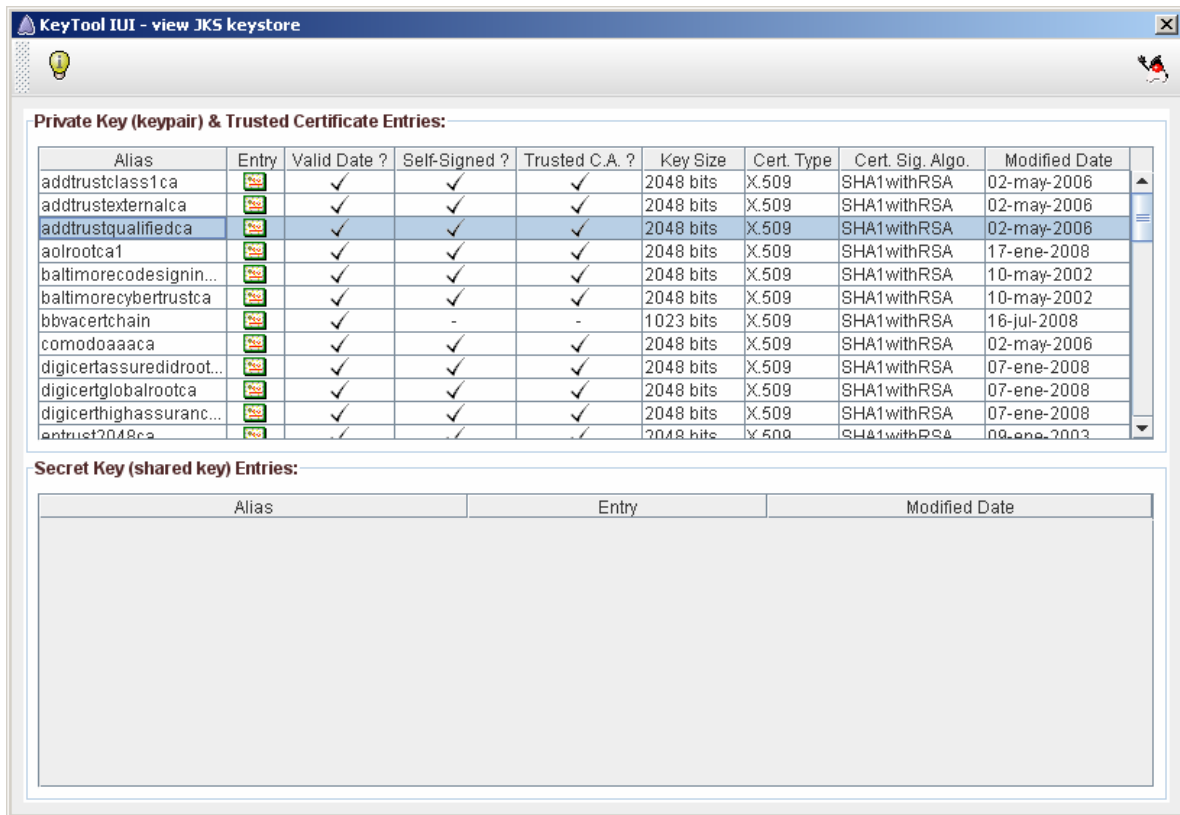
Se muestra en modo texto los datos del certificado o clave.



4.3 Visualización del almacén por defecto de la implementación Java

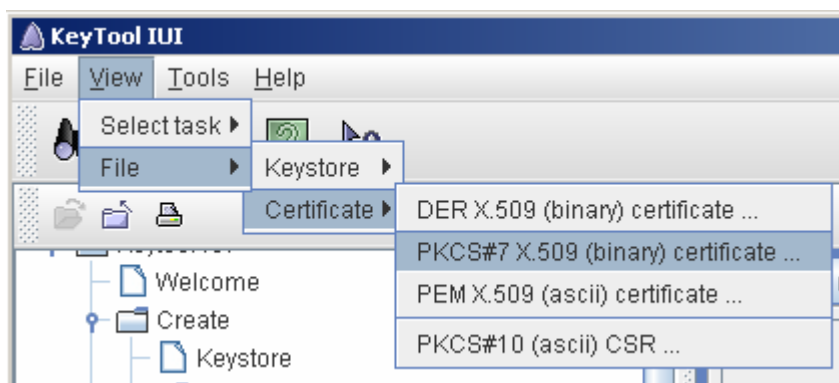
En el almacén de certificados que viene por defecto con el JRE de Java (cacerts), se incluyen varias CAs en las que se confía. Si se desea ver cuáles son, la aplicación KeyTool IUI facilita el proceso, ya que dispone de un menú de acceso directo a dicho almacén (ver la siguiente imagen).



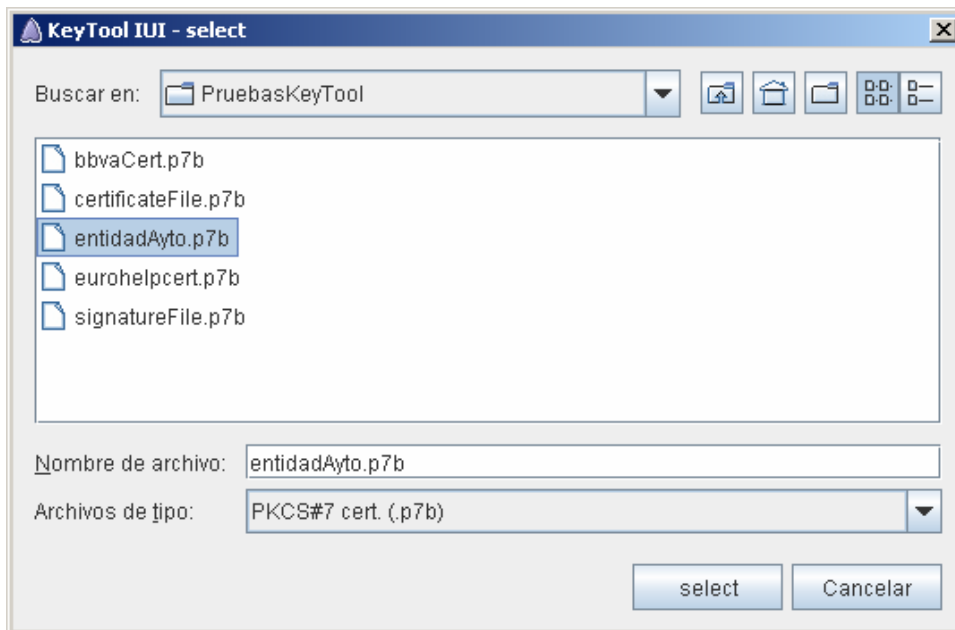


4.4 Visualización de certificados almacenados en fichero

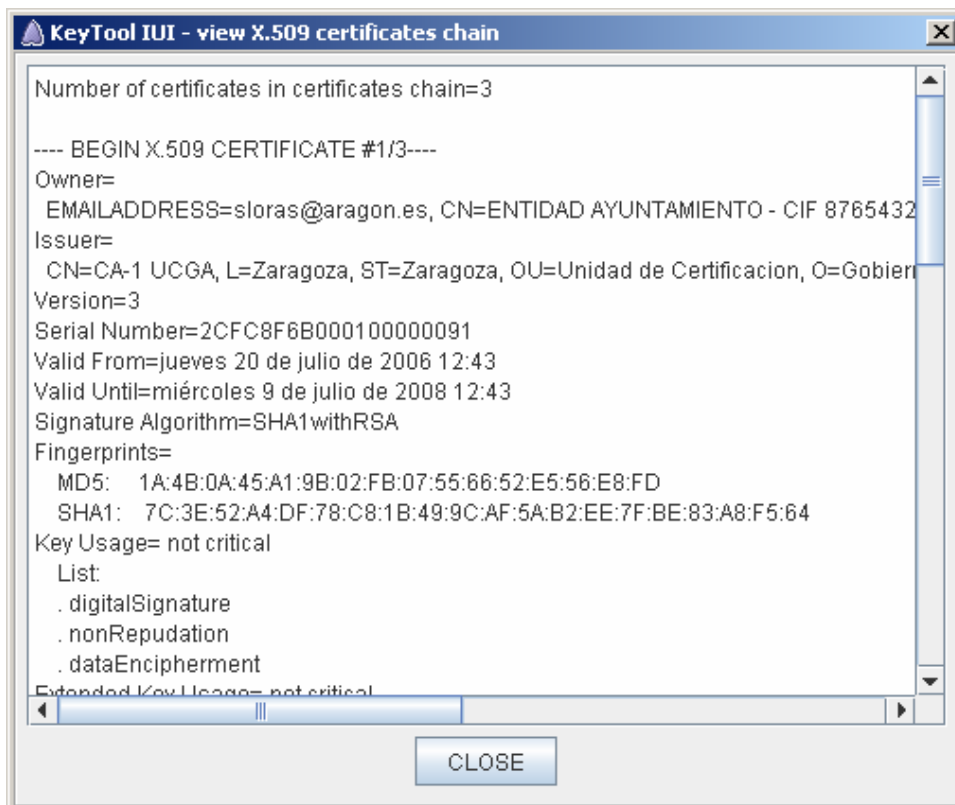
La aplicación dispone también de un menú para permitir la visualización de certificados almacenados en ficheros. Para ello, acceder al menú: “View → File → Certificate” y seleccionar la opción deseada en función del formato del fichero.



