

Cybertech de Colombia Ltda.

Documentación de CryptoVault 1.8.

Tabla de Contenido.

Mejoras con respecto a la versión 1.7	7
Antes de instalar CryptoVault: Requerimientos.	8
Instalación de la aplicación en ambientes Windows.	8
Instalación de la aplicación en otros sistemas operativos ('sabor' CLI).	9
Uso de la variedad CLI.....	11
a. <i>Configurando el almacén de llaves.</i>	11
b. <i>Adicionando un nuevo destinatario.</i>	12
c. <i>Eliminando un destinatario existente.</i>	12
d. <i>Cifrando un archivo.</i>	12
e. <i>Descifrando un archivo.</i>	13
f. <i>Descifrando un archivo y verificando el remitente.</i>	14
g. <i>Listando los destinatarios de la aplicación.</i>	14
h. <i>Cambiando la contraseña del almacén de llaves.</i>	14
i. <i>Referencia rápida.</i>	15
Uso de la variedad API.....	16
a. <i>Configurando el almacén de llaves.</i>	16
b. <i>Iniciando la conexión nativa.</i>	17
c. <i>Adicionando un nuevo destinatario.</i>	18
d. <i>Eliminando un destinatario existente.</i>	18
e. <i>Cifrando un archivo.</i>	19
f. <i>Descifrando un archivo.</i>	20
g. <i>Descifrando un archivo y verificando el remitente.</i>	20
h. <i>Obteniendo la lista de destinatarios de la aplicación</i>	21
i. <i>Cambiando la contraseña del almacén de llaves.</i>	22
j. <i>Terminando la conexión nativa.</i>	23
k. <i>Ejecutando los ejemplos de uso del API.</i>	23
Uso de la variedad JVA.....	25
a. <i>Configurando el almacén de llaves.</i>	25
b. <i>Configurando la ubicación de la aplicación.</i>	26
c. <i>Obteniendo la fachada para aplicaciones Java.</i>	26
d. <i>Adicionando un nuevo destinatario.</i>	26

e. <i>Eliminando un destinatario existente.</i>	27
f. <i>Cifrando un archivo.</i>	27
h. <i>Descifrando un archivo.</i>	28
i. <i>Descifrando un archivo y verificando el remitente.</i>	29
j. <i>Obteniendo la lista de destinatarios de la aplicación.</i>	29
k. <i>Cambiando la contraseña del almacén de llaves.</i>	29
l. <i>Ejecutando el ejemplo de uso.</i>	30
Uso de la variedad GUI.	32
a. <i>Configurando el almacén de llaves.</i>	32
a. <i>Adicionando un nuevo destinatario.</i>	33
b. <i>Eliminando un destinatario existente.</i>	36
c. <i>Cifrando un archivo.</i>	37
d. <i>Descifrando un archivo.</i>	41
e. <i>Consultando la ayuda de la aplicación</i>	45
f. <i>Cambiando la contraseña del almacén de llaves.</i>	45
Actualización de la CRL.	48
Asignación de permisos sobre los recursos en el directorio de instalación de CryptoVault.	50
Cambio periódico de la contraseña del almacén de llaves.	50
Creación de Almacenes de Llaves	61
Importar Certificados de Otras Autoridades Certificadoras	69
Renovación de Certificado	71
Recomendaciones relacionadas a Almacenes de Llaves	72

Introducción.

Este documento contiene información sobre la instalación, uso y procedimientos relacionados con Crypto Vault 1.6, organizada de la siguiente manera:

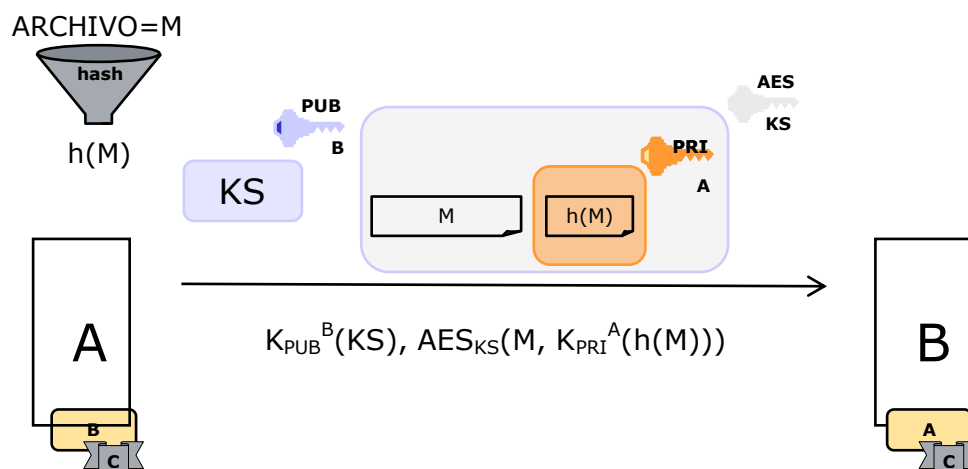
- 1) En el **capítulo 2, “Sobre CryptoVault.”**, se presentan las generalidades de la aplicación, particularmente, se explica en qué consiste su funcionalidad y cuáles son sus variedades o ‘sabores’. Además se presenta una lista las mejoras con respecto a la versión anterior.
- 2) En el **capítulo 3, “Instalando CryptoVault.”**, se listan los requerimientos mínimos para instalar CryptoVault, documentando el procedimiento de instalación tanto en ambientes Windows (por medio del instalador), como en otros sistemas operativos soportados.
- 3) En el **capítulo 4, “Usando CryptoVault.”**, se explica la forma de usar la aplicación, dependiendo del ‘sabor’ o variedad instalada.
- 4) En el **capítulo 5, “Procedimientos Administrativos relacionados con CryptoVault.”**, se explican las principales labores administrativas que han de tenerse en cuenta para mantener el funcionamiento óptimo de CryptoVault.
- 5) En el **capítulo 6, “Procedimientos Administrativos relacionados con CryptoVault.”**, se listan y se explican los códigos que puede retornar la aplicación cuando se ejecuta en modo de línea de comandos (CLI).
- 6) En el **capítulo 7, “Solucionando Problemas Frecuentes desde la Interfaz Gráfica.”**, se presentan guías de solución para algunos problemas frecuentes que pueden surgir a la hora de usar la aplicación desde la Interfaz Gráfica (GUI).
- 7) En el **capítulo 8, “Almacenes de Llaves compatibles con CryptoVault.”**, se hace una breve introducción a los almacenes de llaves PKCS12 que utiliza Crypto Vault, y además se explica cómo hacer tareas relacionadas con ellos como: crear, renovar y Agregar Autoridades Certificadoras.

Sobre CryptoVault.

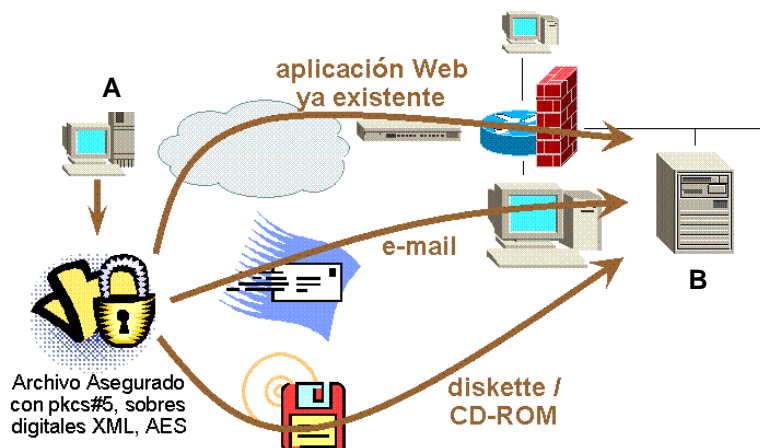
CryptoVault es una aplicación desarrollada por los ingenieros de CyberTech, que implementa aseguramiento de archivos utilizando la tecnología criptográfica más fuerte existente hoy en día comercialmente.

El motor de CryptoVault consiste de aproximadamente 9,000 líneas de código, desarrollado en su totalidad en lenguaje Java, siendo por lo tanto completa y transparentemente portable a una gran cantidad de plataformas de cómputo (IBM, Sun, Windows, etc.).

CryptoVault fue especialmente diseñada para incluir todos los elementos de ingeniería criptográfica, que garantizan fortaleza y excelente desempeño, mediante una implementación extraordinariamente eficiente de sobres digitales XML, como se muestra en la figura de abajo.



Nótese que gracias a esta protección criptográfica, el archivo cifrado y firmado por CryptoVault en el sobre digital puede ser transmitido —en el eventual fallo de los medios de transmisión— por diferentes medios como por ejemplo e-mail, o aun incluso usando diskettes o CD-ROMs, como se muestra en la siguiente figura.



CryptoVault cuenta con cuatro variedades o ‘sabores’ pensados para facilitar su adaptación e integración en el ambiente en el que vaya a ser usada. Estas variedades comparten el núcleo funcional de procesamiento (motor), siendo únicamente diferente la forma de uso e invocación de las operaciones ofrecidas por dicho motor.

En particular, las variedades son las siguientes:

- **CryptoVault CLI:** *Con Interfaz de línea de comandos:* Esta variedad permite invocar a CryptoVault desde la línea de comandos (ventana de *shell* o terminal). Las diferentes operaciones se indican por medio de modificadores y argumentos de línea de comandos.

Este ‘sabor’ es particularmente útil cuando se requiere usar la funcionalidad de CryptoVault desde procesos automáticos ya que no requiere interacción con el usuario, generalmente por medio de invocaciones al shell, scripts o archivos por lotes. También es posible usarla desde aplicaciones legacy por medio de llamadas a través del sistema operativo.

- **CryptoVault GUI:** *Con Interfaz Gráfica:* Esta variedad permite usar CryptoVault interactivamente por medio de una interfaz gráfica limpia e intuitiva que guía al usuario por medio de asistentes para que use la funcionalidad expuesta por la aplicación. Naturalmente, esta pensada para ser usada cuando los procesos se van a llevar a cabo de forma manual por un usuario (operador) que seleccione interactivamente los parámetros de las operaciones a ejecutar.
- **CryptoVault API:** *DLL para Win32:* Esta variedad permite invocar a CryptoVault directamente desde lenguajes de programación para sistemas operativos Win32 (por ejemplo, desde: Visual Basic, C++ y C#). Lo anterior permite una integración sencilla a nivel de lenguaje de programación con aquellos desarrollos para este sistema operativo que requieran hacer uso de la funcionalidad ofrecida por CryptoVault.
- **CryptoVault JAR:** *Librería JAR para aplicaciones desarrolladas en Java:* Esta variedad permite invocar a CryptoVault directamente desde aplicaciones desarrolladas en Java (en cualquier sistema operativo) por medio de clases, interfaces y métodos Java.

Mejoras con respecto a la versión 1.7

Dentro de las mejoras que se incluyeron en la versión 1.8.0 de Crypto Vault se cuentan las siguientes:

- Se adicionaron nuevas funcionalidades, con el objeto de soportar estándares criptográficos como PGP, CMS y SMIME.

Instalando CryptoVault.

Antes de instalar CryptoVault: Requerimientos.

Antes de instalar CryptoVault, es necesario asegurarse que la máquina en la que planea llevarse a cabo la instalación cumpla los siguientes requerimientos mínimos:

Para sistemas operativos **Win32**:

- Aunque es posible instalar CryptoVault sobre Windows 98 y ME, se recomienda usar sistemas operativos iguales o posteriores a Windows NT y Windows2000
- 64 Mbytes de memoria RAM,
- 60 MB de espacio libre en disco.

Para **otros** sistemas operativos:

- El sistema operativo particular debe tener por lo menos un ambiente de ejecución de Java (JRE) instalado. La versión mínima soportada es la 1.5.0.
- 64 Mbytes de memoria RAM,
- 8 MB de espacio libre en disco.

Instalación de la aplicación en ambientes Windows.

1. Dependiendo del 'sabor' de CryptoVault que haya decidido instalar, ejecute el archivo `instalador_cryptovault_<versión>_<sabor>.exe` y siga las instrucciones del asistente.

Recuerde que los archivos de instalación se encuentran en la carpeta "instaladores" del CD de instalación de CryptoVault.

2. Si ya tiene un almacén de llaves para ser usado con CryptoVault, deposítelo en el directorio "keystores". De lo contrario, será necesario crear un nuevo almacén de llaves siguiendo las instrucciones presentadas en el **capítulo 8, "Almacenes de Llaves compatibles con CryptoVault."**
3. Una vez haya generado su par de llaves, obtenga el archivo de lista de revocaciones (CRL) actualizado de su Autoridad Certificadora, y deposítelo en el directorio "crl" de la carpeta de

instalación de CryptoVault. Tenga en cuenta que el archivo debe llamarse "ca.crl" y estar en formato X.509 codificado en DER.

Recuerde además que este archivo debe actualizarse periódicamente como se explica en la sección "**Actualización de la CRL**" del **capítulo 5, "Procedimientos Administrativos relacionados con CryptoVault."**

4. Para usar la aplicación, siga las instrucciones que se encuentran en el capítulo 4 del presente documento.

Instalación de la aplicación en otros sistemas operativos ('sabor' CLI).

Es posible instalar CryptoVault en sistemas operativos como Linux, Solaris, BSD, OS/400, etc., siempre y cuando se satisfagan los requerimientos planteados al comienzo de este capítulo.

Para instalar el 'sabor' CLI de CryptoVault en cualquiera de estos sistemas operativos es necesario conocer la ubicación del ambiente de ejecución de Java (JRE), que se denominará \$JRE a lo largo de estas instrucciones.

El archivo de instalación "cryptovault_cli_all.zip" (en la carpeta "instaladores" del CD de instalación de CryptoVault) contiene todos los archivos y directorios necesarios para la instalación de Crypto Vault 1.6. Los contenidos de éste archivo se describen a continuación:

cryptovault_cli_all

CryptoVaultCLI

Archivos y librerías de la aplicación.

Policy Files

Archivos de políticas de seguridad de Java.

bcprov-jdk15-145.jar

Proveedor BouncyCastle 1.45 para JRE 1.5.

1. Busque la carpeta correspondiente al fabricante de su ambiente de ejecución de Java en el directorio "Policy Files". Luego ubique el subdirectorio correspondiente a la versión de dicho ambiente. Esta carpeta contendrá dos archivos: "local_policy.jar" y "US_export_policy.jar", que deberá copiar a: "\$JRE/lib/security/".

Si es necesario, reemplace los archivos si éstos ya existen.

Nota. Si su máquina virtual de Java fue desarrollada por un fabricante diferente a IBM o SUN, o no existe un subdirectorio con la versión específica en el instalador, será necesario obtener "local_policy.jar" y "US_export_policy.jar" directamente del fabricante particular de su JVM.

2. Abra en el editor de texto de su preferencia el archivo \$JRE/lib/security/java.security y busque la sección "List of providers and their preference orders". Debe aparecer una sección como la siguiente:

#

```
# List of providers and their preference orders (see above):

#

security.provider.1=sun.security.provider.Sun

security.provider.2=com.sun.net.ssl.internal.ssl.Provider

security.provider.3=com.sun.rsa.jca.Provider

security.provider.4=com.sun.crypto.provider.SunJCE

security.provider.5=sun.security.jgss.SunProvider
```

Luego del último proveedor, en este caso: `security.provider.5=sun.security.jgss.SunProvider`, agregue la siguiente línea:

```
security.provider.6=org.bouncycastle.jce.provider.BouncyCastleProvider
```

Note que si tiene más o menos proveedores criptográficos, el número que precede el signo '=' (6) debe cambiar. Al final debe quedar en el archivo la lista de todos los proveedores en orden numérico ascendente (finalizando con el recién insertado).