Se mostrará la ventana para seleccionar el fichero:



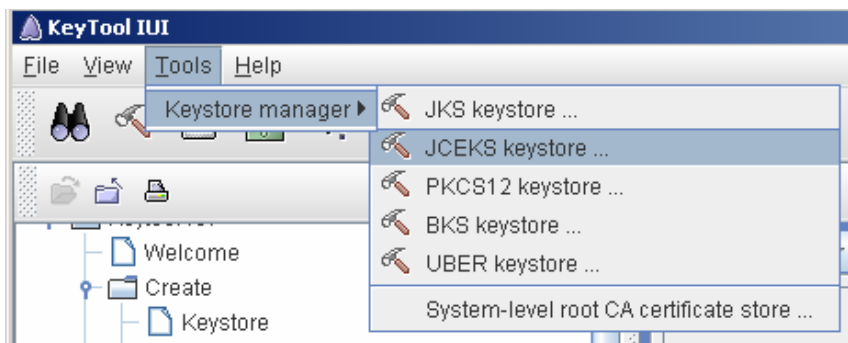
Una vez seleccionado el fichero con el certificado o cadena de certificados, se muestra en modo texto la información:



5 Gestión de los almacenes de certificados y claves

Además de visualizar el contenido de los almacenes de certificados, la aplicación KeyTool IUI permite desde otra opción de menú, gestionarlos, es decir, cambiar la clave de acceso al almacén, ver la información de cada entrada, borrar una entrada, renombrarla o copiarla, y en el caso de claves secretas o privadas, es posible también, cambiar la password de acceso a la misma.

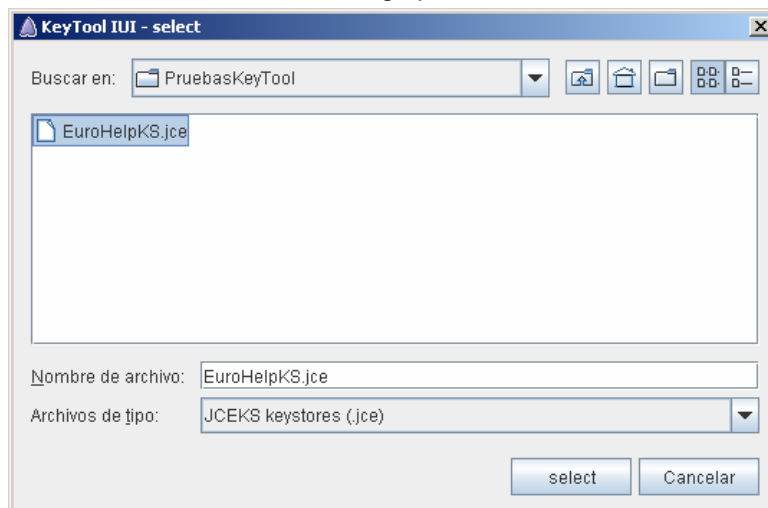
Para ello, acceder al menú: "Tools → Keystore manager" y seleccionar la opción deseada en función del formato del almacén.



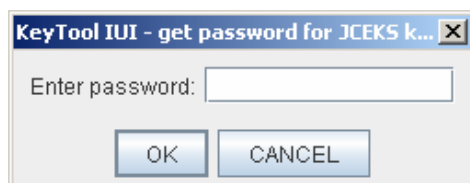
Se puede acceder también mediante el botón de la barra de herramientas:



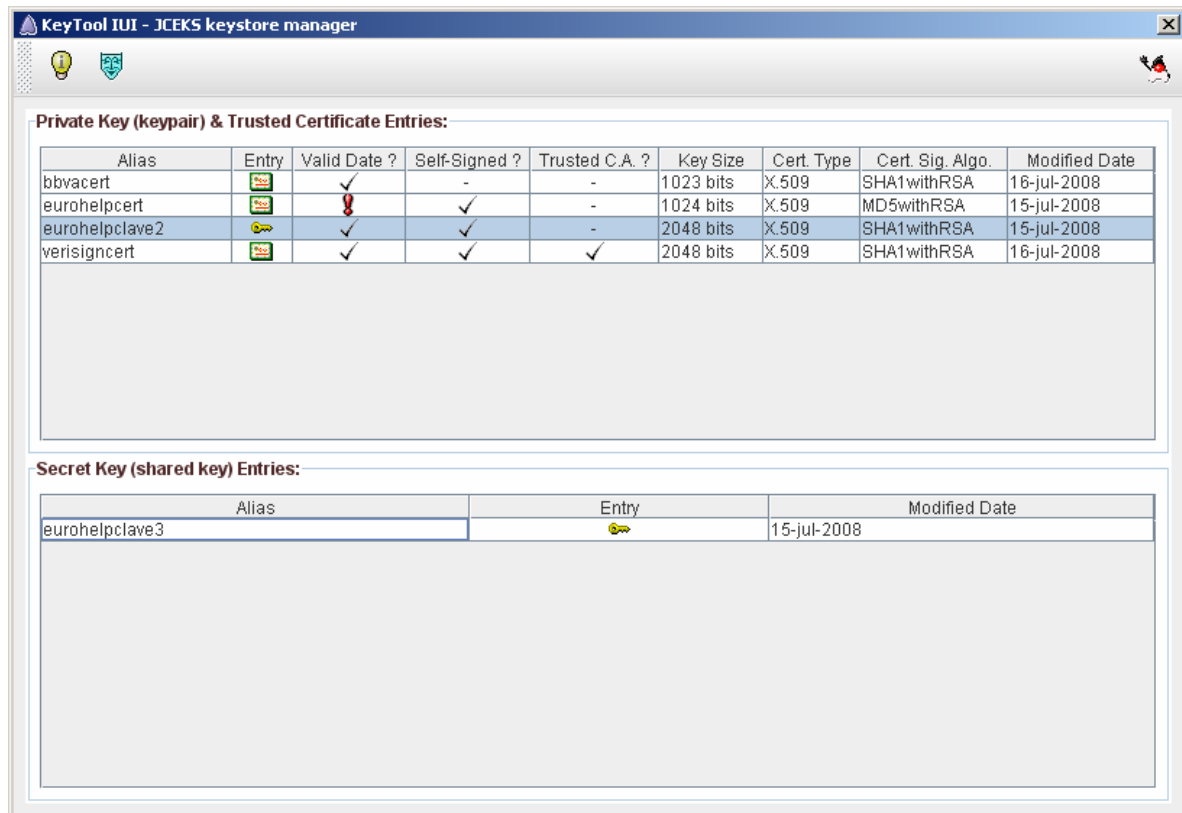
Se mostrará un cuadro de diálogo para seleccionar el almacén:



Como en este caso se puede modificar el contenido del almacén, la aplicación pide la clave de acceso al almacén:



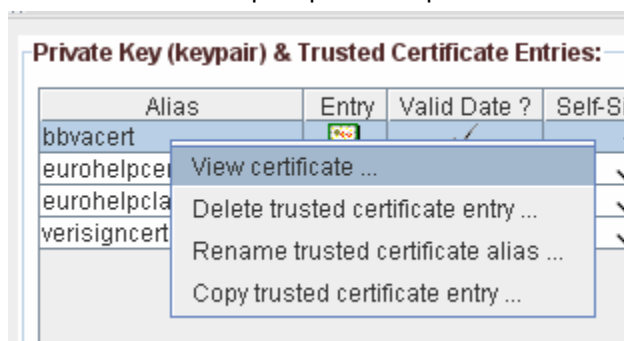
Si se introduce correctamente, se mostrará la siguiente ventana:



En cada entrada, pinchando con el botón derecho del ratón, se mostrará un menú contextual, distinto si la entrada es un certificado o una clave.

5.1 Gestión de los certificados de un almacén

El menú contextual que aparece al pinchar sobre un certificado es el siguiente:

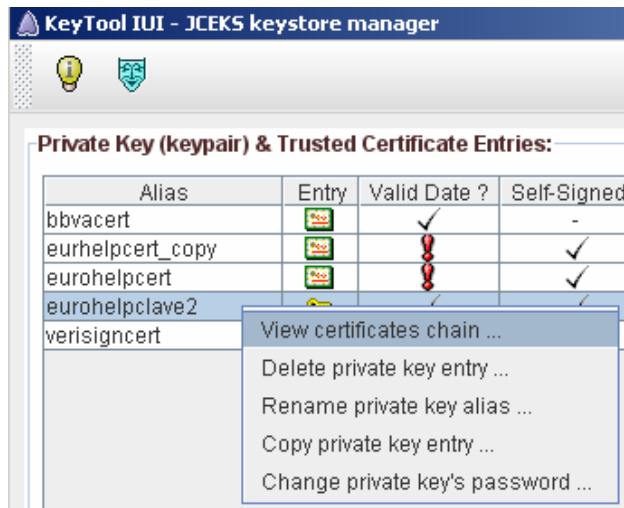


Permite:

- Ver certificado. (View certificate ...). Tal como se ha explicado en el punto 4.2.1
- Borrar la entrada seleccionada (Delete trusted certificate entry ...). Pedirá confirmación.
- Cambiar el nombre del alias del certificado.
- Crear una copia del certificado. Pedirá el nuevo alias para asignárselo a la copia.

5.2 Gestión de las claves (privadas o secretas) de un almacén

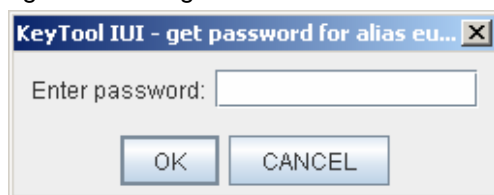
El menú contextual que aparece al pinchar sobre una clave es el siguiente:



Permite:

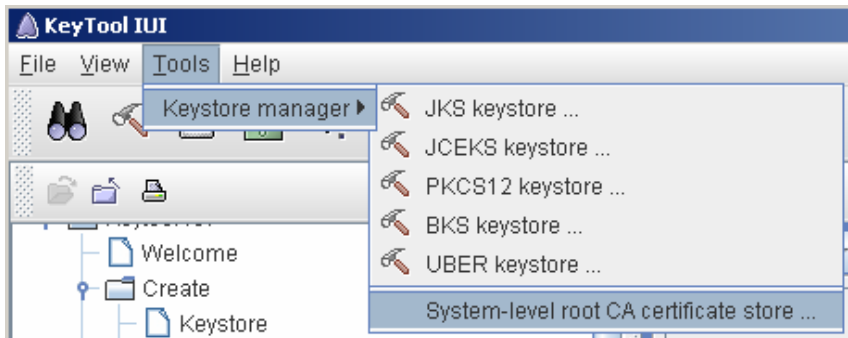
- Ver la información del par de claves pública y privada (View certificates chain ...), o de la clave secreta (View infos ...), en función del tipo de clave seleccionada.
- Borrar la entrada seleccionada (Delete private/secret key entry ...). Pedirá confirmación.
- Cambiar el nombre del alias de la clave (Rename private/secret key alias ...).
- Crear una copia de la clave (Copy private/secret key entry ...). Pedirá el nuevo alias para asignárselo a la copia.
- Cambiar la password de acceso a la clave (Change private/secret key's password ...).

Las claves en un almacén, ya sean privadas o secretas, están protegidas con una password de acceso. Para hacer cualquiera de las acciones anteriores (excepto ver información) se pedirá la password con un cuadro de diálogo como el siguiente:



5.3 Gestión del almacén por defecto de la implementación Java


La aplicación KeyTool IUI, al igual que en la opción de visualización, permite gestionar el almacén de certificados que viene por defecto con el JRE de Java (cacerts). Para ello acceder al menú: “Tools → Keystore manager → System-level root CA certificate store ...”

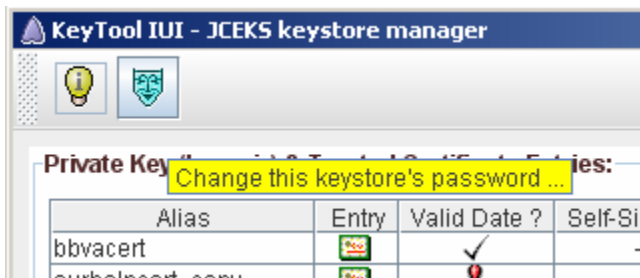


Se solicitará la password de acceso al almacén, que si no se ha cambiado, debería ser: “changeit”. A continuación se mostrará una ventana con los certificados en los que se confía. Se podrá gestionar dichos certificados de la forma descrita en el apartado 5.1.

5.4 Cambio de la password de acceso al almacén.

Los almacenes de certificados y claves pueden estar protegidos por password de acceso. La aplicación KeyTool IUI permite también cambiar dicha password.

Para ello, en la ventana de gestión del almacén, seleccionar el icono: . (Ver la siguiente imagen)



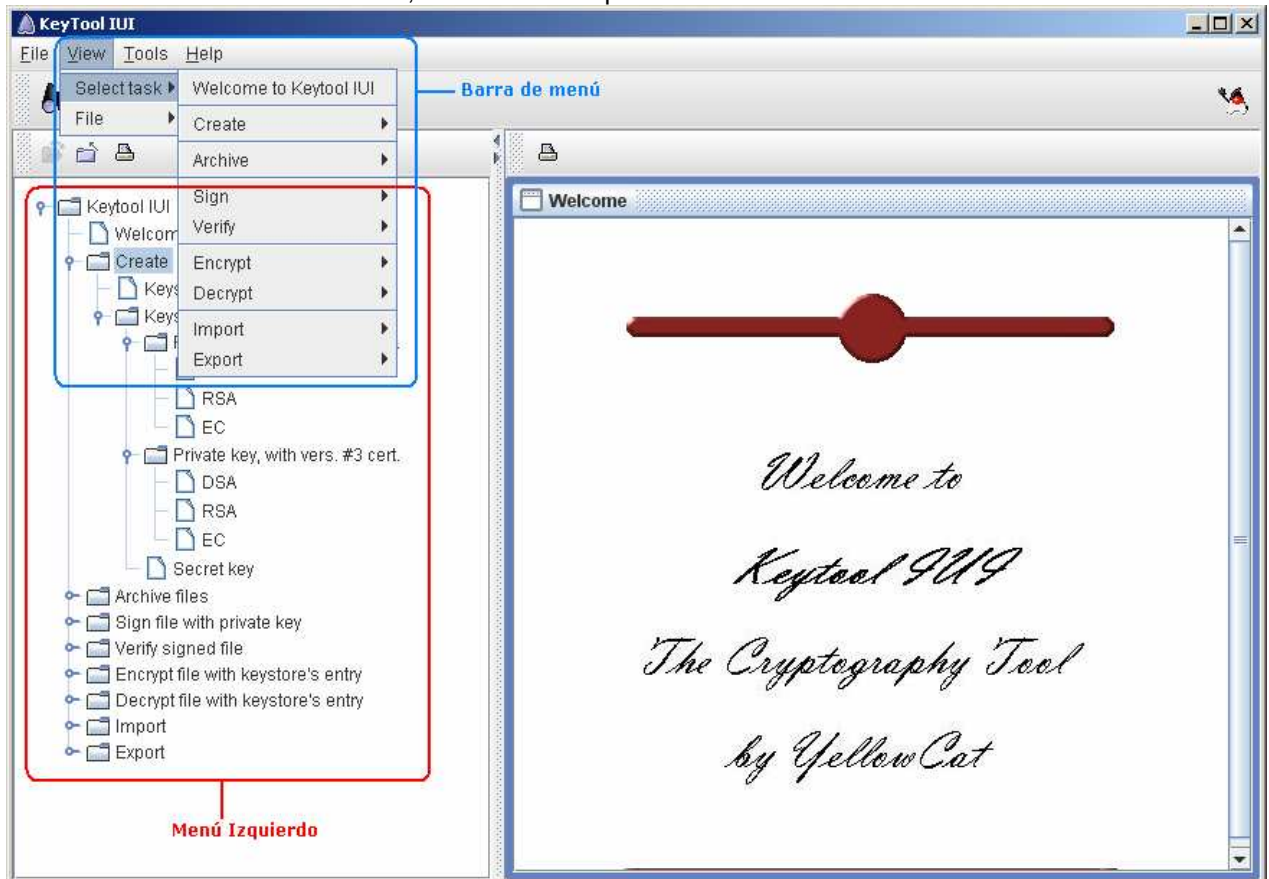
Se mostrará un cuadro de diálogo solicitando la nueva password y la confirmación, para detectar posibles equivocaciones al escribir.



6 Funcionalidades

En el presente capítulo se describen el resto de funcionalidades que implementa la aplicación KeyTool IUI.

A cada una de las tareas que se describirá se podrá acceder tanto desde el menú izquierdo de la aplicación como desde la barra de menú, a través de la opción: "View → Select task".



Únicamente se describirá el acceso a cada tarea a través del menú izquierdo, por considerar que el acceso desde la barra de menú se desprende del anterior.

Por otro lado, para todas las tareas cabe señalar que, de los formularios que se presentan, los campos con asterisco son obligatorios.