Guarde y cierre \$JRE/lib/security/java.security.

3. Copie el archivo "bcprov-jdk15-145.jar" al directorio "\$JRE/lib/ext/".
4. Copie la carpeta `CryptoVaultCLI` en el sistema de archivos, en el directorio en el que desee instalar la aplicación.
5. Si ya tiene un almacén de llaves para ser usado con `CryptoVault`, deposítelo en el directorio "keystores". De lo contrario, será necesario crear un nuevo almacén de llaves siguiendo las instrucciones presentadas en el **capítulo 8, "Almacenes de Llaves compatibles con `CryptoVault`."**
6. Una vez haya generado su par de llaves, obtenga el archivo de lista de revocaciones (CRL) actualizado de su Autoridad Certificadora, y deposítelo en el directorio "crl" de la carpeta de instalación de `CryptoVault`. Tenga en cuenta que el archivo debe llamarse "ca.crl" y estar en formato X.509 codificado en DER.

Recuerde además que este archivo debe actualizarse periódicamente como se explica en la sección "**Actualización de la CRL**" del **capítulo 5, "Procedimientos Administrativos relacionados con `CryptoVault`."**

7. Para usar la aplicación, siga las instrucciones que se encuentran a continuación en el capítulo 4 del presente documento.

Usando CryptoVault.

Uso de la variedad CLI.

Para ejecutar la aplicación en modo línea de comandos, es necesario abrir una ventana de terminal (*shell*, o ventana de comandos) y ubicarse en el directorio de instalación de la aplicación. Desde allí, dependiendo del sistema operativo, se puede ejecutar la aplicación de la siguiente manera:

a. En Windows.

```
CryptoVault.exe <opciones>
```

b. En otros sistemas operativos

```
java -jar CryptoVault-1.8CLI.jar <opciones>
```

Las opciones dependen de la tarea que se desee ejecutar, como se vera a continuación:

a. Configurando el almacén de llaves.

La ubicación del almacén de llaves se indica actualizando el valor de la entrada 'keystore' en el archivo 'keystore.cfg', que se encuentra en la carpeta de instalación de CryptoVault. Dicho archivo puede ser como el que se muestra a continuación:

```
# CryptoVault - Cybertech de Colombia Ltda. 2004
# keystore.cfg : Este archivo contiene la ubicación y contraseña
#                del almacén de llaves del usuario.
#-----

#-----
# Ubicación del almacén de llaves
#-----

keystore=keystores/user.p12

#-----
#Contraseña del almacén de llaves
#-----

password=changeit
```

```
#-----
#Alias de las llaves (OPCIONAL)
#-----

alias=user
```

La entrada “keystore” corresponde a la ubicación del almacén de llaves seleccionado, la entrada “password” corresponde a la contraseña que protege dicho almacén de llaves. Y la entrada “alias”, que es opcional, corresponde al alias dentro del almacén de llaves, en el cual está guardada la clave privada. Para más información de cómo generar almacenes compatibles con Crypto Vault refiérase al **capítulo 8 “Almacenes de Llaves compatibles con CryptoVault.”**.

b. Adicionando un nuevo destinatario.

CryptoVault acepta certificados digitales en formato X.509 Versión 3, codificados en formato DER-binario o DER-Base64.

Para agregar un destinatario, debe tener su certificado digital, expedido por la misma Autoridad Certificadora que expidió el suyo. En caso de que no lo sea, hay que agregar primero el certificado de la Autoridad Certificadora que expidió el certificado del destinatario al almacén de llaves. Para esto refiérase al **capítulo 8 “Almacenes de Llaves compatibles con CryptoVault.”**.

Para agregar un destinatario, debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -a <alias> <arch certificado>
```

Por ejemplo:

```
java -jar CryptoVault-1.8CLI.jar -a Juan_Perez perez.cer
```

Agregaré el usuario "Juan_Perez" utilizando el certificado almacenado en el archivo "perez.cer".

c. Eliminando un destinatario existente.

Para eliminar un destinatario, debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -e <alias>
```

Por ejemplo:

```
java -jar CryptoVault-1.8CLI.jar -e Juan_Perez
```

Eliminaré el destinatario "Juan_Perez" de la aplicación.

d. Cifrando un archivo.

Esta funcionalidad se puede ejecutar en los siguientes estándares: CryptoVault, SMIME, PGP y CMS.

Para cifrar un archivo debe ejecutar la aplicación referenciando el estándar que se quiere utilizar:

Para el caso de CryptoVault, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -c <archivo fuente>
<archivo destino> <destinatario>
```

Para el caso de PGP, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -cpgp <archivo fuente>
<archivo destino> <destinatario>
```

Para el caso de SMIME, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -csmime <archivo fuente>
<archivo destino> <destinatario>
```

Para el caso de CMS, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -ccms <archivo fuente>
<archivo destino> <destinatario>
```

Por ejemplo:

```
java -jar CryptoVault-1.8CLI.jar -c /documento.txt
/documento.txt.env Receptor
```

Cifrá el archivo de nombre "/documento.txt" y lo guardará en un sobre digital llamado "/documento.txt.env" dirigido al destinatario "Receptor"

e. Descifrando un archivo.

Esta funcionalidad se puede ejecutar en los siguientes estándares: CryptoVault, SMIME, PGP y CMS.

Para descifrar un archivo debe ejecutar la aplicación referenciando el estándar que se quiere utilizar:

Para el caso de CryptoVault, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -d <archivo fuente>
<archivo destino>
```

Para el caso de PGP, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -dpgp <archivo fuente>
<archivo destino>
```

Para el caso de SMIME, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -dsmime <archivo fuente>
<archivo destino>
```

Para el caso de CMS, se debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -dcms <archivo fuente>
<archivo destino>
```

Por ejemplo:

```
java -jar CryptoVault-1.8CLI.jar -d /documento.txt.env
/documento.txt
```

Descifrará el sobre digital contenido en "/documento.txt.env" y guardará el contenido en el archivo "/documento.txt"

f. Descifrando un archivo y verificando el remitente.

Para descifrar un archivo y verificar el remitente del mismo debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -dv <archivo fuente>
<archivo destino> <alias del remitente>
```

Por ejemplo:

```
java -jar CryptoVault-1.8CLI.jar -dv /documento.txt.env
/documento.txt Remitente
```

Descifrará el sobre digital contenido en "/documento.txt.env" y guardará el contenido en el archivo "/documento.txt". Además verificará que el sobre haya sido firmado por el remitente "Remitente". En CryptoVault, los destinatarios registrados se consideran también potenciales remitentes.

g. Listando los destinatarios de la aplicación.

Para listar los destinatarios disponibles, debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -l
```

h. Cambiando la contraseña del almacén de llaves.

Para cambiar la contraseña del almacén de llaves, debe ejecutar la aplicación de la siguiente manera:

```
java -jar CryptoVault-1.8CLI.jar -p
```

A continuación, la aplicación le solicitará confirmar la contraseña actual y luego le pedirá que ingrese la nueva contraseña (con su respectiva entrada de verificación).

i. Referencia rápida.

Modificador	Parámetros	Acción
-a	<alias> <arch certificado>	Agrega un destinatario.
-c	<arch fuente> <arch destino> <destinatario>	Cifra un archivo y crea un sobre digital CryptoVault.
-cpgp	<arch fuente> <arch destino> <destinatario>	Cifra un archivo y crea un sobre digital PGP.
-ccms	<arch fuente> <arch destino> <destinatario>	Cifra un archivo y crea un sobre digital CMS.
-csmime	<arch fuente> <arch destino> <destinatario>	Cifra un archivo y crea un sobre digital SMIME.
-d	<arch fuente> <arch destino>	Descifra el archivo contenido en un sobre digital CryptoVault.
-dpgp	<arch fuente> <arch destino>	Descifra el archivo contenido en un sobre digital PGP.
-dcms	<arch fuente> <arch destino>	Descifra el archivo contenido en un sobre digital CMS.
-dsmime	<arch fuente> <arch destino>	Descifra el archivo contenido en un sobre digital SMIME.
-dv	<arch fuente> <arch destino> <remitente>	Descifra el archivo contenido en un sobre digital y verifica que el remitente del sobre sea "remitente".
-e	<alias>	Elimina un destinatario.
-l		Lista los destinatarios instalados.
-p		Cambia el <i>password</i> del almacén de llaves (de forma interactiva)

Uso de la variedad API.

Una vez se ha instalado la aplicación, la carpeta de instalación contiene una subcarpeta “api” que contiene los siguientes archivos y subcarpetas:

api	
examples.....	Ejemplos de uso del API
CPlusPlusCryptoVaultDLLTester.....	Uso del API desde Microsoft Visual C++ 6.0
CSharpCryptoVaultDLLTester.....	Uso del API desde Microsoft C# .NET
VBasicCryptoVaultDLLTester.....	Uso del API desde Microsoft Visual Basic .NET
temp.....	Archivos de prueba usados por los ejemplos
CryptoVaultDLL.h.....	Declaración y especificación de las funciones del API
CryptoVaultDLL.lib.....	Archivo de librería de CryptoVault
CryptoVaultDLL.dll.....	Librería de Funciones de CryptoVault
pthreadVC2.dll.....	Librería POSIX para manejo de threads.
pthreadVC2.lib	Archivo de librería POSIX para manejo de threads.

La especificación y uso de las funciones ofrecidas por el API de Crypto Vault se describe a continuación y en el archivo de declaración “CryptoVaultDLL.h”.

NOTA:

- La DLL de CryptoVault también soporta llamados concurrentes desde varios hilos de ejecución.
- Para un correcto funcionamiento de la DLL de CryptoVault, bien sea que se use en ambientes concurrentes o no, es necesario dejar la DLL pthreadVC2.dll en la misma ubicación que la DLL de CryptoVault.

a. Configurando el almacén de llaves.

La ubicación del almacén de llaves se indica actualizando el valor de la entrada ‘keystore’ en el archivo ‘keystore.cfg’, que se encuentra en la carpeta de instalación de CryptoVault. Dicho archivo puede ser como el que se muestra a continuación:


```
# CryptoVault - Cybertech de Colombia Ltda. 2004
# keystore.cfg : Este archivo contiene la ubicación y contraseña
#                 del almacén de llaves del usuario.
#-----

#-----
# Ubicación del almacén de llaves
#-----

keystore=keystores/user.p12

#-----
#Contraseña del almacén de llaves
#-----

password=changeit

#-----
#Alias de las llaves (OPCIONAL)
#-----

alias=user
```

La entrada “keystore” corresponde a la ubicación del almacén de llaves seleccionado, la entrada “password” corresponde a la contraseña que protege dicho almacén de llaves. Y la entrada “alias”, que es opcional, corresponde al alias dentro del almacén de llaves, en el cual está guardada la clave privada. Para más información de cómo generar almacenes compatibles con Crypto Vault refiérase al **capítulo 8 “Almacenes de Llaves compatibles con CryptoVault.”**

b. Iniciando la conexión nativa.

Antes de usar cualquier función del API se recomienda invocar la función “**inicializarEnlace**” para inicializar el ambiente de ejecución de Crypto Vault.

Prototipo de la función:

```
LPSTR inicializarEnlace()
```

Especificación de la función:

Inicializa la conexión con el motor de CryptoVault instalado en la máquina local. Es deseable que se invoque una única vez antes que cualquier otra función del API.

return Cadena de Resultado con formato:

```
<Código de Resultado><Espacio>
[Mensaje de Resultado (opcional)]
```

Si la inicialización del ambiente de ejecución fue exitosa, el código de resultado es 200. De lo contrario, se retorna el código y mensaje del error específico (Ver la sección de códigos de retorno en la documentación de CryptoVault).

c. Adicionando un nuevo destinatario.

CryptoVault acepta certificados digitales en formato X.509 Versión 3, codificados en formato DER-binario o DER-Base64.

Para agregar un destinatario, debe tener su certificado digital, expedido por la misma Autoridad Certificadora que expidió el suyo. En caso de que no lo sea, hay que agregar primero el certificado de la Autoridad Certificadora que expidió el certificado del destinatario al almacén de llaves. Para esto refiérase al capítulo 8 “Creando Almacenes de Llaves compatibles con Crypto Vault”.

Para adicionar un nuevo destinatario se usa la función “**adicionarDestinatario**”.

Prototipo de la función:

```
LPSTR adicionarDestinatario (
    LPSTR pathCertificado,
    LPSTR aliasDestinatario
)
```

Especificación de la función:

Adiciona un nuevo destinatario a la lista de destinatarios de Crypto Vault

param <pathCertificado> [IN]
URI (incluyendo el nombre) del archivo que contiene el certificado digital X.509 del nuevo destinatario

param <aliasDestinatario> [IN]
Alias con el que se identificará el nuevo destinatario.

return Cadena de Resultado con formato:

```
<Código de Resultado><Espacio>
[Mensaje de Resultado (opcional)]
```

Si fue posible agregar el nuevo destinatario, el código de resultado es 200. De lo contrario, se retorna el código y mensaje del error específico (Ver la sección de códigos de retorno en la documentación de CryptoVault).

d. Eliminando un destinatario existente.

Para eliminar un destinatario se usa la función “**removeDestinatario**”.

Prototipo de la función:

```
LPSTR removeDestinatario (
    LPSTR aliasDestinatario
)
```

Especificación de la función:

Remueve un destinatario existente en la lista de destinatarios de Crypto Vault

param <aliasDestinatario> [IN]
Alias del destinatario a remover

param <cadenaDeRespuesta> [OUT]
Corresponde a un doble apuntador a una cadena de caracteres (El primer apuntador no puede ser NULL).

Al finalizar el llamado, lo apuntado por el primer apuntador corresponde a un apuntador a una cadena de caracteres que contiene la descripción textual del resultado de la ejecución de la función.

return Cadena de Resultado con formato:

<Código de Resultado><Espacio>
[Mensaje de Resultado (opcional)]

Si fue posible remover el destinatario de la lista, el código de resultado es 200. De lo contrario, se retorna el código y mensaje del error específico. (Ver la sección de códigos de retorno en la documentación de CryptoVault).

e. Cifrando un archivo.

Para cifrar y firmar un archivo se usa la función “**cifrarYfirmar**”.

Prototipo de la función:

```
LPSTR cifrarYfirmar (  
    LPSTR pathOrigen,  
    LPSTR pathDestino,  
    LPSTR aliasDestinatario  
)
```

Especificación de la función:

Genera un sobre digital que incluye un archivo que se cifra y se firma digitalmente

param <pathOrigen> [IN]
URI (incluyendo el nombre) del archivo a cifrar

param <pathDestino> [IN]

URI (incluyendo el nombre) del archivo en el cual se colocara sobre digital que contiene el archivo cifrado y firmado. Si ya existe, se reemplaza.

param <aliasDestinatario> [IN]
Nombre del destinatario del sobre digital.

return Cadena de Resultado con formato:

<Código de Resultado><Espacio>
[Mensaje de Resultado (opcional)]

Si el sobre se generó correctamente, el código de resultado es 200. De lo contrario, se retorna el código y mensaje del error específico (Ver la sección de códigos de retorno en la documentación de CryptoVault).

f. Descifrando un archivo.

Para descifrar y verificar la firma digital de un archivo se usa la función “**descifrarYverificarFirma**”.

Prototipo de la función:

```
LPSTR descifrarYverificarFirma(  
    LPSTR pathOrigen,  
    LPSTR pathDestino  
)
```

Especificación de la función:

Abre un sobre digital, descifrando su contenido y verificando la firma digital del mismo.

param <pathOrigen> [IN]
URI (incluyendo el nombre) del archivo que contiene el sobre digital

param <pathDestino> [IN]
URI (incluyendo el nombre) del archivo en el cual se colocara el contenido del sobre digital una vez se descifre.

return Cadena de Resultado con formato:

<Código de Resultado><Espacio>
[Mensaje de Resultado (opcional)]

Si fue posible acceder el contenido del sobre validando su autenticidad e integridad, el código de resultado es 200. De lo contrario, se retorna el código y mensaje del error específico (Ver la sección de códigos de retorno en la documentación de CryptoVault).

g. Descifrando un archivo y verificando el remitente.

Para descifrar, verificar la firma digital de un archivo y verificar el remitente del sobre digital se usa la función “**descifrarYverificarFirmaYRemitente**”.

Prototipo de la función:

```
LPSTR descifrarYverificarFirmaYRemitente(  
    LPSTR pathOrigen,  
    LPSTR pathDestino,  
    LPSTR remitente  
    )
```

Especificación de la función:

Abre un sobre digital, descifrando su contenido y verificando la firma digital del mismo. Además verifica que quién haya enviado el sobre sea el remitente dado como parámetro.

param <pathOrigen> [IN]
URI (incluyendo el nombre) del archivo que contiene el sobre digital

param <pathDestino> [IN]
URI (incluyendo el nombre) del archivo en el cual se colocara el contenido del sobre digital una vez se descifre.

param <remitente> [IN]
Alias del remitente que se desea verificar. Los posibles valores de este parámetro corresponden a los alias asignados a los destinatarios CryptoVault. En CryptoVault, los destinatarios se consideran también potenciales remitentes.

return Cadena de Resultado con formato:

```
<Código de Resultado><Espacio>  
[Mensaje de Resultado (opcional)]
```

Si fue posible acceder el contenido del sobre validando su autenticidad, integridad y remitente, el código de resultado es 200. De lo contrario, se retorna el código y mensaje del error específico (Ver la sección de códigos de retorno en la documentación de CryptoVault).

h. Obteniendo la lista de destinatarios de la aplicación

Para listar los alias de los destinatarios actuales, se usa la función “**listarDestinatarios**”.

Prototipo de la función:

```
listarDestinatarios()
```

Especificación de la función:

Retorna la lista de nombres de destinatarios actuales.

return Cadena de Resultado y listado de destinatarios. En caso que sea posible acceder la lista de destinatarios, se retorna una cadena con el siguiente formato:

```
200<Espacio><Número de destinatarios><\n (salto de
línea)>
<Alias del Destinatario1><\n (salto de línea)>
<Alias del Destinatario2><\n (salto de línea)>
...
<Alias del DestinatarioN><\n (salto de línea)>
```

En caso que no sea posible retornar el listado de destinatarios, se retorna una cadena de Resultado con formato:

```
<Código de Error><Espacio>
[Mensaje de Error (opcional)]
```

(Ver la sección de códigos de retorno en la documentación de CryptoVault).

i. Cambiando la contraseña del almacén de llaves.

Para cambiar la contraseña de la llave privada se usa la función “**cambiarPasswordLlavePrivada**”.

Prototipo de la función:

```
LPSTR cambiarPasswordLlavePrivada(
    LPSTR passwordActual,
    LPSTR passwordNuevo
);
```

Especificación de la función:

Cambia la contraseña de la llave privada del usuario

param <passwordActual> [IN/OUT]

Corresponde al password actual del almacén de llaves PKCS12 que contiene la llave privada del usuario. Se hace el mejor intento por borrar su contenido (se llena de caracteres '\0').

param <passwordNuevo> [IN/OUT]

Corresponde al password que se quiere asignar al almacén de llaves PKCS12 que contiene la llave privada del usuario. Se hace el mejor intento por borrar su contenido (se llena de caracteres '\0')

return Cadena de Resultado con formato:

```
<Código de Resultado><Espacio>  
[Mensaje de Resultado (opcional)]
```

Si fue posible cambiar la contraseña de la llave privada, el código de resultado es 200. De lo contrario, se retorna el código y mensaje del error específico. (Ver la sección de códigos de retorno en la documentación de CryptoVault).

j. Terminando la conexión nativa.

Una vez no sea necesario usar ninguna otra función del API, puede llamar la función “**terminarEnlace**” para liberar los recursos usados por el motor de Crypto Vault.

Prototipo de la función:

```
terminarEnlace()
```

Especificación de la función:

Finaliza la conexión con el motor de CryptoVault instalado en la máquina local.

Debe invocarse solamente cuando ya no se tenga planeado invocar otras funciones del API para liberar los recursos asignados al enlace.

Una vez se ejecute, cualquier llamado al API desde el mismo proceso fallará.

Nota importante. Una vez se invoque la función ‘terminarEnlace()’ no será posible usar ninguna función del API de CryptoVault, *ni siquiera* volviendo a inicializar el enlace. Lo anterior aduce a que solamente es posible levantar una JVM por proceso.

Por lo anterior, se recomienda no invocar la terminación del enlace a menos que se esté completamente seguro que no se requiere hacer ningún uso posterior del API; o mejor, abstenerse de llamar dicha función y dejar que la terminación del enlace se produzca automáticamente producto de la finalización del proceso padre que alberga la aplicación cliente.

Cuando se usa en ambientes concurrentes (con varios hilos de ejecución en paralelo) es necesario realizar un join de todos los hilos antes de invocar la función ‘terminarEnlace()’. Si se llama esta función y los hilos aún se siguen ejecutando o no han sido eliminados del proceso, se pueden generar excepciones. En estos casos es recomendable no llamar la función ‘terminarEnlace()’ y permitir que el proceso padre que alberga tanto los hilos de ejecución como la Java Virtual Machine JVM (que ejecuta CryptoVault) termine el enlace automáticamente consecuencia de la finalización de su ejecución.

k. Ejecutando los ejemplos de uso del API.

Como se mencionó al inicio de este capítulo, la carpeta “api” contiene una subcarpeta “examples” que a su vez contiene tres proyectos de prueba del API de Crypto Vault desde C++, C# y Visual Basic.

Para ejecutar estos ejemplos deben seguirse los siguientes pasos:

1. Instalar la aplicación y configurarla para que use el keystore “ana.p12” que se encuentra en la carpeta “examples\certicamara\keystores”. Recuerde que el password por defecto de los almacenes de llaves de prueba es: “changeit”
2. No olvidar copiar el archivo CRL de “examples\certicamara\crl” a la carpeta “crl” en el directorio de instalación de CryptoVault.
3. La carpeta “api\examples\temp” contiene 2 archivos requeridos por los proyectos de prueba. Por defecto, copiarlos a la carpeta “C:\temp”, o modificar la ruta absoluta de esta carpeta en cada uno de los proyectos (terminada en “\”), así:
 - a. CPlusPlusCryptoVaultDLLTester: En el archivo src\impl\CrptoVaultDLLTester.cpp, Variable global carpetaArchsPrueba (línea 10)
 - b. CSharpCryptoVaultDLLTester: En el archivo Class1.cs, Atributo estático privado carpetaArchsPrueba (línea 57).
 - c. VBasicCryptoVaultDLLTester: En el archivo Module1.vb, Atributo estático privado carpetaArchsPrueba (línea 37)
4. La librería CryptoVaultDLL.dll se encuentra en el directorio de ejecución de cada uno de los proyectos (el mismo donde se genera el ejecutable). Es posible sin embargo, copiarla y dejarla únicamente en el directorio de librerías del sistema v.gr. c:\Windows\System32 para poder invocarla desde cualquier proyecto que la requiera.
5. Abrir cada uno de los proyectos en el correspondiente IDE y ejecutarlos.

Uso de la variedad JVA.

Una vez se ha instalado la aplicación, la carpeta de instalación contiene una subcarpeta “jva” que contiene los siguientes archivos y subcarpetas:

jva	
examples.....	Ejemplo de uso de la interfaz para aplicaciones Java
lib.....	Librerías requeridas por la aplicación
bcprov-jdk14-128.jar.....	Librería de implementación de operaciones criptográficas
CryptoVault-1.8JVA.jar.....	Librería de CryptoVault para aplicaciones Java.

La especificación y uso de las clases, interfaces y métodos ofrecidas por la librería JAR de CryptoVault se describen a continuación.

a. Configurando el almacén de llaves.

La ubicación del almacén de llaves se indica actualizando el valor de la entrada ‘keystore’ en el archivo ‘keystore.cfg’, que se encuentra en la carpeta de instalación de CryptoVault. Dicho archivo puede ser como el que se muestra a continuación:

```
# CryptoVault - Cybertech de Colombia Ltda. 2004
# keystore.cfg : Este archivo contiene la ubicación y contraseña
#                del almacén de llaves del usuario.
#-----

#-----
# Ubicación del almacén de llaves
#-----

keystore=keystores/user.p12

#-----
#Contraseña del almacén de llaves
#-----

password=changeit

#-----
#Alias de las llaves (OPCIONAL)
#-----

alias=user
```

La entrada “keystore” corresponde a la ubicación del almacén de llaves seleccionado, la entrada “password” corresponde a la contraseña que protege dicho almacén de llaves. Y la entrada “alias”, que es opcional, corresponde al alias dentro del almacén de llaves, en el cual está guardada la clave privada. Para más información de cómo generar almacenes compatibles con Crypto Vault refiérase al **capítulo 8 “Almacenes de Llaves compatibles con CryptoVault.”**.

b. Configurando la ubicación de la aplicación.

Antes de usar la librería de CryptoVault para Java, es preciso incluirla en el *classpath* de la aplicación cliente. Asimismo, es necesario incluir la librería de implementación de operaciones criptográficas (*bcprov-jdk14-128.jar*) en dicho *classpath*.

A continuación debe usarse el método estático “**initApplicationEnvironment**” de la clase “*CryptoVaultNativeJavaFacade*” para configurar el directorio de instalación de CryptoVault. Esto con el fin que el motor de CryptoVault contenido en la librería JAR este en capacidad de ubicar los archivos requeridos por la aplicación para funcionar correctamente.

La especificación de este método aparece a continuación:

```
public static ReturnMessage  
initApplicationEnvironment(java.io.File  
applicationInstallationDirectory)
```

Initializes the application environment by setting the Application installation folder

Parameters:

applicationInstallationDirectory - Folder where CryptoVault is installed

Returns:

ReturnMessage describing the operation result

c. Obteniendo la fachada para aplicaciones Java.

Una vez se ha configurado la ubicación de la aplicación, debe obtenerse la fachada para aplicaciones Java por medio del método estático “**getInstance**” de la clase “*CryptoVaultNativeJavaFacade*”:

```
public static ICryptoVaultNativeJavaFacade getInstance() throws  
CryptoVaultException
```

Returns the unique CryptoVaultNativeJavaFacade instance

Returns:

The unique CryptoVaultNativeJavaFacade instance

Throws:

CryptoVaultException - In case the application logger cannot be initialized

d. Adicionando un nuevo destinatario.

CryptoVault acepta certificados digitales en formato X.509 Versión 3, codificados en formato DER-binario o DER-Base64.

Para agregar un destinatario, debe tener su certificado digital, expedido por la misma Autoridad Certificadora que expidió el suyo. En caso de que no lo sea, hay que agregar primero el certificado de la Autoridad Certificadora que expidió el certificado del destinatario al almacén de

llaves. Para esto refiérase al capítulo 8 “Creando Almacenes de Llaves compatibles con Crypto Vault”.

Para adicionar un nuevo destinatario, usar el método “**addRecipient**” declarado en la interfaz “ICryptoVaultNativeJavaFacade”:

```
public ReturnMessage addRecipient(java.lang.String alias,  
java.lang.String certFile)
```

Adds a new recipient

Parameters:

alias - Recipient alias

certFile - Recipient certificate filename (as a complete path)

Returns:

The operation result.

e. Eliminando un destinatario existente.

Para eliminar un destinatario existente, usar el método “**removeRecipient**” declarado en la interfaz “ICryptoVaultNativeJavaFacade”:

```
public ReturnMessage removeRecipient(java.lang.String alias)
```

Removes an existing recipient from the recipients list

Parameters:

alias - Recipient-to-be-removed alias

Returns:

The operation result.

f. Cifrando un archivo.

Para cifrar un archivo, usar el método “**encryptAndSignDocument**” declarado en la interfaz “ICryptoVaultNativeJavaFacade”:

```
public ReturnMessage encryptAndSignDocument(java.lang.String  
source, java.lang.String target, java.lang.String alias)
```

Encrypts and signs a given file generating a digital envelope with the signed content.

Parameters:

source - The source file name as an absolute path (the file to be encrypted)

target - The target file name as an absolute path (the file where the digital envelope is to be created)

alias - The digital envelope recipient alias

Returns:

The operation result.

g. Cifrando un archivo (a partir de un flujo de bytes)

Para cifrar un arreglo de bytes, usar el método “**encryptAndSignDocument**” declarado en la interfaz “**ICryptoVaultNativeJavaFacade**”:

```
public ReturnMessage encryptAndSignDocument(byte[] source,  
java.lang.String target, java.lang.String alias)
```

Encrypts and signs a given byte array generating a digital envelope with the signed content.

Parameters:

source - The byte array to be encrypted

target - The target file name as an absolute path (the file where the digital envelope is to be created)

alias - The digital envelope recipient alias

Returns:

The operation result.

h. Descifrando un archivo.

Para descifrar un archivo, usar el método “**decryptAndVerifyEnvelope**” declarado en la interfaz “**ICryptoVaultNativeJavaFacade**”:

```
public ReturnMessage decryptAndVerifyEnvelope(java.lang.String  
source, java.lang.String target)
```

Opens a digital envelope decrypting its contents and verifying its' signature

Parameters:

source - The source file name as an absolute path (the digital envelope)

target - The target file name as an absolute path (the file where the decrypted content is to be placed)

Returns:

The operation result.

i. Descifrando un archivo y verificando el remitente.

Para descifrar un archivo y verificar el remitente del sobre digital, usar el método **“decryptVerifyEnvelopeAndVerifySender”** declarado en la interfaz **“ICryptoVaultNativeJavaFacade”**:

```
public ReturnMessage  
decryptVerifyEnvelopeAndVerifySender(java.lang.String source,  
java.lang.String target, java.lang.String sender)
```

Opens a digital envelope decrypting its contents, verifying its' signature and verifying that the sender of the envelope has the same digital certificate as the one assigned to the alias "sender" in the CryptoVault registered recipients. CryptoVault considers all its recipients as potential senders.

Parameters:

source - The source file name as an absolute path (the digital envelope)

target - The target file name as an absolute path (the file where the decrypted content is to be placed)

sender – A CryptoVault sender alias. CryptoVault considers all its recipients as potential senders.

Returns:

The operation result.

j. Obteniendo la lista de destinatarios de la aplicación.

Para obtener la lista de destinatarios de la aplicación, usar el método **“listRecipientAliases”** declarado en la interfaz **“ICryptoVaultNativeJavaFacade”**:

```
public java.lang.String[] listRecipientAliases() throws  
CryptoVaultException
```

Returns the registered recipients aliases.

Returns:

The registered recipients aliases.

Throws:

CryptoVaultException - In case something goes wrong with the operation.

k. Cambiando la contraseña del almacén de llaves.

Para cambiar la contraseña del almacén de llaves, usar el método “**changePrivateKeyPassword**” declarado en la interfaz “ICryptoVaultNativeJavaFacade”:

```
public ReturnMessage changePrivateKeyPassword(char[] oldpass,  
                                                char[] newpass)
```

Changes the keystore (private key) password

Parameters:

oldpass - Old password

newpass - New password

Returns:

The operation result.