Inicialmente, el botón para llevar a cabo la tarea correspondiente aparece deshabilitado



Se habilitará cuando todos los campos obligatorios hayan sido rellenados




6.1 Creación de un almacén de certificados

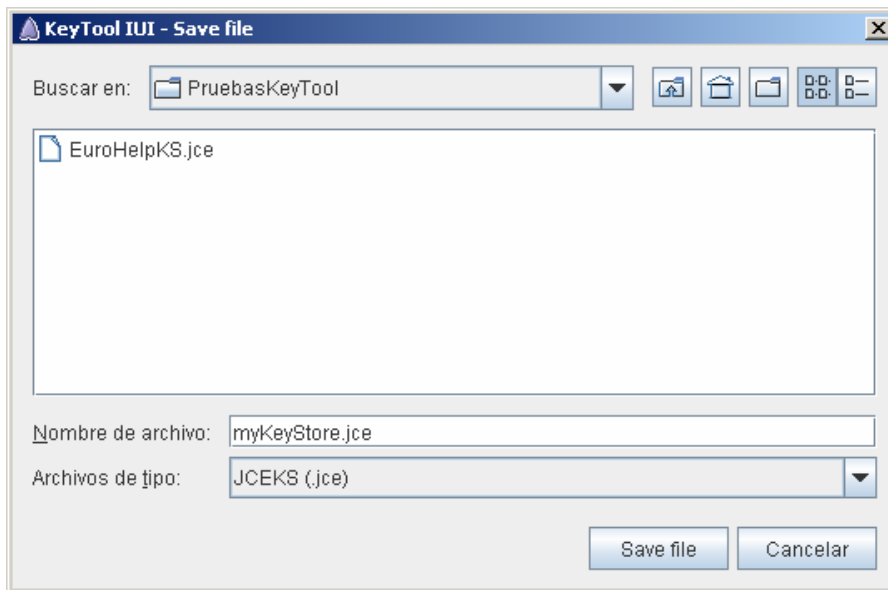
Para crear un nuevo almacén de certificados vacío, ir a la opción “Create → Keystore” del menú izquierdo de la aplicación.



En la parte derecha de la aplicación se mostrará el siguiente formulario:

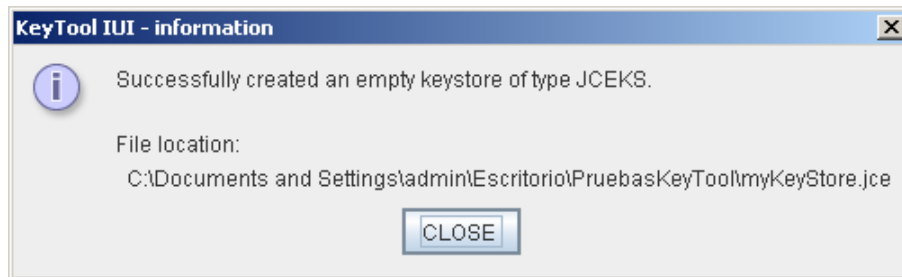
Seleccionar el formato para el almacén entre los disponibles en la aplicación: JKS, JCEKS, PKCS12, BKS y UBER (ver más información de los formatos aceptados por la aplicación en el punto titulado: “*Formatos admitidos para certificados y claves*” de este mismo manual).

Pinchar en el icono:  y en la ventana que aparece, seleccionar la ubicación preferida para el fichero del almacén e indicar un nombre.



Puede indicarse también una password de acceso para el almacén de certificados y claves. Se recomienda que se indique una.

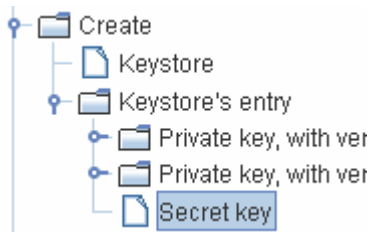
Pulsar el botón “OK” para terminar el proceso. Deberá aparecer un mensaje avisando de que la operación se ha realizado con éxito.



6.2 Creación de una clave secreta y asignación a un almacén

En esta tarea se creará una nueva clave secreta, de uso en el cifrado simétrico, y se incluirá en un almacén previamente definido.

Para crear una nueva clave secreta ir a la opción “Create → Keystore’s entry → Secret key” del menú izquierdo de la aplicación.



En la parte derecha de la aplicación se mostrará el siguiente formulario:

Create secret key (shared key) entry

Source:

* Keystore file: JCEKS BKS UBER

Keystore password:


Target:

New Entry - Secret Key:

* Secret key algorithm:

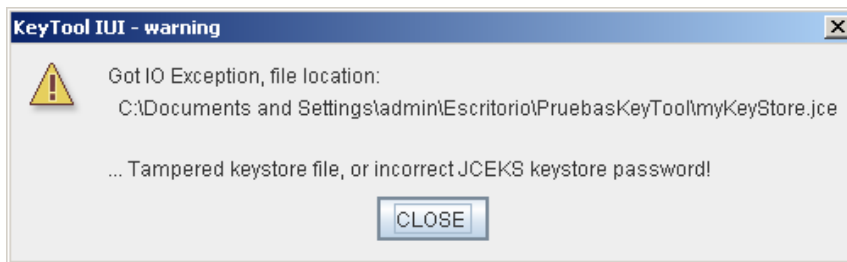
OK

Seleccionar el formato para el almacén entre los disponibles: JCEKS, BKS y UBER. No todos los formatos de almacén disponibles en la aplicación permiten incluir en ellos claves secretas, así que únicamente se puede seleccionar entre los formatos que si lo permiten.

Pinchar en el icono  y en la ventana que aparece, seleccionar el fichero del almacén.

Elegir entre los disponibles un algoritmo de creación de la clave (Secret key algorithm). Pulsar OK para terminar el proceso.

Si el almacén tuviera password de acceso, y no se ha indicado una en el campo "Keystore password" aparecerá el siguiente mensaje:



En caso contrario, aparecerá una ventana con el contenido del almacén y en la parte inferior un formulario para rellenar:

Enter new secret key entry's alias:	<input type="text"/>
Enter new password:	<input type="password"/>
Confirm new password:	<input type="password"/>
<input type="button" value="OK"/> <input type="button" value="CANCEL"/>	

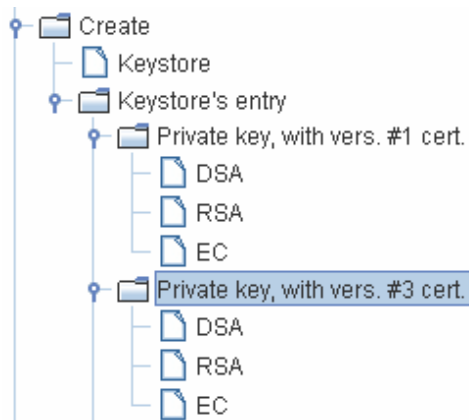
En él, señalar un nombre para la clave secreta y una password de acceso. Todos los campos del formulario son obligatorios (a pesar de no estar indicado con el asterisco), salvo para el caso de almacenes en formato PKCS12.

Pulsar OK para terminar la operación o CANCEL para cancelar.

6.3 Creación de una clave privada y asignación a un almacén

En esta tarea se creará una clave privada, de uso en el cifrado asimétrico, y se incluirá en un almacén previamente definido.

Para crear un nuevo par ir a la opción "Create → Keystore's entry" del menú izquierdo de la aplicación. Desplegar el nodo en función de si se desea crear una par de claves con versión 1 del estándar X.509 (Private key, with vers. #1 cert.) o versión 3 (Private key, with vers. #3 cert.). Dentro de dicho nodo, seleccionar el tipo de algoritmo deseado para generar la clave: DSA (Digital Signature Algorithm), RSA (Rivest, Shamir, Adleman) y EC (Elliptic Curve).



En la parte derecha de la aplicación se mostrará el siguiente formulario:

Create RSA private key entry, with vers. #1 cert

Source:

JKS
 JCEKS
 PKCS12
 BKS
 UBER

* Keystore file:

Keystore password:

Target:

Private Key:

* Key size (bits): 2048

* Signature algorithm: SHA1withRSA

* Validity (days): 1095

Cert. - X.500 Distinguished Name:

* Common (or domain) name:

Organizational unit:

Organization:

City or locality:

State or province:

* 2-letter country code: US (United States)

OK

Seleccionar el formato para el almacén entre los disponibles: JKS, JCEKS, PKCS12, BKS y UBER.

Rellenar los campos obligatorios del formulario, de forma similar a como se ha descrito en el punto anterior. Rellenar los campos opcionales que se desee.

Respecto a las extensiones, campos opcionales al final del formulario (ver imagen siguiente), en la ayuda de la aplicación se muestra qué opciones activar en la creación del par de claves para diferentes casos de uso.