I. Ejecutando el ejemplo de uso.

Como se mencionó al inicio de este capítulo, la carpeta “jva” contiene una subcarpeta “examples” que a su vez contiene un ejemplo de uso de la librería JAR de CryptoVault. Para ejecutar este ejemplo deben seguirse los siguientes pasos:

1. Instalar la aplicación y configurarla para que use el keystore “ana.p12” que se encuentra en la carpeta “examples\certicamara\keystores”. Recuerde que el password por defecto de los almacenes de llaves de prueba es: “changeit”
2. No olvidar copiar el archivo CRL de “examples\certicamara\crl” a la carpeta “crl” en el directorio de instalación de CryptoVault.
3. Compilar el archivo “CryptoVaultClient.java”. No olvide incluir en el classpath las librerías CryptoVault-1.6JVA.jar y bcprov-jdk14-128.jar.
4. Ejecutar la clase CryptoVaultClient teniendo en cuenta:

a. Debe incluir en el classpath las librerías CryptoVault-1.6JVA.jar y bcprov-jdk14-128.jar.

b. Los argumentos esperados por la aplicación son:

args[0]=Directorio de instalación de CryptoVault

args[1]=Nombre de alias del destinatario a adicionar

args[2]=Ubicación del certificado del destinatario a adicionar

args[3]=Ubicación del archivo a cifrar

args[4]=Ubicación del archivo donde se guardara el sobre con el archivo cifrado

Uso de la variedad GUI.

La aplicación puede ejecutarse desde el acceso directo que se creó durante el proceso de instalación en el menú inicio (Programas/CryptoVault GUI/CryptoVault) o ejecutando directamente la aplicación en {Directorio de Instalación}\ Crypto Vault GUI.exe.

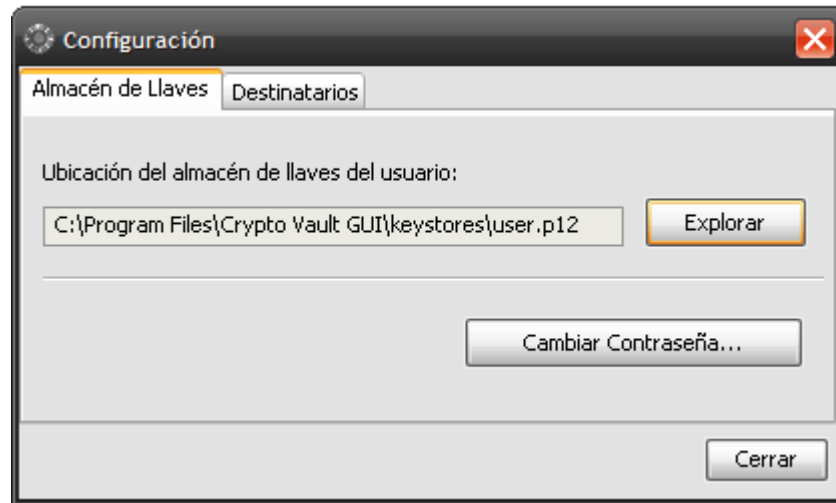
Una vez ha iniciado la ejecución de Crypto Vault se muestra la ventana principal de la aplicación, que se ilustra a continuación:



Desde esta ventana es posible cifrar un archivo, descifrar un archivo, configurar la aplicación y ver la versión “incorporada” de la ayuda, que despliega las instrucciones de uso de la aplicación haciendo uso del visor de ayuda del sistema.

a. Configurando el almacén de llaves.

1. Acceda al diálogo de configuración presionando el botón "Configurar Aplicación" que aparece en la pantalla principal de la aplicación.
2. Oprima el botón "Explorar" que aparece en la pestaña "Almacén de llaves".



3. Seleccione la ubicación del archivo que contiene el almacén de llaves.
4. CryptoVault le preguntará si está seguro de que desea cambiar el almacén de llaves. Presione el botón "Si".

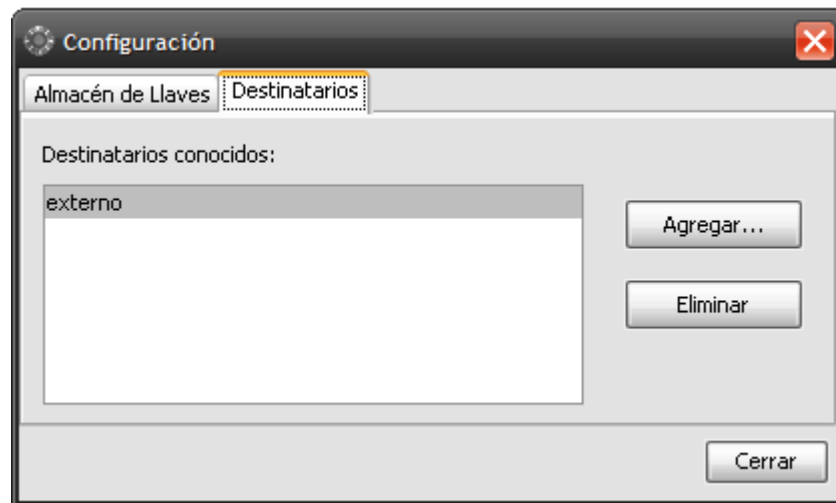
a. Adicionando un nuevo destinatario.

CryptoVault acepta certificados digitales en formato X.509 Versión 3, codificados en formato DER-binario o DER-Base64.

Para agregar un destinatario, debe tener su certificado digital, expedido por la misma Autoridad Certificadora que expidió el suyo. En caso de que no lo sea, hay que agregar primero el certificado de la Autoridad Certificadora que expidió el certificado del destinatario al almacén de llaves. Para esto refiérase al **capítulo 8 "Almacenes de Llaves compatibles con CryptoVault."**

Para agregar un destinatario, siga los siguientes pasos:

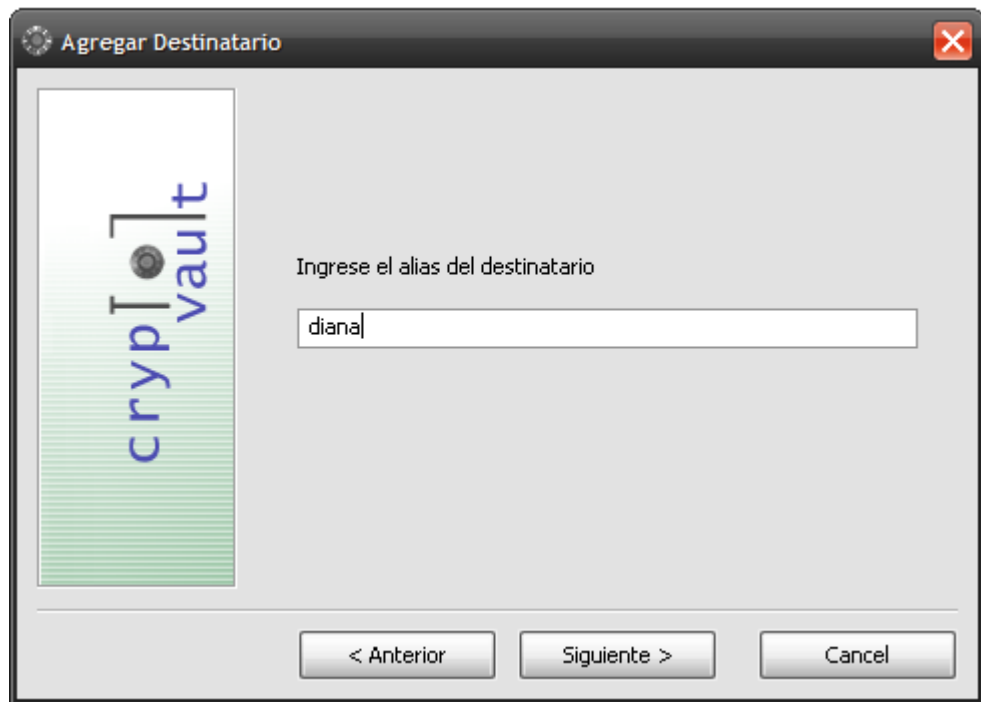
1. Acceda al diálogo de configuración presionando el botón "**Configurar Aplicación**" que aparece en la pantalla principal de la aplicación.
2. Seleccione la pestaña "Destinatarios" y a continuación oprima el botón "Agregar". Esto abrirá el ayudante para agregar un destinatario.



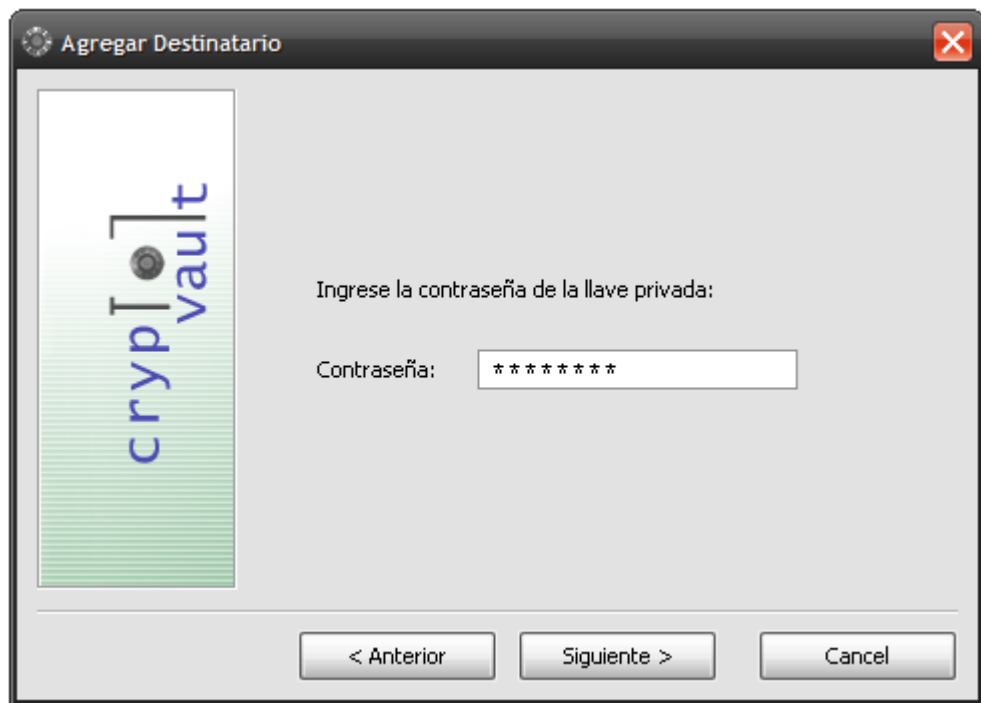
3. Oprima el botón "Explorar" y seleccione la ubicación del certificado digital del destinatario. Una vez seleccionado, oprima el botón "Siguiente"



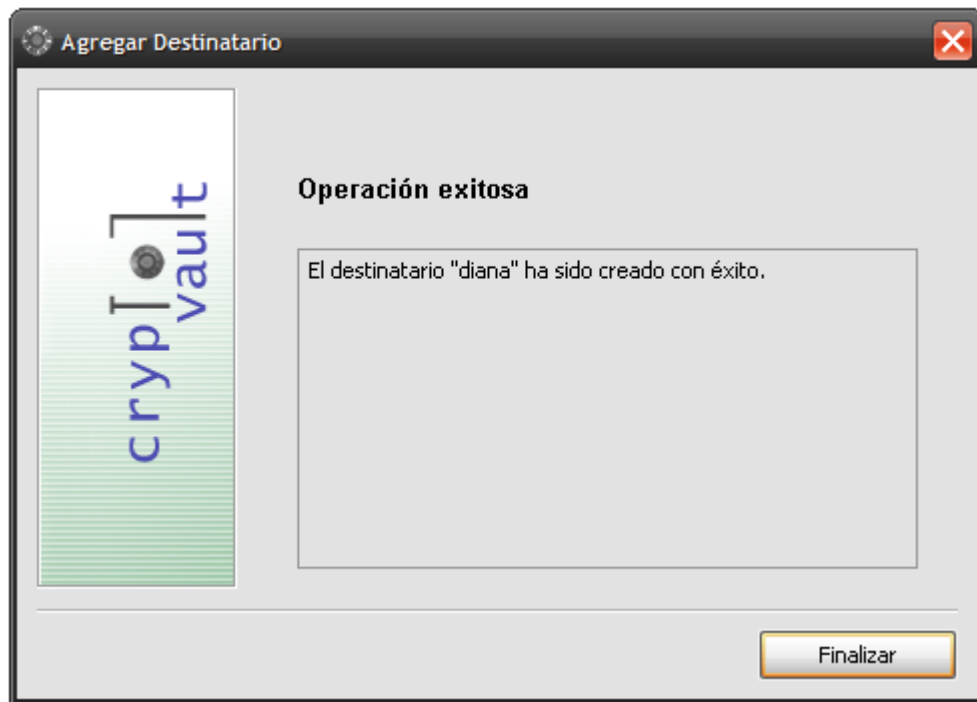
4. Escriba el nombre del destinatario que desea agregar y oprima el botón "Siguiente".



5. Escriba la contraseña de la llave privada. Una vez escrita, oprima el botón "Siguiete".

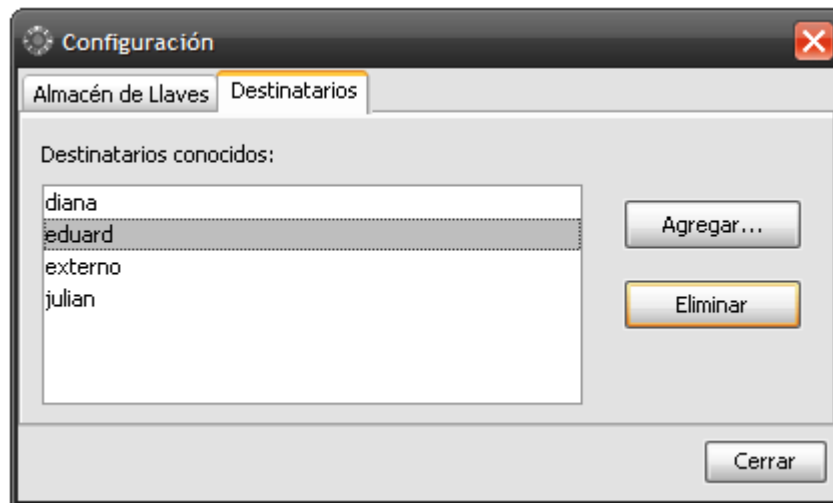


6. Aparecerá el mensaje "El destinatario ha sido agregado".



b. Eliminando un destinatario existente.

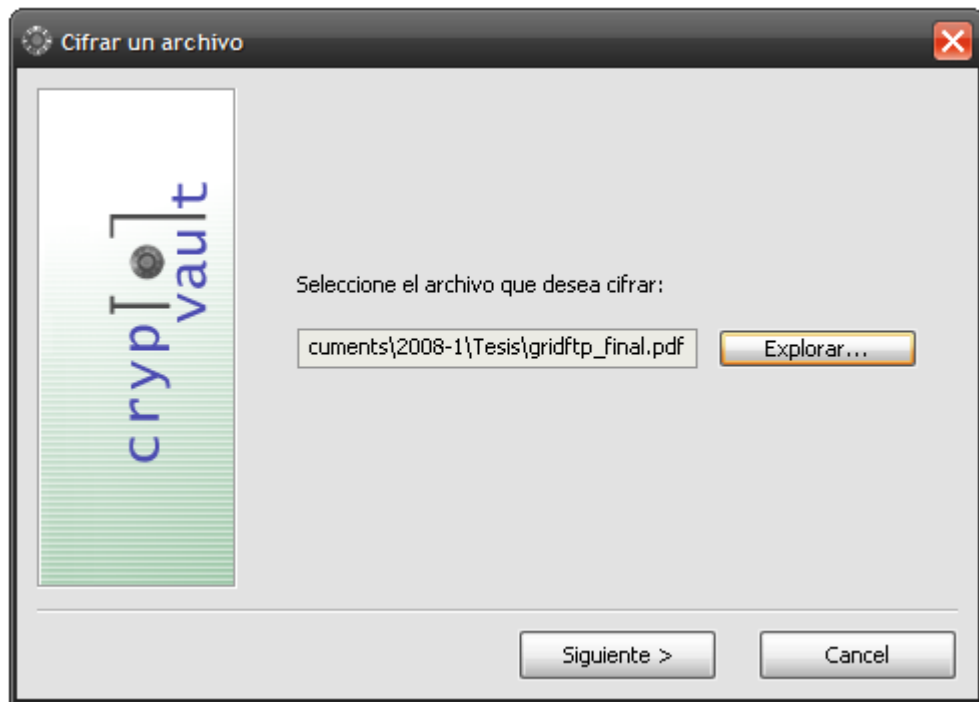
1. Acceda al diálogo de configuración presionando el botón "Configurar Aplicación" que aparece en la pantalla principal de la aplicación.
2. Seleccione la pestaña "Destinatarios" y a continuación oprima el botón "Eliminar".



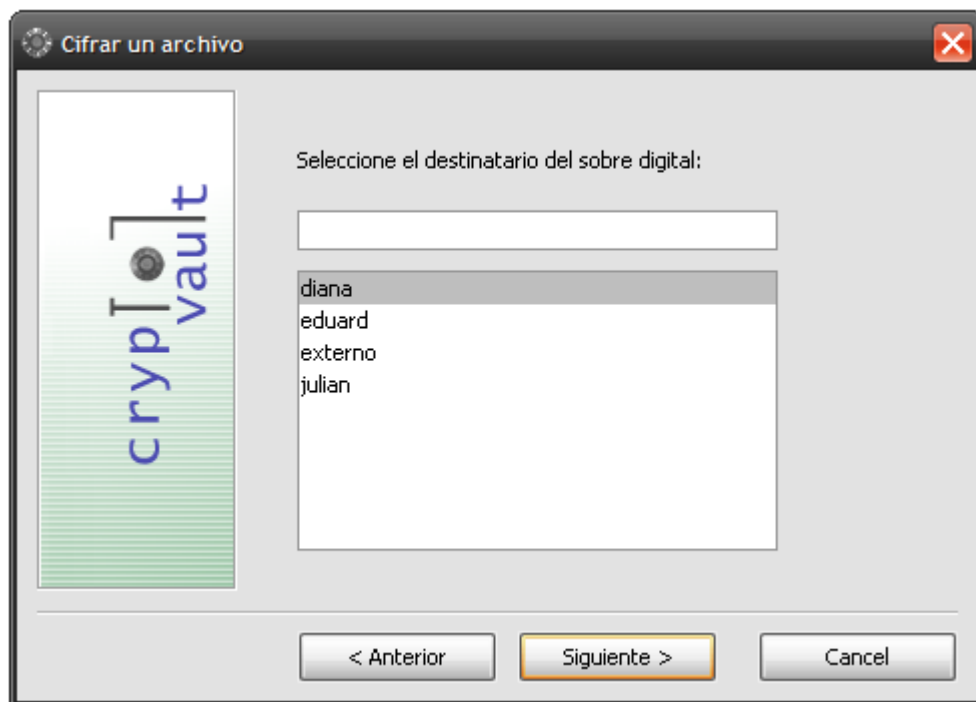
3. CryptoVault le preguntará si está seguro de que eliminar el destinatario. Presione el botón "Si".

c. Cifrando un archivo.

1. Oprima el botón "Cifrar un archivo" en la pantalla principal. Esto abrirá el ayudante para cifrar un archivo.
2. Oprima el botón "Explorar" y seleccione el archivo que desea cifrar. Una vez seleccionado, oprima el botón "Siguiente".

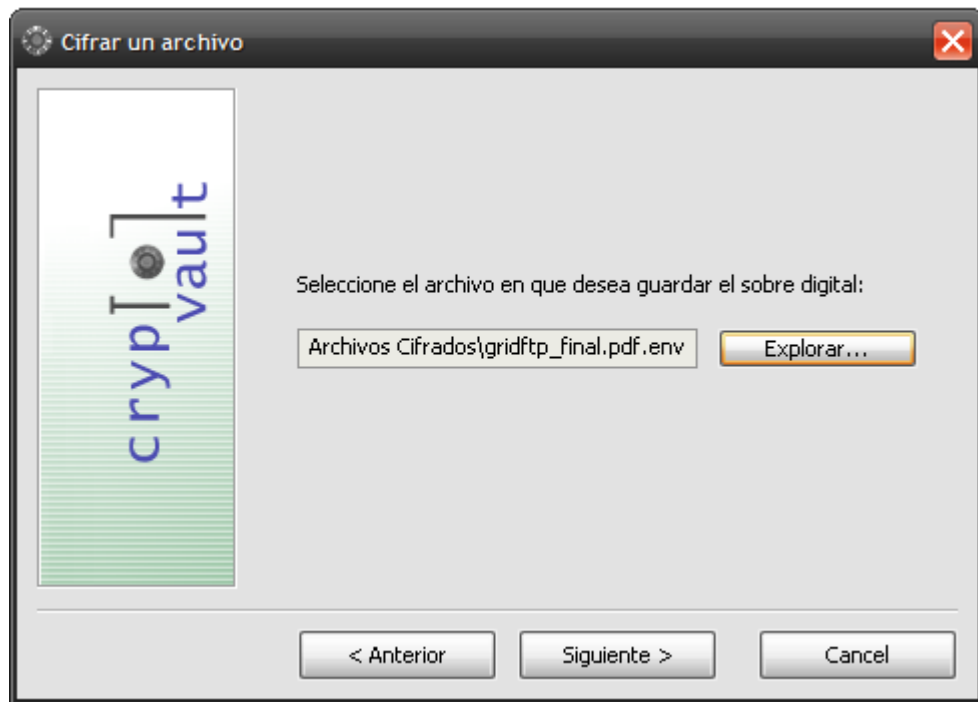


3. Escoja de la lista el destinatario del sobre digital y a continuación, oprima el botón "Siguiente".

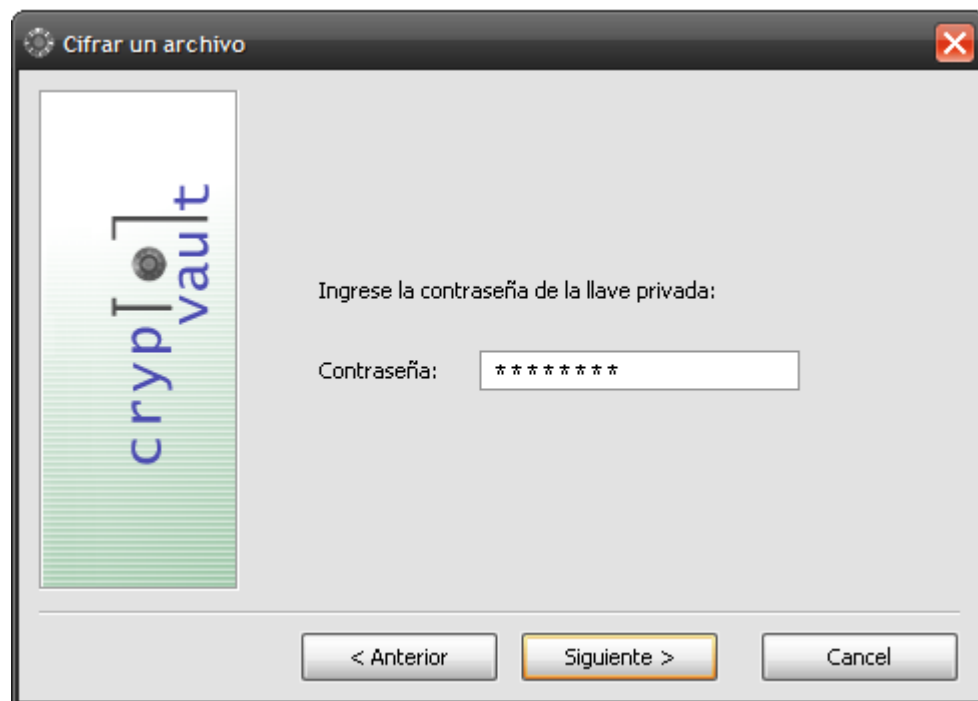


Puede escribir en la casilla de texto las primeras letras del alias del destinatario y la aplicación mostrará únicamente los destinatarios cuyos nombres comiencen por dichas letras.

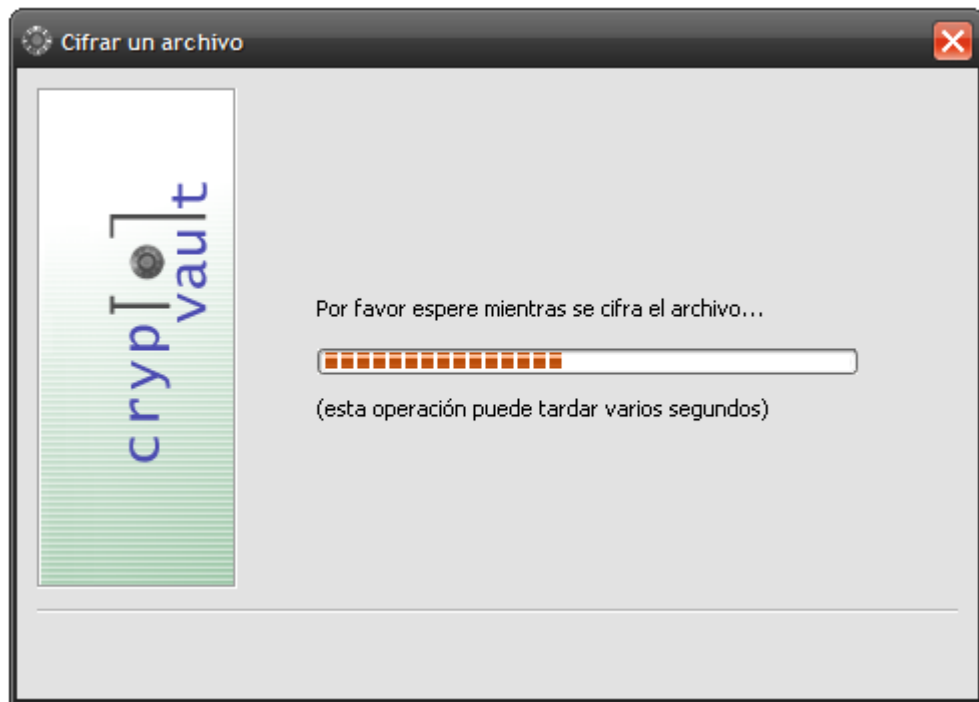
4. Escoja el archivo en que desea guardar el sobre digital. Crypto Vault le sugerirá un nombre de archivo en la carpeta "Mis Archivos Descifrados", pero si desea cambiarlo puede oprimir el botón "Explorar" y especificar otra ubicación. Cuando haya seleccionado el nombre del archivo destino, oprima el botón "Siguiente".



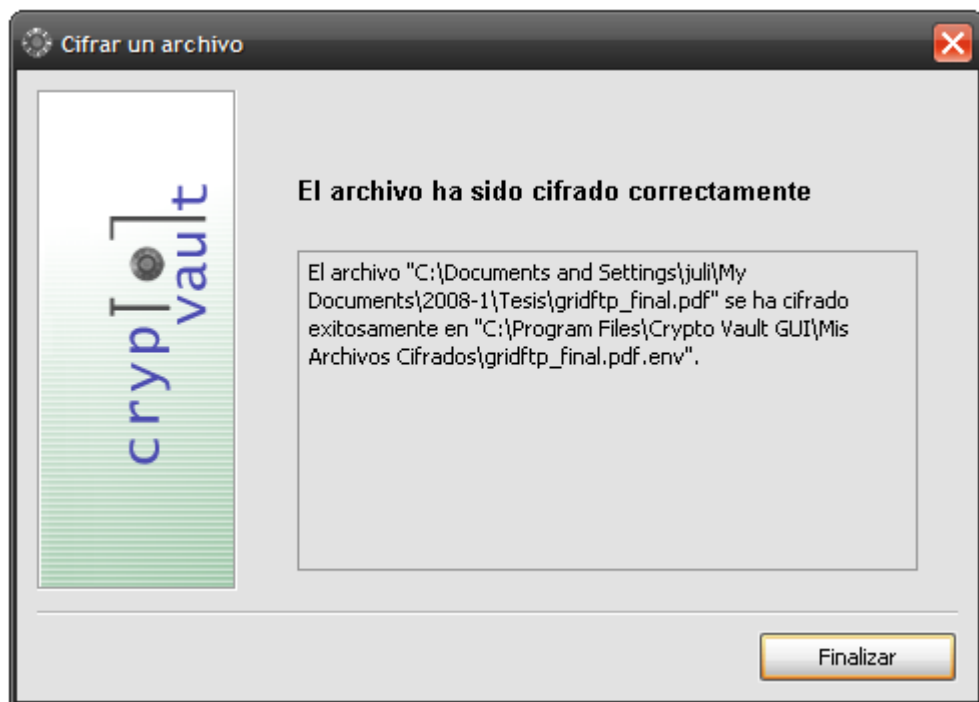
5. Escriba la contraseña de la llave privada. Una vez escrita, oprima el botón "Siguiete".



6. Crypto Vault le indicará el progreso del proceso de ciframiento del archivo.

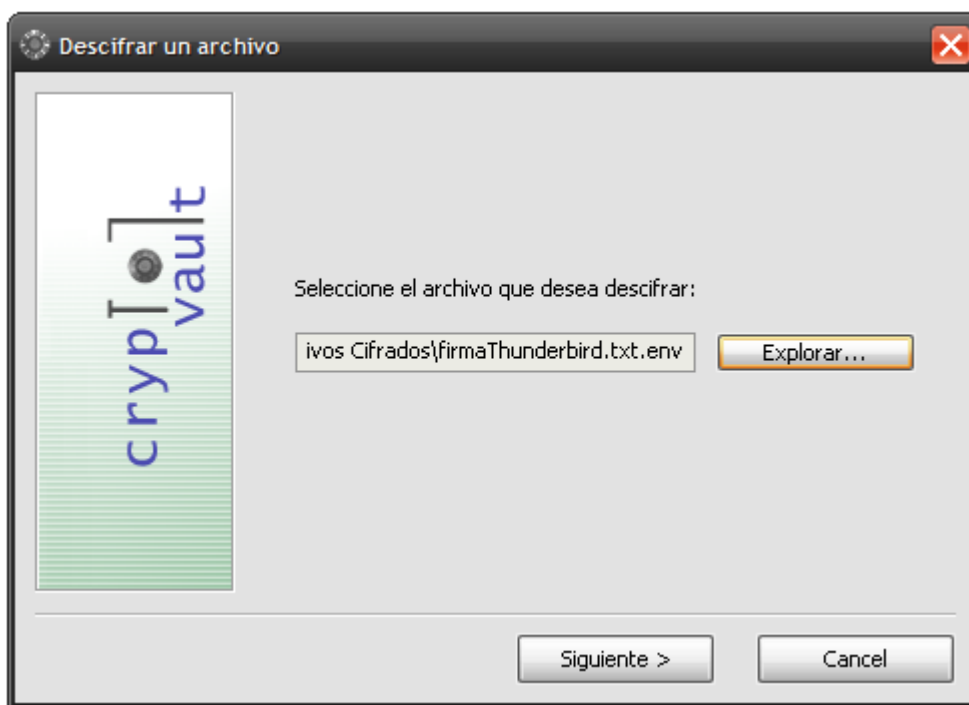


Una vez terminada la operación, aparecerá el mensaje "El archivo ha sido cifrado exitosamente".