Cert. Extension - KeyUsage:

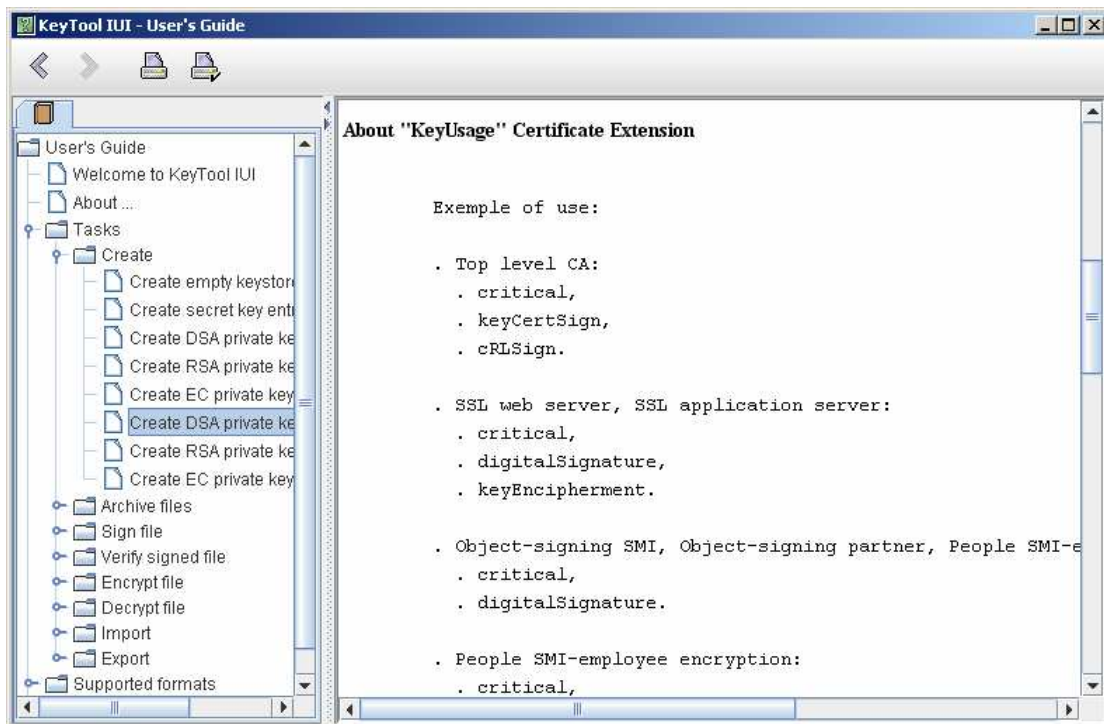
ENABLED?

- CRITICAL?
- digitalSignature:
- nonRepudiation:
- keyEncipherment:
- dataEncipherment:
- keyAgreement:
- keyCertSign:
- cRLSign:
- encipherOnly:
- decipherOnly:

Cert. Extension - ExtKeyUsage:

ENABLED?

- CRITICAL?
- serverAuth:
- clientAuth:
- codeSigning:
- emailProtection:
- ipsecEndSystem:
- ipsecTunnel:
- ipsecUser:
- timeStamping:
- OCSPSigning:
- Microsoft Enrollment Infrastructure: smartcardlogin:
- Microsoft Crypto 2.0: server gated crypto
- Microsoft Crypto 2.0: serialized
- Microsoft Crypto 2.0: EFS crypto
- Microsoft Crypto 2.0: EFS recovery
- Adobe: CDS PKI
- unknown key usage



Pulsar OK para crear la clave y comenzar el proceso de asignación al almacén. Si el formulario se ha rellenado correctamente, aparecerá la ventana con el contenido del almacén y el mismo formulario que en el punto anterior (claves secretas). Proceder de forma similar.

En este caso, si el almacén es de tipo PKCS12, no será necesario rellenar la password de acceso a la clave.

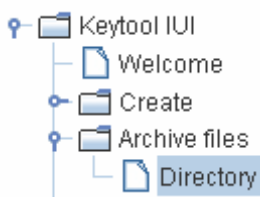
Problema conocido

Existe un problema reconocido por el autor de la aplicación en la creación de claves con algoritmo DSA, versión 1 o versión 3. Para los almacenes de tipo: PKCS12, BKS y UBER, aunque aparezca como algoritmo de firma (Signature algorithm) SHA1withDSA, la clave se estará almacenando como DSA.

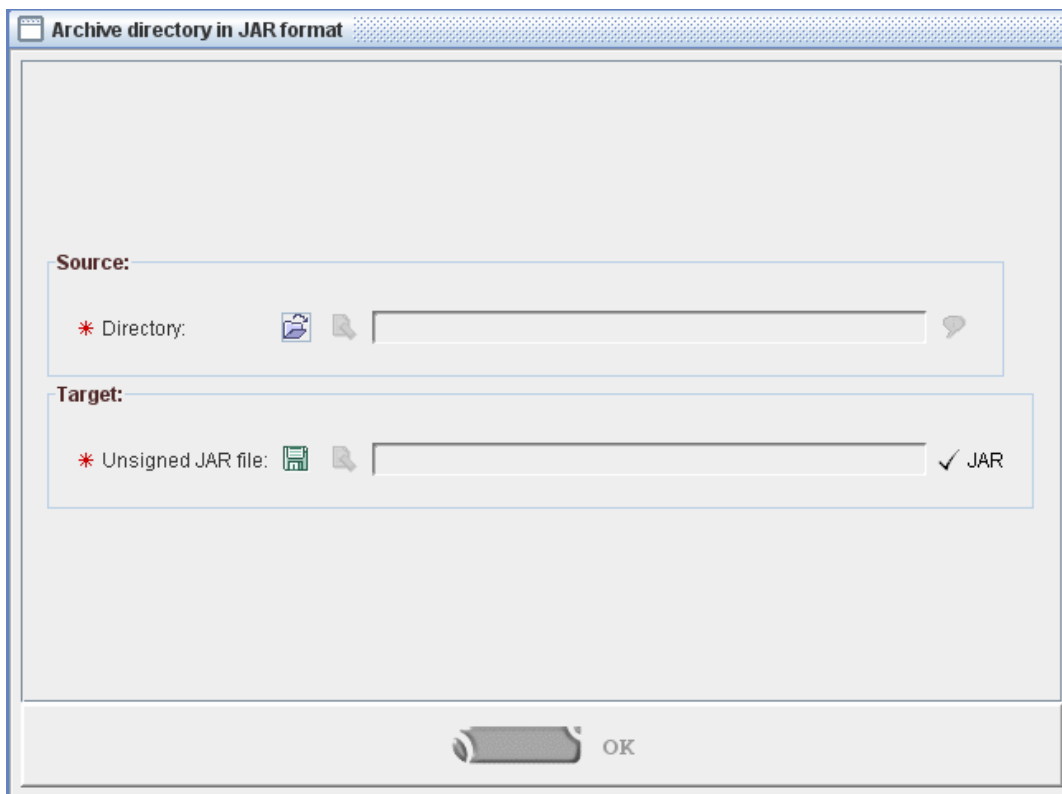
6.4 Creación del jar de un directorio

En esta tarea se creará un fichero .jar con el contenido de un directorio dado.

Para crear un jar con el contenido de un directorio ir a la opción “Archive files → Directory” del menú izquierdo de la aplicación.



En la parte derecha de la aplicación se mostrará el siguiente formulario:



Seleccionar el directorio cuyo contenido se quiere empaquetar en un jar.

Seleccionar el directorio y el nombre del fichero .jar que se creará.

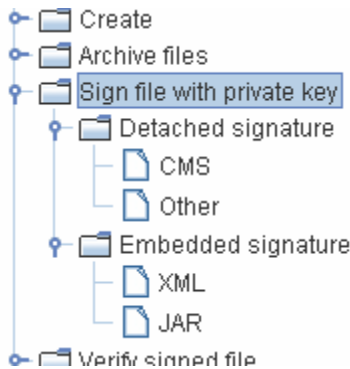
Pulsar OK para terminar la operación.

6.5 Firma digital de documentos

Se describirá en este punto la funcionalidad que ofrece la aplicación KeyTool IUI para firmar documentos.

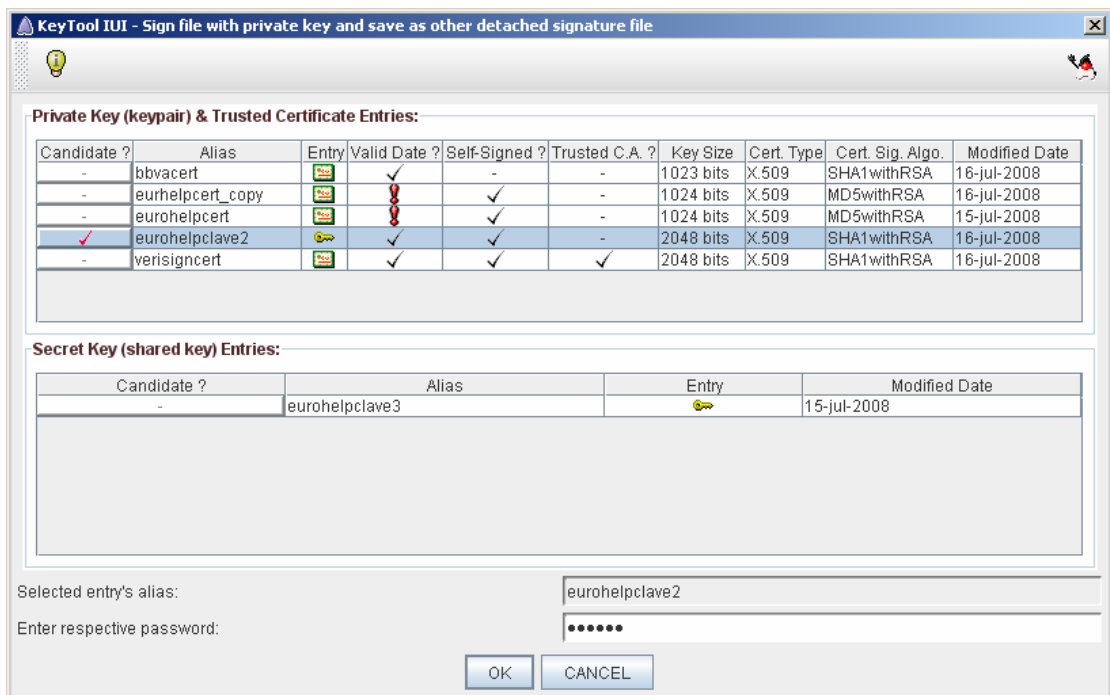
Permite firma en modo 'detached' cualquier documento, o en modo 'embedded' ficheros XML o jar. La diferencia entre ambos métodos de cifrado es que en modo 'detached' el fichero de firma generado no incluye los datos firmados, sin embargo en el modo 'embedded' sí.