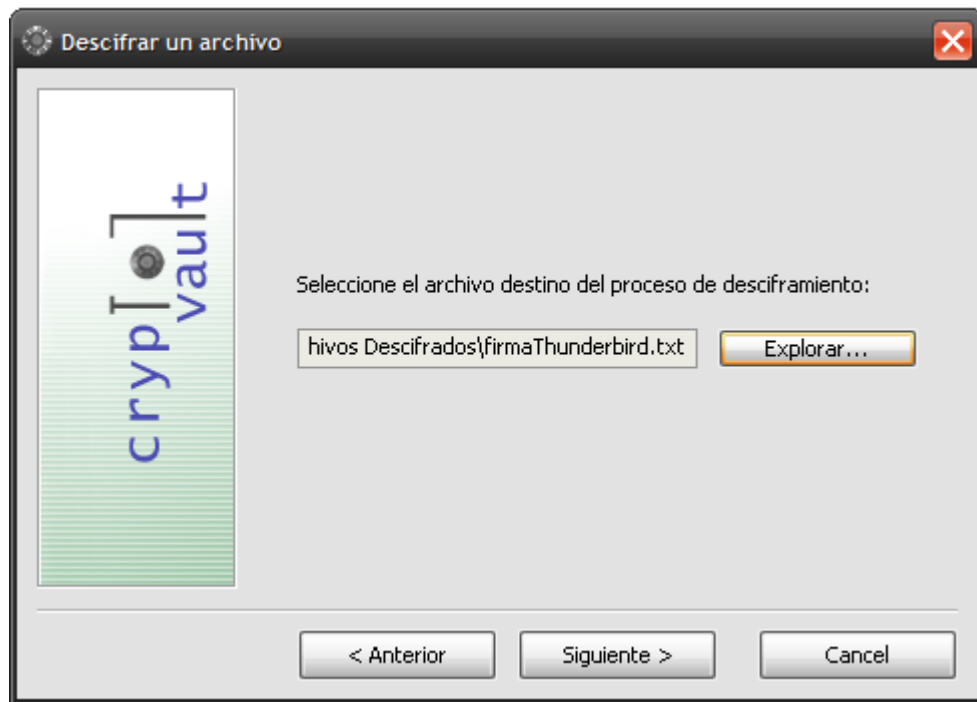


d. Descifrando un archivo.

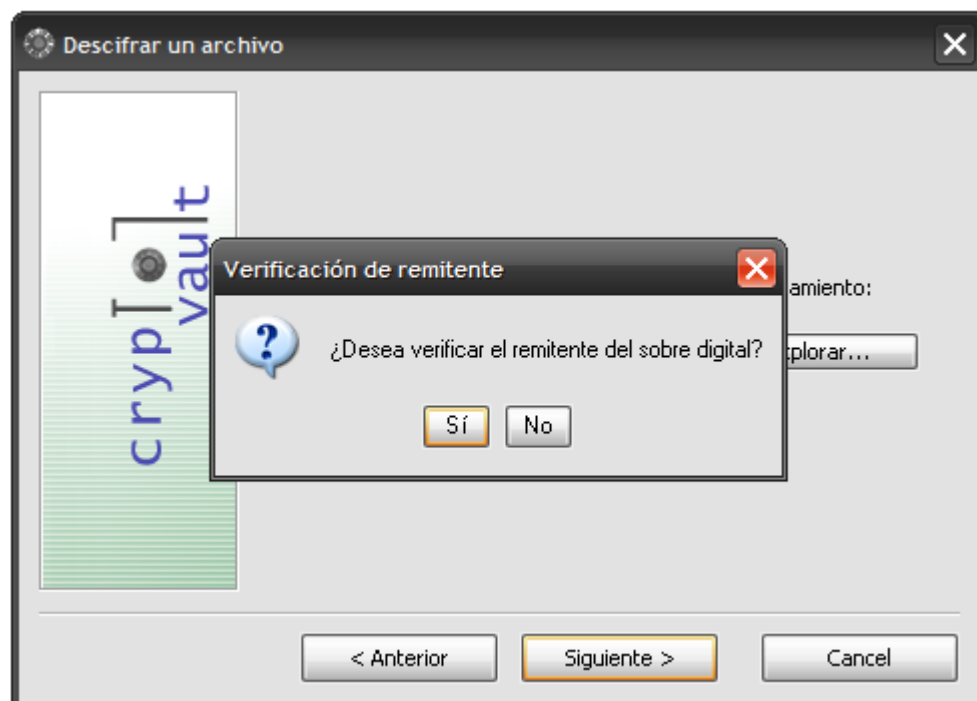
1. Oprima el botón "Descifrar un archivo" en la pantalla principal. Esto abrirá el ayudante para descifrar un archivo.
2. Oprima el botón "Explorar" y seleccione el archivo que desea descifrar. Una vez seleccionado, oprima el botón "Siguiente".



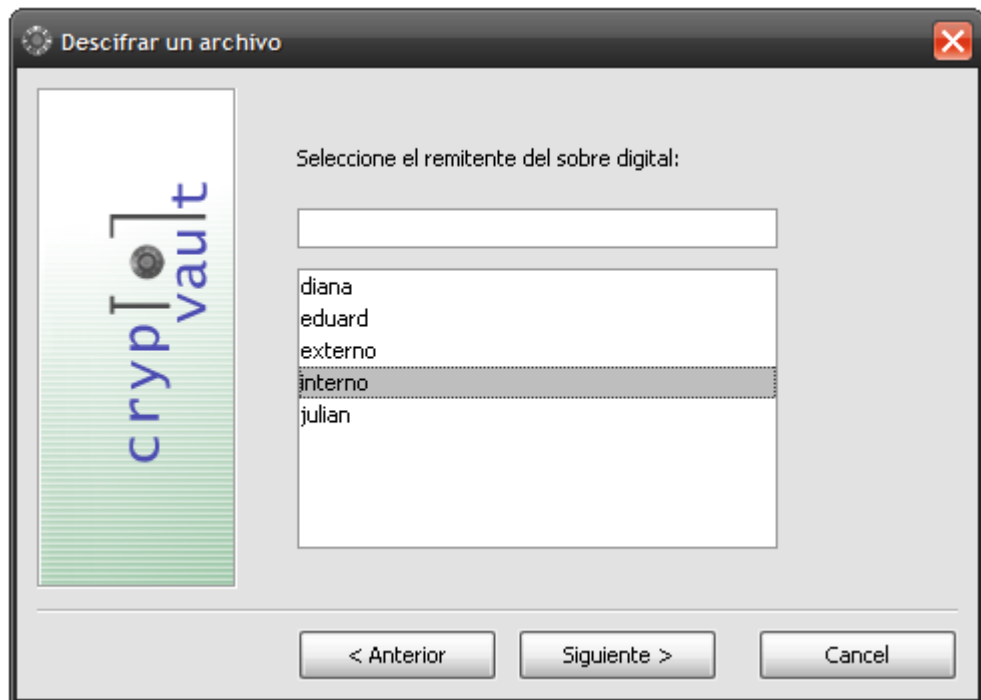
3. Escoja el directorio en donde desea guardar el archivo descifrado. Crypto Vault le sugerirá guardarlo en la carpeta "Mis Archivos Descifrados", pero si desea guardarlo en otra ubicación puede oprimir "Explorar" y seleccionar otro directorio.



4. Seleccione si desea o no verificar el remitente del sobre digital. Esto le permitirá estar seguro de quién envió el sobre antes de abrirlo. Si selecciona "Sí" continúe con el paso 5, de lo contrario vaya al paso 6.



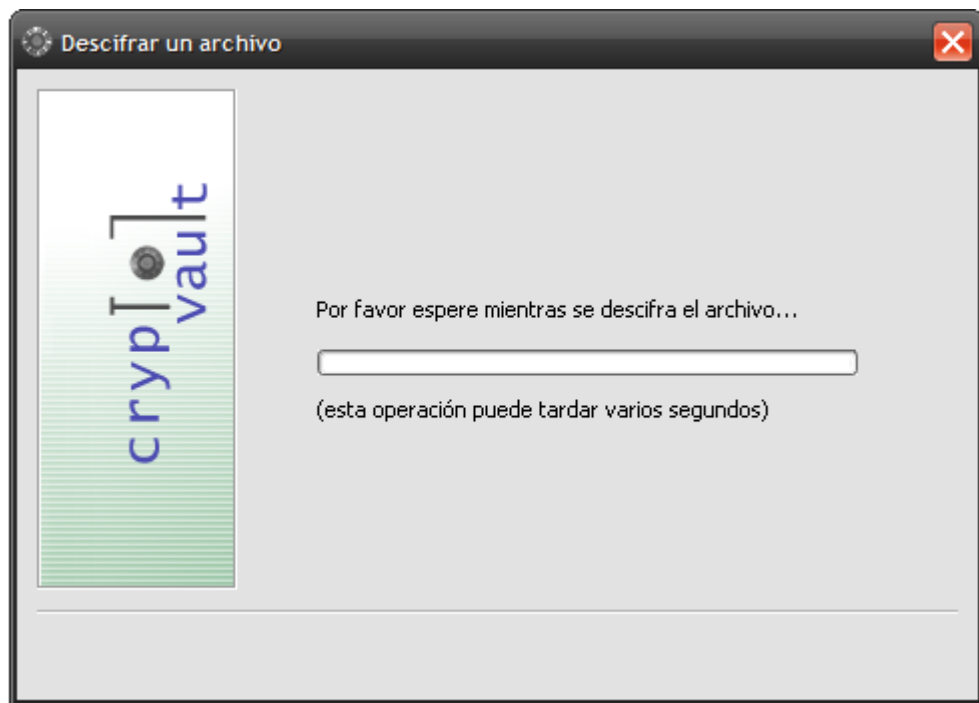
5. Si desea verificar el remitente del sobre digital aparecerá una ventana de selección del remitente. En Crypto Vault, los destinatarios registrados se consideran también potenciales remitentes. Si no encuentra el remitente que desea deberá registrarlo como si fuera un nuevo destinatario (ver *b. Adicionando un nuevo destinatario*).



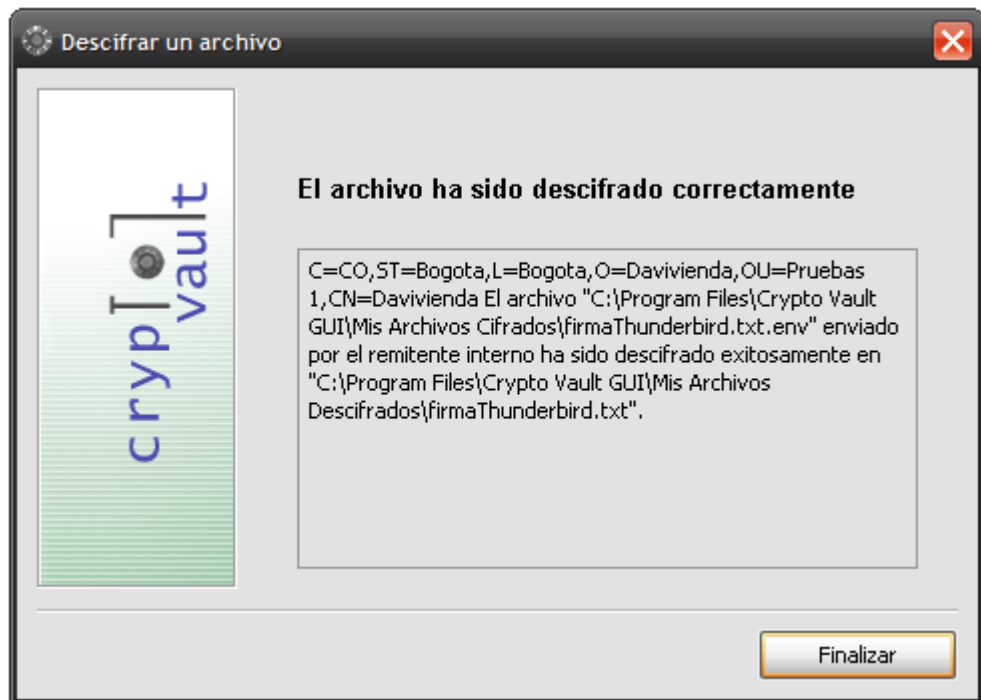
6. Escriba la contraseña de la llave privada y a continuación, oprima el botón "Siguiete".



7. Crypto Vault le indicará el progreso del proceso de desciframiento del archivo.

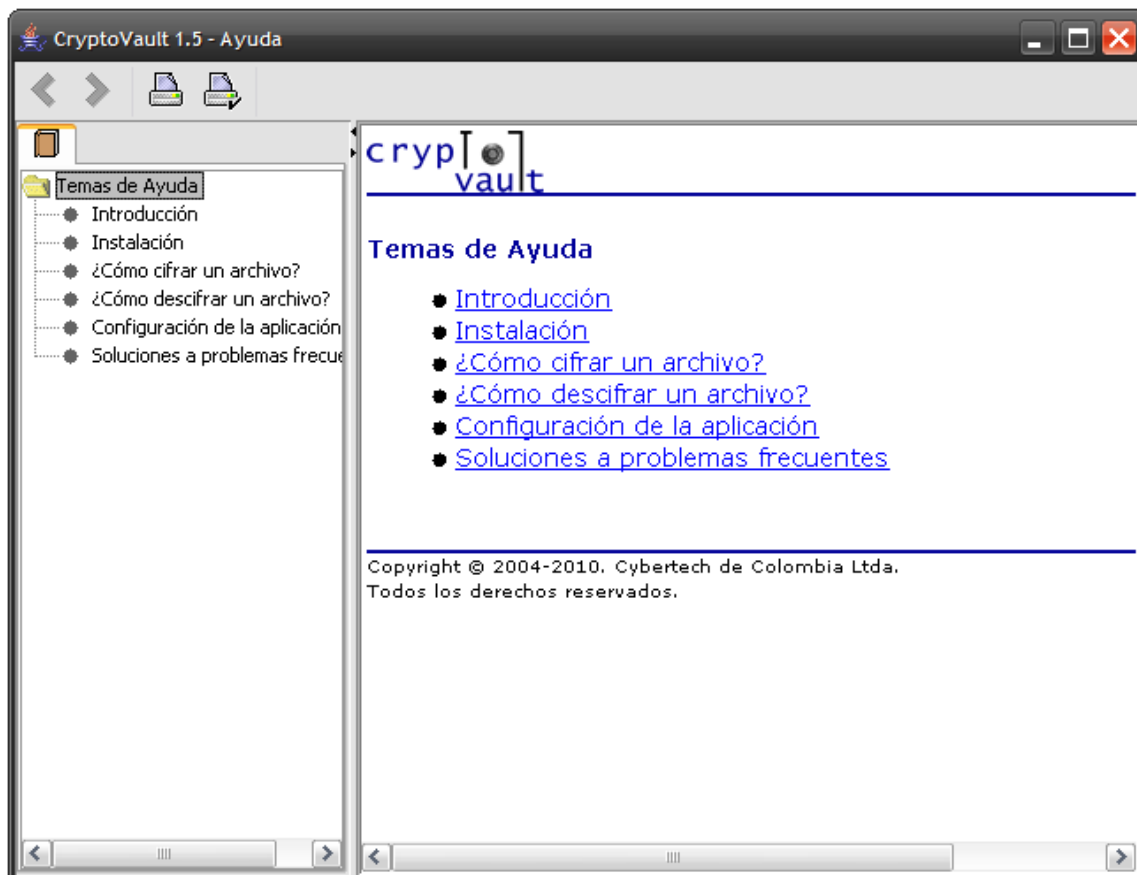


Una vez terminada la operación, aparecerá el mensaje "El archivo ha sido descifrado exitosamente".



e. Consultando la ayuda de la aplicación

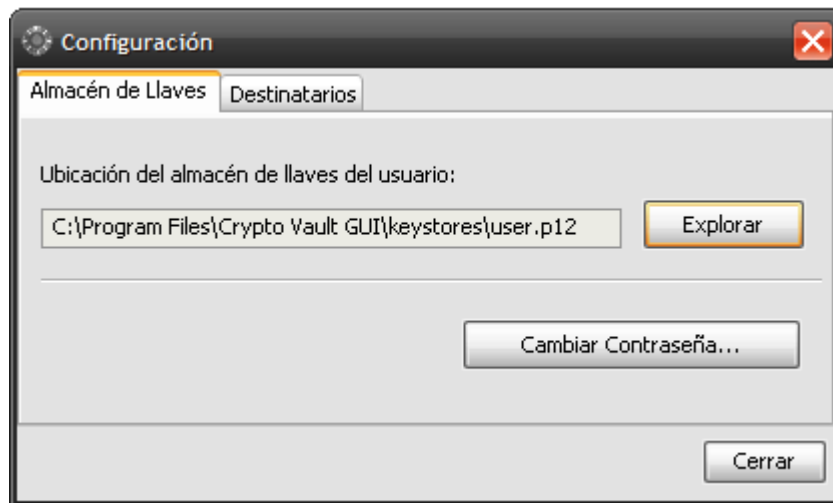
1. Oprima el botón "?" en la parte inferior derecha de la pantalla principal. Esto abrirá el visor de ayuda. La ayuda incorporada se ve como aparece en la siguiente figura:



Para navegar la ayuda basta utilizar el árbol en el marco izquierdo o los enlaces que aparecen en el marco de contenido

f. Cambiando la contraseña del almacén de llaves.