Para firmar un documento, ir a la opción "Sign file with private key" del menú izquierdo de la aplicación y seleccionar el método de firma deseado.



El proceso de firma en todos los casos es el mismo:

1. En el formulario que se mostrará en la parte derecha de la aplicación, rellenar los campos obligatorios (marcados con asterisco) y los opcionales que se desee. Pulsar OK para continuar con el proceso.
2. Aparecerá una ventana mostrando el contenido del almacén seleccionado. Se indicarán las claves que pueden ser utilizadas para firmar con el símbolo . Seleccionar entre todas ellas la clave con la que se desea firmar:



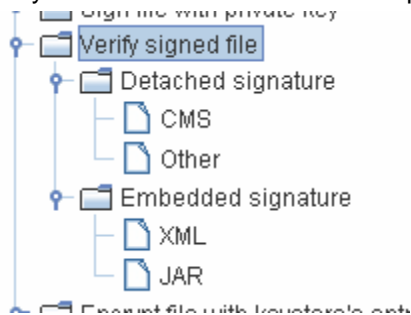
3. En el formulario inferior indicar la password de acceso de la clave seleccionada.

Al pulsar OK aparecerá un mensaje de confirmación indicando que se ha realizado con éxito la operación.

6.6 Validación de la firma digital de un documento

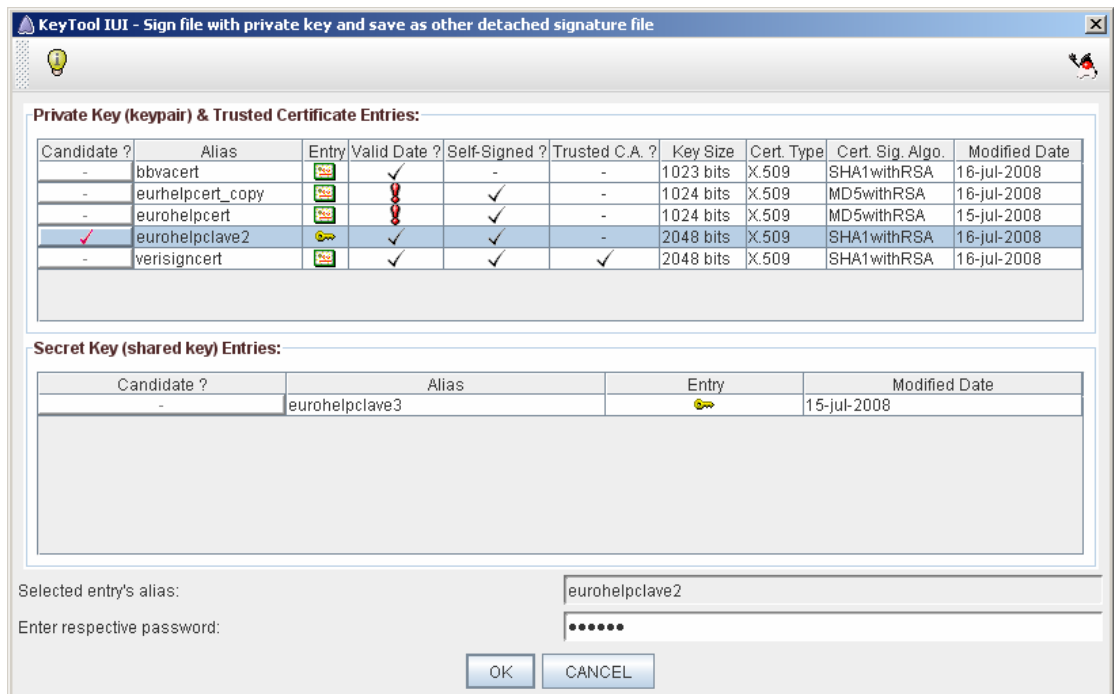
Se describirá en este punto la funcionalidad que ofrece la aplicación KeyTool IUI para validar documentos firmados utilizando los métodos descritos en el punto anterior.

Para validar la firma de un documento, ir a la opción “Verify signed file” del menú izquierdo de la aplicación y seleccionar el método con el que se sabe que está firmado el documento.



El proceso de validación es muy parecido en todos los casos:

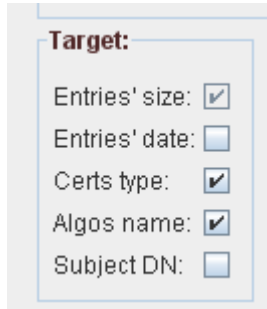
1. En el formulario que se mostrará en la parte derecha de la aplicación, rellenar los campos obligatorios (marcados con asterisco) y los opcionales que se desee. Pulsar OK para continuar con el proceso.
2. En el modo ‘Other detached signature’ aparecerá una ventana mostrando el contenido del almacén elegido. Se indicarán los certificados que pueden ser utilizados para validar la firma con el símbolo . Seleccionar el de la persona o entidad que firmó el documento.



En el formulario inferior indicar la password de acceso de la clave seleccionada.

3. En el modo ‘JAR embedded signature’ se deberán seleccionar los puntos que se quieren validar

(apartado 'target' del formulario).



Target:

Entries' size:

Entries' date:

Certs type:

Algos name:

Subject DN:

4. Al pulsar OK aparecerá un mensaje de confirmación indicando que se ha realizado con éxito la operación.

En el modo 'JAR embedded signature' el mensaje de confirmación es el devuelto por la aplicación jarsigner al invocarla con la opción `-verify`.

6.7 Cifrado de documentos utilizando la entrada de un almacén

Se describirá en este punto la funcionalidad que ofrece la aplicación KeyTool UI para cifrar documentos utilizando una entrada previamente almacenada en un almacén de certificados y claves.

Para cifrar documentos, ir a la opción "Encrypt file with keystore's entry" y seleccionar la opción deseada en base al tipo de cifrado requerido.

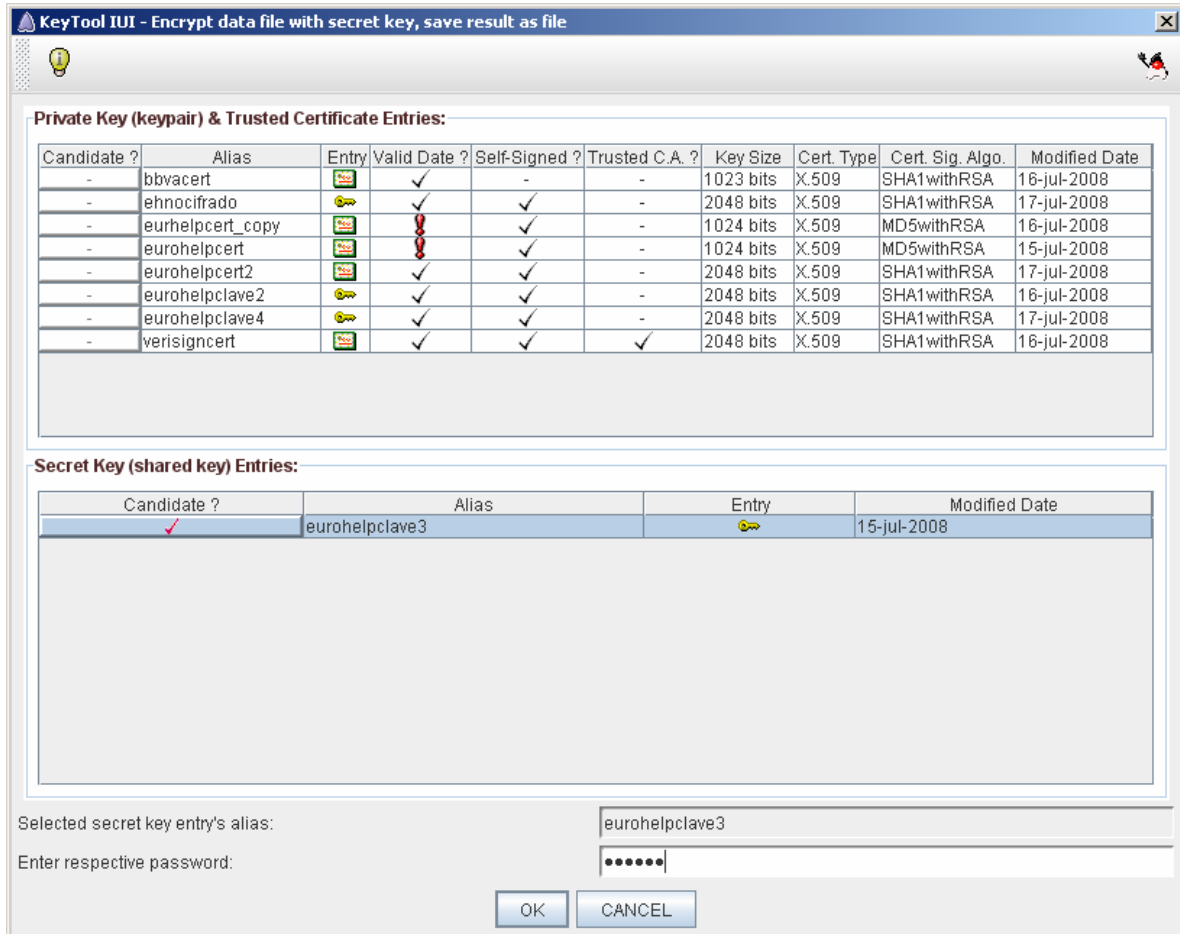


- Secret key: Utilizado en cifrado simétrico.
- RSA trusted certificate: Utilizado para cifrar documentos con la clave pública de una persona. De este modo nos aseguramos que únicamente dicha persona pueda leer el mensaje.
- RSA private key: Utilizado para cifrar un documento con nuestra clave privada. De este modo nos aseguramos que únicamente podrán leer el mensaje aquellas personas que tengan nuestra clave pública.

El proceso de cifrado es similar en todos los casos:

1. En el formulario que se mostrará en la parte derecha de la aplicación, rellenar los campos obligatorios (marcados con asterisco).
2. En cifrado con clave secreta el algoritmo de cifrado vendrá definido por el algoritmo que se indicó al crear dicha clave, en los otros dos casos se puede seleccionar de una lista dada (campo 'RSA Encryption Algorithm').

3. Pulsar OK para continuar con el proceso.
4. Aparecerá una ventana mostrando el contenido del almacén seleccionado. Se indicarán las entradas que pueden ser utilizadas para cifrar el documento con el símbolo . Seleccionar entre todas ellas la clave o el certificado con el que se desea cifrar:



5. En el formulario inferior indicar la password de acceso de la clave seleccionada.

Al pulsar OK aparecerá un mensaje de confirmación indicando que se ha realizado con éxito la operación.

Limitaciones

Para el cifrado con clave secreta, los algoritmos de cifrado Hmac (md5, sha1, sha256, ...) no funcionan correctamente.

Para los cifrados RSA, el tamaño de los datos a cifrar no debería ser mayor (en bytes) de:
 $(\text{tamaño de la clave de cifrado} / 8) - 11$


6.8 Descifrado de documentos utilizando una clave de un almacén

Se describirá en este punto la funcionalidad que ofrece la aplicación KeyTool IUI para descifrar documentos utilizando una clave (privada o secreta) existente en un almacén.

Para descifrar documentos, ir a la opción “Decrypt file with keystore’s entry” y seleccionar la opción deseada en base al tipo de cifrado del documento.



El proceso es similar en ambos casos:

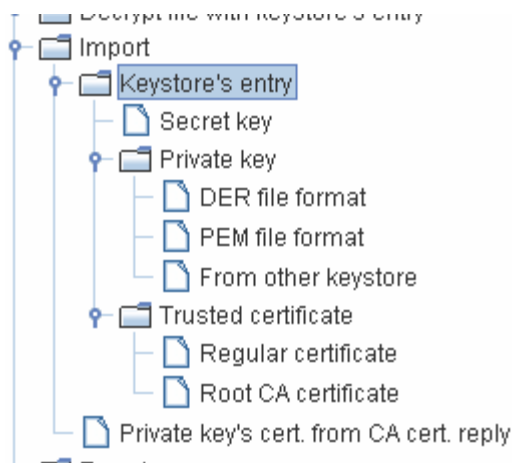
1. En el formulario que se mostrará en la parte derecha de la aplicación, rellenar los campos obligatorios (marcados con asterisco).
2. En cifrado con clave privada RSA habrá que indicar el algoritmo con el que está cifrado el documento, en el caso de clave secreta no es necesario.
3. Pulsar OK para continuar con el proceso.
4. Aparecerá una ventana mostrando el contenido del almacén seleccionado. Se indicarán las entradas que pueden ser utilizadas para descifrar el documento con el símbolo . Seleccionar entre todas ellas la clave con la que se desea descifrar.
5. En el formulario inferior indicar la password de acceso de la clave seleccionada.
6. Pulsar OK para terminar la operación.

En este caso, si todo ha ido correctamente no aparecerá mensaje de confirmación, pero sí se indicará un mensaje de error en el caso de que haya algún problema.

6.9 Importación de claves y certificados a un almacén

Se describirá en este punto la funcionalidad que ofrece la aplicación KeyTool IUI para importar claves secretas, claves privadas y certificados a un almacén previamente definido.

Para importar documentos, ir a la opción “Import → Keystore’s entry” y seleccionar la opción deseada en base al tipo de fichero que deseamos importar.

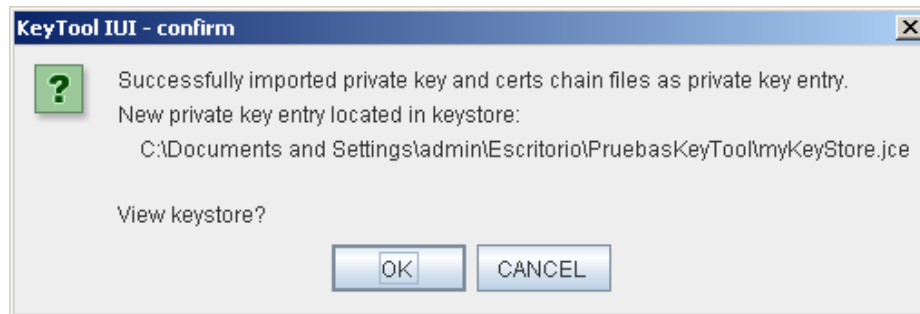


- **Secret key:** Para importar claves secretas.
Se deberá conocer además de la password de acceso a la clave, el algoritmo de cifrado para el que se creó dicha clave. La aplicación no puede comprobar si el algoritmo introducido es correcto o no, así que debe ser el usuario el que se asegure de ello.
- **Private key:** Para importar claves privadas.
En este caso, habrá que seleccionar el fichero desde el que importar la clave privada y el fichero desde el que importar el certificado (o cadena de certificados) con la clave pública.
Para importar una clave desde un fichero en formato DER, se deberá conocer el algoritmo de cifrado para el que se creó dicha clave.
- **Trusted certificate (Regular certificate):** Para importar certificados de confianza.
Se seleccionará el formato del fichero a importar entre los tres posibles: DER, PKCS#7 y PEM.
La aplicación permite seleccionar si se comprobará que la CA del certificado sea una de las incluidas en el almacén de autoridades certificadoras de raíz (Root CA certs store).
El procedimiento para importar certificados de confianza de Autoridades Certificadoras raíz (Root CA certificate), se explicará más adelante.

El proceso para la importación de claves y certificados es similar en todos los casos:

1. Rellenar los campos del formulario que se mostrará en la parte derecha de la aplicación.
2. Pulsar OK para continuar con el proceso.
3. Si se ha seleccionado 'importar desde otro almacén' (From other keystore), aparecerá una ventana con el contenido del almacén origen. Se indicarán las entradas que pueden ser utilizadas para importar con el símbolo ✓. Seleccionar entre todas ellas la clave que se desea importar e indicar su password de acceso en el formulario inferior. Si se ha seleccionado cualquier otra opción se pasará directamente al punto 4.
4. Al pulsar OK para continuar con el proceso, aparecerá una ventana mostrando el contenido del almacén seleccionado como destino.
5. Rellenar los campos que aparecen en el formulario inferior, indicando el alias para la clave importada y la password de acceso a la misma.
6. Pulsar OK para continuar con el proceso. Aparecerá un cuadro de diálogo indicando que se ha

completado con éxito la operación y preguntando si se desea ver el contenido del almacén.

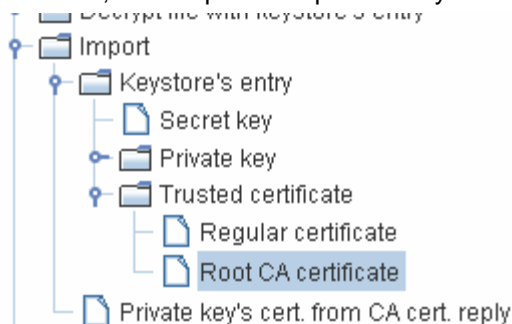


7. Si se pulsa OK, aparecerá de nuevo la ventana con el contenido del almacén, donde podrá comprobarse que se ha importado la clave con éxito.

6.9.1. Importación al almacén de Autoridades Certificadoras Raíz: cacerts

Como se ha comentado en puntos anteriores, la aplicación KeyTool IUI permite la gestión del almacén de entidades certificadoras (CAs) de confianza que viene por defecto en la implementación Java (cacerts). Así pues, con respecto a la importación de certificados, la aplicación permite importar los certificados de dichas entidades, para confiar automáticamente en cualquier certificado autorizado por las mismas.