1. Acceda al diálogo de configuración presionando el botón **"Configurar Aplicación"** que aparece en la pantalla principal de la aplicación.
2. Oprima el botón "Cambiar Contraseña" que aparece en la pestaña "Almacén de llaves". Esto abrirá el ayudante para cambiar contraseña.



3. Escriba su contraseña actual en la casilla "Contraseña Anterior" y la contraseña nueva en las casillas "Contraseña nueva" y "Confirmar Contraseña". A continuación, oprima el botón "Siguiente".



La contraseña nueva debe tener al menos ocho caracteres de longitud. El botón "Siguiente" no se habilitará a menos que dicha contraseña tenga la longitud mínima y haya sido escrita de forma idéntica en las dos casillas.

Adicionalmente, la contraseña nueva debe ser diferente a las últimas nueve contraseñas seleccionadas.

4. Aparecerá el mensaje "La contraseña de la llave privada ha sido modificada".



Procedimientos Administrativos relacionados con CryptoVault.

Este capítulo describe los procedimientos administrativos que es necesario atender para mantener a CryptoVault funcionando de manera óptima

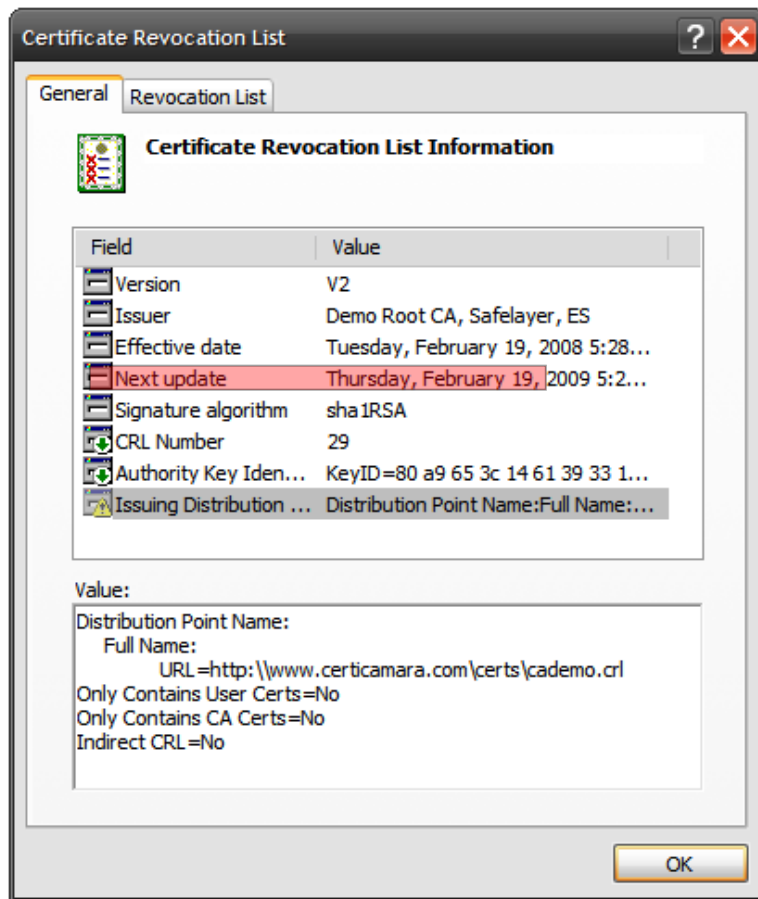
Actualización de la CRL.

La lista de Certificados Revocados (CRL) contiene los certificados que han sido revocados, esto es que no son válidos, por diferentes razones como:

- La entidad que solicitó el certificado ya no existe.
- El almacén de llaves asociado a este certificado ha sido comprometido, entonces alguna persona no autorizada podría firmar o descifrar información.

La lista de Certificados Revocados es expedida por la Autoridad Certificadora con la cual se esté trabajando. Dado que la versión actual de CryptoVault acepta varias Autoridades Certificadoras, es necesario que se disponga de las CRL de cada una de ellas.

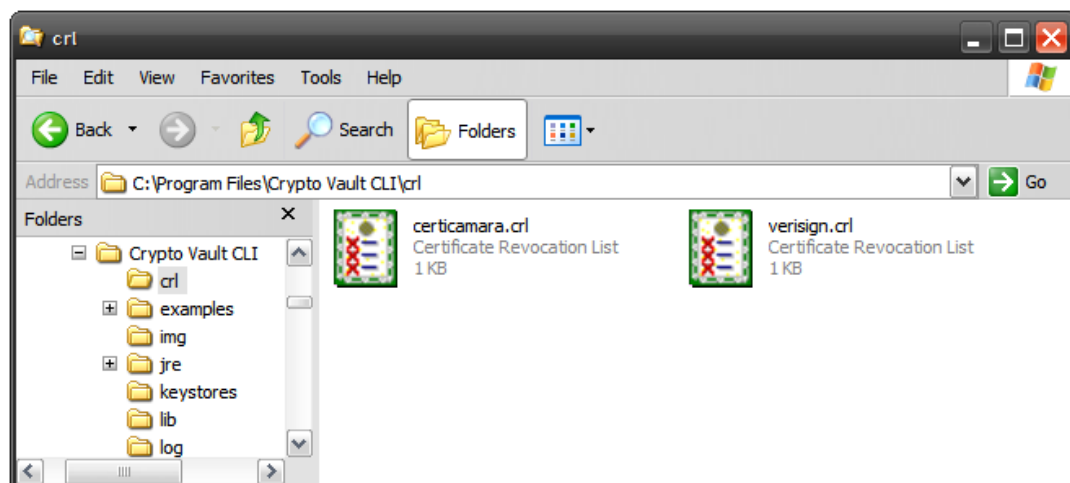
Para determinar la fecha de publicación de la siguiente CRL, obtenga una versión actualizada de ésta y consulte el campo "Next Update".



Este campo indica la fecha en la que una nueva versión de la CRL será emitida. En ese momento será necesario obtenerla y colocarla en la carpeta “crl” del directorio de instalación.

En las versiones anteriores de CryptoVault era necesario que el archivo se llamara “ca.crl”. Sin embargo en esta versión se toman todos los archivos de tipo CRL que estén en esta carpeta.

Recuerde que si usted acepta más de una Autoridad Certificadora, debe tener las CRL de cada una de ellas:



Asignación de permisos sobre los recursos en el directorio de instalación de CryptoVault.

Se recomienda proteger de manera especial el archivo “keystore.cfg” presente en todas las variedades de CryptoVault excepto en la GUI, dado que éste contiene tanto la ubicación del almacén de llaves como su *password* (con el fin de habilitar el uso automático de CryptoVault por parte de otras aplicaciones).

Adicionalmente se sugiere atender la siguiente tabla de asignación de permisos para aquellos usuarios autorizados a usar CryptoVault:

Recurso	Requiere escritura	Requiere lectura	Requiere ejecución
crl/ca.crl		•	
jre/*.*		•	
lib/*.*		•	
log/*.*	•	•	
Mis Archivos Cifrados	•	•	
Mis Archivos Descifrados	•	•	
CryptoVaultDLL.dll		•	
pthreadVC2.dll		•	
consecutive.cfg	•	•	
envelope.dtd		•	
signedData.dtd		•	
keystore.cfg	*	•	
<keystore>.p12	*	•	
recipients.p12	•	•	
Uninstall.exe		•	•
CryptoVault-1.6???.exe		•	•
CryptoVault-1.6???.jar		•	
history.dat	•	•	
msvcrt.dll		•	

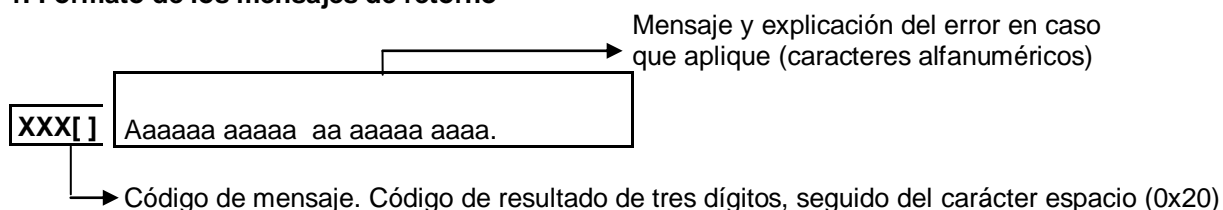
* Solo es necesario si se planea cambiar la contraseña por medio de la aplicación.

Cambio periódico de la contraseña del almacén de llaves.

Recuerde cambiar periódicamente la contraseña del almacén de llaves. La contraseña por defecto de los almacenes de prueba es “changeit”. Una vez estos están en producción se debería cambiar la contraseña para evitar usos malintencionados los almacenes.

Códigos y mensajes de error generados por CryptoVault.

1. Formato de los mensajes de retorno



2. Descripción detallada de los mensajes de retorno

Cód.	Descripción.	Explicación.	Operación.
501	Error interno de la aplicación.	El mensaje de retorno incluye una descripción detallada del error (puede ocupar varias líneas).	Aplicación
503	No se proporcionó ningún argumento a la aplicación.	Es necesario usar modificadores y argumentos para hacer uso de la funcionalidad de la aplicación.	
505	No se puede acceder (leer o escribir) un recurso de la aplicación.	Revisar el mensaje de retorno para determinar cual es el recurso inaccesible y asegurarse que este disponible y listo para ser usado por la aplicación. La sintaxis del mensaje de retorno es: No es posible acceder "<Nombre del recurso inaccesible>".	
507	Los parámetros del modificador seleccionado no son correctos, o están incompletos.	Verificar la sintaxis de uso de la aplicación.	
509	La contraseña del almacén de llaves no es correcta.	Verificar la contraseña del almacén de llaves.	
513	La lista de revocaciones "crl/ca.crl" no contiene una lista de revocaciones válida.	Verificar la integridad de "crl/ca.crl". En caso que efectivamente este corrupta, obtener la CRL válida de la Autoridad Certificadora.	
Cód.	Descripción.	Explicación.	Operación.

517	El ambiente de ejecución de Java no está correctamente configurado.	<p>Verificar que los archivos de jurisdicción "local_policy.jar" y "US_export_policy.jar" en la carpeta lib/security del JRE estén debidamente actualizados para uso de criptografía de fortaleza ilimitada.</p> <p>También, verificar que el proveedor de BC esté debidamente registrado en el archivo "java.security" (en la misma carpeta) y que la implementación del proveedor se encuentre incluida en el directorio "lib/ext" del JRE.</p>	Aplicación (.)
515	Hay un problema con la lista de revocaciones.	Revisar el mensaje de retorno para determinar el problema específico.	
511	No es posible acceder el certificado digital de la CA del almacén de llaves.	Verificar que la entrada de llave privada del usuario (alias "user") contenga la cadena de certificación, en donde la primera entrada debe corresponder al certificado de la CA (que a su vez debe coincidir con la entrada confiable de alias "ca").	
200	El proceso de ciframiento del archivo fue exitoso.	<p>El mensaje de respuesta tiene la siguiente sintaxis:</p> <p>El archivo "<Nombre del archivo a cifrar>" se ha cifrado exitosamente en "<Nombre del archivo con el sobre digital XML>".</p>	Ciframiento
201	El proceso de ciframiento del contenido fue exitoso.	<p>El mensaje de respuesta tiene la siguiente sintaxis:</p> <p>Se han cifrado exitosamente "<Cantidad de bytes a cifrar>" bytes en "<Nombre del archivo con el sobre digital XML>".</p>	
405	El certificado del usuario o del remitente ha expirado.	Para generar el sobre digital es necesario que tanto el certificado del usuario (remitente), como el del destinatario no hayan expirado. En caso contrario, es preciso renovar el certificado vencido con la Autoridad Certificadora.	
451	No existe el destinatario especificado en la lista de destinatarios de la aplicación.	Revise el alias del destinatario del sobre.	
Cód.	Descripción.	Explicación.	Operación.

453	No existe el archivo a ser cifrado en el sobre digital.	El mensaje de respuesta indica el path completo del archivo inexistente con la siguiente sintaxis: El archivo a ser cifrado (<Ubicación del archivo inexistente>) no existe o no es posible abrirlo.	Ciframiento (.)
455	No es posible crear el archivo XML con el sobre digital.	El mensaje de respuesta indica el path completo del archivo que no se puede crear con la siguiente sintaxis: No es posible crear el archivo cifrado en "<Ubicación inaccesible>".	
457	No es posible acceder a la llave privada en el almacén de llaves.	Revise que la llave privada esté almacenada bajo el alias "user" en el almacén de llaves especificado en el archivo "keystore.cfg".	
459	No es posible acceder al archivo de consecutivos de sobres digitales.	Verifique que el archivo "consecutive.cfg" se encuentre en el directorio de instalación de la aplicación y que tenga permisos de lectura y escritura para el usuario que ejecuta la aplicación.	
200	La contraseña de la llave privada ha sido alterada con éxito.	-	Cambio de Contraseña Almacén de Llaves
485	La contraseña actual no coincide con la suministrada.	Verificar la contraseña suministrada para que coincida con la actual.	
487	Los campos de entrada de la nueva contraseña no coinciden.	Es necesario que la nueva contraseña coincida con su respectiva entrada de verificación.	
489	La contraseña nueva y la actual son idénticas.	No tiene sentido cambiar la contraseña por una idéntica.	
491	La nueva contraseña es muy corta.	El mensaje de retorno indica la longitud mínima esperada para el nuevo password con la siguiente sintaxis: La contraseña nueva es demasiado corta, debería tener por lo menos <Número mínimo de caracteres> caracteres.	
Cód.	Descripción.	Explicación.	Operación.

493	La contraseña se cambió exitosamente en el almacén de llaves pero no se pudo actualizar en el archivo de configuración (keystore.cfg).	La contraseña del almacén de llaves se cambió exitosamente. Sin embargo, no se pudo actualizar en el archivo de configuración "keystore.cfg", posiblemente por ausencia de permisos para modificar tal archivo. Es necesario actualizar el valor de la contraseña en este archivo manualmente.	Cambio de Contraseña Almacén de Llaves (.)
200	El proceso de desciframiento del archivo fue exitoso.	El mensaje de respuesta tiene la siguiente sintaxis: <subject del certificado del remitente> El archivo "<Nombre del archivo XML que contiene el sobre>" ha sido descifrado exitosamente en "<Nombre del archivo descifrado>".	Desciframiento
301	El archivo XML que contiene el sobre digital está mal formado.	Revisar la sintaxis del archivo que contiene el sobre digital.	
303	El archivo XML que contiene el contenido firmado del sobre digital está mal formado.	Revisar la sintaxis de generación del archivo que contiene el contenido firmado del sobre digital.	
401	El sobre esta bien formado, pero el desciframiento no fue exitoso por firma inválida.	Posiblemente el contenido del sobre fue manipulado. No es posible confiar en dicho contenido.	
403	El sobre esta bien formado, pero el desciframiento no fue exitoso porque el certificado del remitente es inválido.	Revisar el certificado digital del remitente.	
405	El sobre esta bien formado, pero el desciframiento no fue exitoso porque el certificado del remitente ha expirado.	Es necesario que el remitente renueve su certificado digital con la Autoridad Certificadora en uso.	
407	No existe el archivo con el sobre digital a ser descifrado.	El mensaje de respuesta indica el path completo del archivo inexistente con la siguiente sintaxis: El archivo a ser descifrado <Ubicación del archivo inexistente> no existe o no es posible abrirlo.	
409	No es posible crear el archivo descifrado en la ubicación especificada.	El mensaje de respuesta indica el path completo del archivo que no se puede crear con la siguiente sintaxis: No es posible crear el archivo descifrado en "<Ubicación inaccesible>".	
Cód.	Descripción.	Explicación.	Operación.

411	El sobre digital no está dirigido al usuario.	El sobre debe estar dirigido al usuario para poder acceder a su contenido descifrado.	Desciframiento (.)
413	La Autoridad Certificadora emisora del certificado del nuevo destinatario/remite nte no coincide con ninguna de las Autoridades Certificadoras en su almacén de llaves.	Es necesario que importe el certificado de la Autoridad Certificadora que firmó el certificado del nuevo remitente/destinatario. (ver Importar certificados de otras Autoridades Certificadoras)	
415	La Autoridad Certificadora emisora del certificado del remitente del mensaje no coincide con ninguna de las Autoridades Certificadoras en su almacén de llaves.	Es necesario que importe el certificado de la Autoridad Certificadora que firmó el certificado del remitente para poder verificar su certificado. (ver Importar certificados de otras Autoridades Certificadoras)	
417	El certificado del firmante ha sido revocado por la Autoridad Certificadora.	El certificado que usa el remitente para firmar no puede estar revocado por la Autoridad Certificadora.	
419	El remitente del sobre digital no es el esperado.	El certificado digital correspondiente al alias proporcionado como remitente no corresponde al certificado digital con que fue firmado el sobre digital, por lo tanto el remitente no es el esperado.	
421	No fue posible comparar que el remitente del sobre sea el mismo que el esperado.	Este error ocurre cuando no es posible comparar el certificado digital del remitente del sobre digital con el certificado digital correspondiente al alias proporcionado para verificación.	
200	El destinatario ha sido agregado con éxito.	El mensaje de retorno incluye el alias del nuevo destinatario con la siguiente sintaxis: El destinatario "<Alias del nuevo destinatario>" ha sido creado con éxito.	Adición Destinatario
403	El certificado del nuevo destinatario es inválido.	El archivo especificado donde se encuentra el certificado digital del nuevo destinatario no contiene un certificado X.509 válido.	
405	El certificado digital del nuevo destinatario ha expirado.	Es necesario que el nuevo destinatario renueve su certificado ante la Autoridad Certificadora.	
413	La Autoridad Certificadora emisora del certificado del nuevo destinatario/remite nte no coincide con ninguna de las Autoridades Certificadoras en su almacén de llaves.	Es necesario que importe el certificado de la Autoridad Certificadora que firmó el certificado del nuevo remitente/destinatario.	
417	El certificado del nuevo destinatario ha sido revocado por la Autoridad Certificadora.	El certificado del nuevo destinatario no puede estar revocado por la Autoridad Certificadora.	

475	El alias de destinatario ya está en uso.	Usar otro alias para adicionar el nuevo destinatario.	
Cód.	Descripción.	Explicación.	Operación.
477	No es posible encontrar el archivo que contiene el certificado digital del destinatario.	El mensaje de retorno contiene el path completo del archivo que no es posible encontrar con la siguiente sintaxis: No es posible encontrar el certificado del destinatario en "<Ubicación inválida del certificado>".	Adición Destinatario
479	No es posible adicionar el nuevo destinatario debido a que su certificado ya se encuentra en uso bajo otro alias.	Basta asociar un certificado con un solo alias para poder usarlo como destinatario de la aplicación.	
200	El destinatario ha sido eliminado con éxito.	El mensaje de retorno incluye el alias del destinatario recién eliminado con la siguiente sintaxis: El destinatario "<Alias del destinatario eliminado>" ha sido eliminado con éxito.	Eliminación Destinatario
481	El alias del destinatario no existe.	No es posible eliminar un destinatario inexistente.	
601	Alguno de los argumentos suministrados a la aplicación no es válido.	Verificar que los argumentos que se pasan a las funciones del API cumplan con la especificación de éstas.	API (Invocación nativa Win32)
602	No es posible determinar la ruta de instalación de la aplicación.	Revisar la entrada del registro que contiene el directorio de instalación de Crypto Vault.	
603	Error interno: No fue posible inicializar una clase.	Solicite soporte a su proveedor.	
604	No fue posible instanciar la máquina virtual de Java.	Solicite soporte a su proveedor.	
605	No fue posible establecer el ambiente de ejecución de la aplicación (directorio de trabajo y variable de entorno PATH).	Solicite soporte a su proveedor.	
606	Error interno: No fue posible obtener un método.	Solicite soporte a su proveedor.	
607	Error interno: No fue posible invocar un método.	Solicite soporte a su proveedor.	
608	La operación se ejecutó, pero no fue posible procesar la respuesta retornada por un llamado a un método.	Solicite soporte a su proveedor.	

609	La operación se ejecutó, pero no fue posible reestablecer el ambiente del invocador (directorio de trabajo o variable de entorno PATH).	Solicite soporte a su proveedor.	
Cód.	Descripción.	Explicación.	Operación.
200	El ambiente de invocación directa desde aplicaciones Java se ha configurado correctamente.	Es posible usar la fachada para invocar servicios de la aplicación.	JVA (Invocación directa desde aplicaciones Java)
701	No es posible determinar la ruta de instalación de la aplicación.	Es necesario configurar la ubicación de la aplicación en el sistema de archivos para poder hacer uso de la aplicación desde otras aplicaciones Java.	
703	El ambiente de ejecución de Java no está correctamente configurado.	<p>Verificar que los archivos de jurisdicción "local_policy.jar" y "US_export_policy.jar" en la carpeta lib/security del JRE estén debidamente actualizados para uso de criptografía de fortaleza ilimitada.</p> <p>También, verificar que el proveedor de BC esté debidamente registrado en el archivo "java.security" (en la misma carpeta) y que la implementación del proveedor se encuentre incluida en el directorio "lib/ext" del JRE. (En su defecto, debe incluirse dicha librería en el classpath de la aplicación usuaria).</p>	

Solucionando Problemas Frecuentes desde la Interfaz Gráfica.

1. Al intentar cifrar un archivo aparece el mensaje "No hay ningún destinatario disponible"

Puede agregar destinatarios desde el dialogo "Configurar Aplicación".

2. Al intentar cifrar o descifrar archivo aparece el mensaje "No se encontró el almacén de llaves"

CryptoVault no encuentra el almacén de llaves del usuario.

Para cifrar y descifrar archivos, CryptoVault necesita un almacén de llaves en formato PKCS12 que contiene la llave pública y privada del usuario. Por defecto, CryptoVault busca dicho almacén en un archivo de nombre "user.p12" en el directorio "keystores" que se encuentra en la carpeta de instalación del programa.

Recuerde que puede especificar una ubicación diferente para el almacén de llaves desde el diálogo de configuración.

3. Al intentar cambiar la contraseña aparece el mensaje "La contraseña seleccionada ya ha sido utilizada previamente"

CryptoVault controla que al cambiar la contraseña, la contraseña seleccionada sea diferente a las nueve contraseñas anteriores. Intente cambiarla nuevamente, utilizando una contraseña distinta.

4. Al descifrar un archivo aparece el mensaje "El sobre digital no se encuentra dirigido al usuario"

Al cifrar un archivo CryptoVault genera un sobre digital dirigido a un único destinatario. No es posible descifrar un sobre digital a menos que el sobre digital se encuentre dirigido al usuario actual de la aplicación. Verifique que el almacén de llaves se encuentre configurado correctamente.

5. No encuentro el archivo que acabo de cifrar

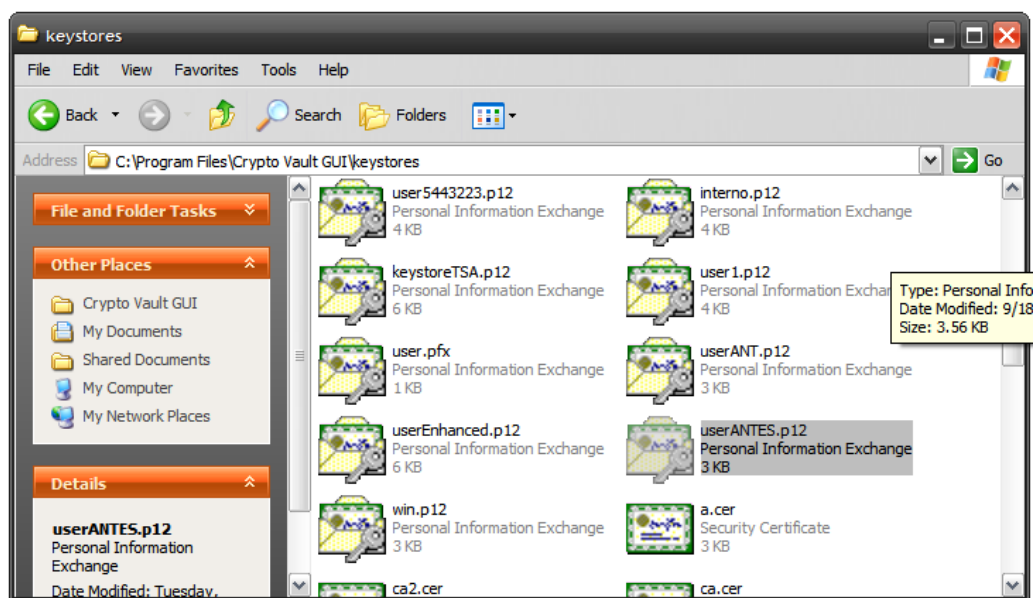
Al finalizar el proceso de cifrado de un archivo, CryptoVault informa la ubicación del archivo de destino. Por defecto, CryptoVault deposita los sobres digitales en el directorio "Mis Archivos Cifrados" ubicada en la carpeta de instalación de la aplicación.

6. No encuentro el archivo que acabo de descifrar

Al finalizar el proceso de descifrado de un archivo, CryptoVault informa la ubicación del archivo de destino. Por defecto, CryptoVault deposita los archivos descifrados en el directorio "Mis Archivos Descifrados" ubicada en la carpeta de instalación de la aplicación.

Almacenes de Llaves compatibles con CryptoVault.

Los Almacenes de llaves se emplean para guardar las claves públicas y privadas que se necesitan al utilizar Crypto Vault. Estos almacenes de llaves suelen ser archivos con extensión .p12 (o .pfx si son generados por los componentes criptográficos de Windows).



Dado que los almacenes de llaves permiten cifrar y descifrar mensajes, son **INFORMACIÓN CONFIDENCIAL**, y deben ser tratados con sumo cuidado (por ejemplo no se deben enviar por correo electrónico). Los almacenes de llaves están protegidos por una contraseña que deben conocer únicamente las personas autorizadas.

El estándar para almacenes de llaves se llama PKCS12, que fue definido por RSA, y se encuentra en la siguiente URL:

<http://www.rsa.com/rsalabs/node.asp?id=2138>

Para que un almacén de llaves pueda ser usado para cifrar y descifrar mensajes de Crypto Vault, **debe cumplir los siguientes requerimientos:**

- Respetar el estándar PKCS12.
- Tener la llave privada y pública propias ("keyEntry"). **Estas llaves deben ser de 2048 bits.**

- Tener el certificado propio.
- Tener uno o más certificados de las Autoridades Certificadoras confiables ("trustedCertEntry").

Existen varias maneras de crear almacenes de llaves, para las explicaciones de este capítulo utilizaremos la herramienta "keytool" que viene instalada con Java. Dado que las versiones GUI, CLI y API tienen su propio ambiente de java (JRE), se recomienda utilizar dicho keytool. La documentación de esta herramienta se puede encontrar en:

<http://java.sun.com/j2se/1.4.2/docs/tooldocs/windows/keytool.html>

Las opciones de keytool que se utilizarán son las siguientes:

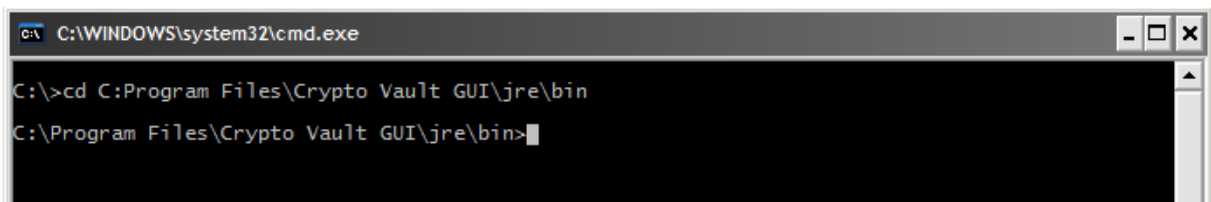
- "-genkey": Genera un almacén de llaves con una clave privada y pública.
- "-list -v": Muestra el contenido de un almacén de llaves.
- "-certreq": Genera peticiones de certificado con la llave pública propia.
- "-import": Importa certificados, sirve para importar certificados propios y de Autoridades Certificadoras confiables.

A continuación se describen algunas tareas relacionadas a almacenes de llaves, que son necesarias para usar Crypto Vault.

Creación de Almacenes de Llaves

Este procedimiento crea un almacén de llaves con una llave pública y una llave privada propia.

1. Abrir una ventana de línea de comandos.
2. Para facilitar los comandos, nos vamos a ubicar en la carpeta donde está el keytool.



Si se tiene la variedad GUI, CLI o API, keytool estará dentro de la carpeta:

<Directorio de Instalacion de CryptoVault>/jre/bin

En caso de que se esté utilizando otro ambiente Java, estará en la carpeta:

<Directorio de JAVA>/bin