Para ello, ir a la opción "Import → Keystore's entry → Root CA certificate".



El proceso es similar al descrito en el punto anterior, la única diferencia es que en el campo del almacén viene seleccionado por defecto el definido por la implementación Java y no es posible seleccionarlo.


6.9.2. Importación desde el fichero de respuesta de una CA

Cuando una entidad desea que una CA autorice un certificado suyo, envía una solicitud a una CA raíz, mediante un mensaje CSR (Certificate Signing Request) firmado por su clave privada. La entidad raíz le responde con un mensaje que utiliza la sintaxis definida en PKCS#7. La aplicación KeyTool IUI permite importar desde el fichero de respuesta de la CA raíz.

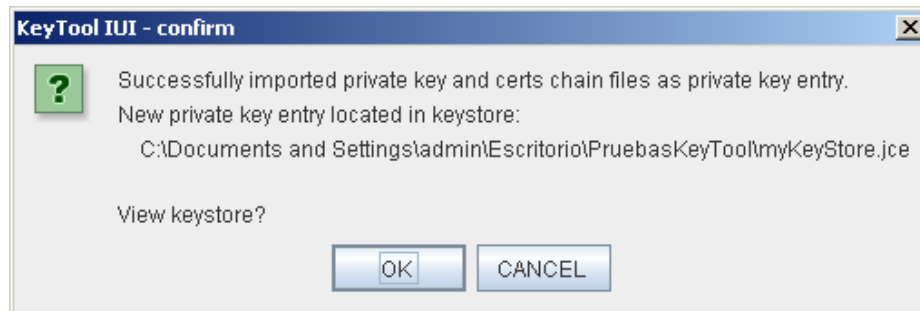
Para ello, ir a la opción "Import → Private key's cert. from CA cert. reply".



Los pasos a seguir son los siguientes:

1. En el formulario que se mostrará en la parte derecha de la aplicación, rellenar los siguientes datos:
 - CA cert. reply file: con el fichero de respuesta a la solicitud CSR.
 - Keystore file: con el almacén que contiene la clave privada con la que se firmó la solicitud CSR.
2. Al pulsar OK para continuar con el proceso, aparecerá una ventana con el contenido del almacén origen. Se indicarán las entradas que pueden ser utilizadas para la importación, con el símbolo . Seleccionar entre todas ellas la clave con la que se firmó la solicitud.
3. Completar el formulario inferior.
4. Pulsar OK para continuar con el proceso.

Aparecerá un mensaje indicando que se ha realizado con éxito la operación y preguntando si se desea ver el almacén con los cambios efectuados.



6.10 Exportación de claves y certificados a fichero

Se describirá en este punto la funcionalidad que ofrece la aplicación KeyTool IUI para exportar claves y certificados que estén incluidos en un almacén a un fichero.

Para importar documentos, ir a la opción "Export → Keystore's entry" y seleccionar la opción deseada en base al tipo de clave o certificado que vamos a exportar.




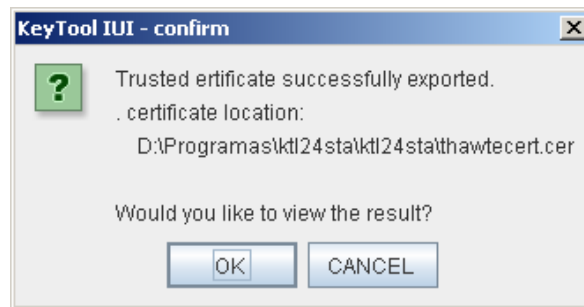
- Secret key: Para exportar claves secretas.
- Private key: Para exportar claves privadas.
Se exportará tanto la clave privada como la cadena de certificados de la clave pública, es decir, el certificado con la clave pública en sí y los certificados de las entidades que validan el par clave privada-clave pública, si los hubiera.
- Trusted certificate (Regular certificate): Para exportar certificados de confianza.

El proceso para la exportación de claves y certificados es similar en los tres casos:

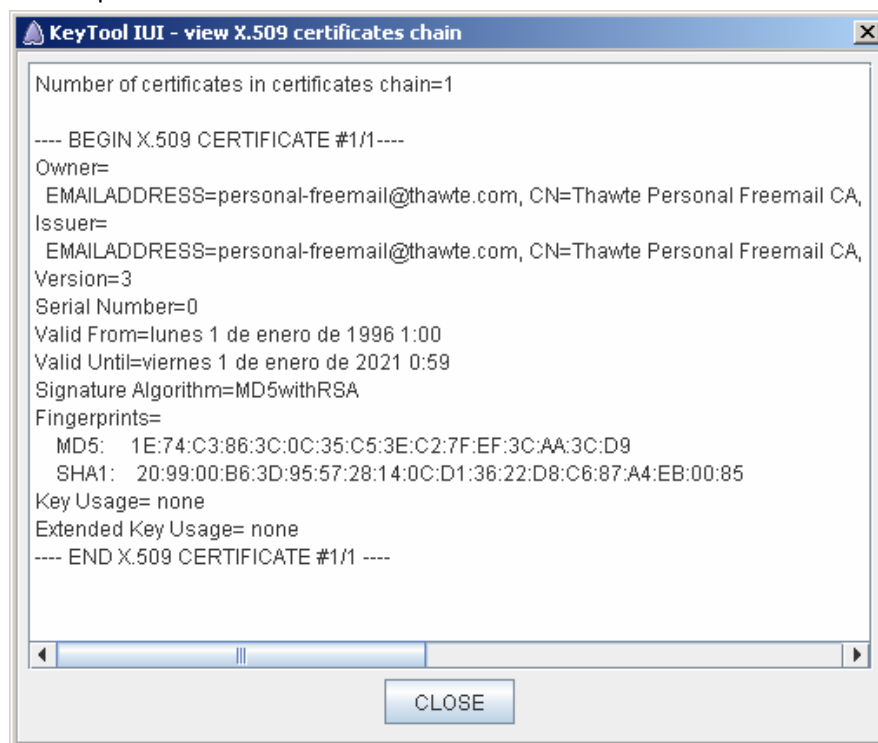
1. Rellenar los campos del formulario que se mostrará en la parte derecha de la aplicación. Seleccionar el formato del fichero que contendrá la clave y/o certificados. Los formatos que admite la aplicación para los ficheros exportados son:

Formato	Extensión del Fichero	
DER	.der	Fichero binario
PEM	.pem	Fichero ascii
PKCS#7	.p7b	Fichero binario

2. Pulsar OK para continuar con el proceso.
3. Aparecerá una ventana con el contenido del almacén donde se encuentra la clave o certificado a exportar. Se indicarán las entradas que pueden ser utilizadas para importar con el símbolo . Seleccionar entre todas ellas la que se desea importar e indicar la password de acceso en el formulario inferior para el caso de las claves.
4. Aparecerá un cuadro de diálogo indicando que se ha completado con éxito la operación y preguntando si se desea ver los datos de la clave o certificado.



5. Si se pulsa OK, aparecerá un cuadro de diálogo donde podrán comprobarse los datos de la clave o certificado exportado.



6.10.1. Exportación del certificado con la clave pública

Lo que se ha nombrado a lo largo de todo el manual como clave privada, es como ya se ha dicho, la suma de: la clave privada, el certificado con la clave pública y los certificados de las entidades que validan la clave. Este punto describe la funcionalidad que ofrece la aplicación KeyTool IUI para obtener el primer certificado de la cadena, es decir, el certificado con la clave pública asociada a la clave privada.

Para ello, ir a la opción "Export → Private key's first certificate in chain" y seleccionar la opción deseada:



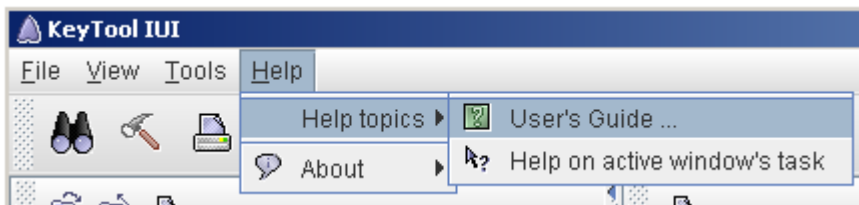
- As certificate signing request file: Para crear el mensaje CSR (Certificate Signing Request) comentado en el punto 6.9.2.
- As simple certificate file: Para exportar el certificado con la clave pública asociada a una clave privada.

El proceso es similar al descrito en el punto anterior, por lo que no se describirá.

7 Acceso a la ayuda de la aplicación

El autor de la aplicación KeyTool IUI ofrece una ayuda muy simple, pero en la que muestra información relevante en muchos casos, como son limitaciones a las funcionalidades que se ofrecen (han sido descritas todas en este manual) y casos de uso en la creación de certificados. Por ello, se considera importante el conocer como se accede a la ayuda de la aplicación.

Para acceder a la ayuda de la aplicación, ir a la opción "Help → Help topics → User's Guide ...".



Se puede acceder también mediante el botón de la barra de herramientas:



Aparecerá una ventana con dos paneles, a la izquierda los puntos de la ayuda y a la derecha el panel donde se muestra la explicación de cada punto:

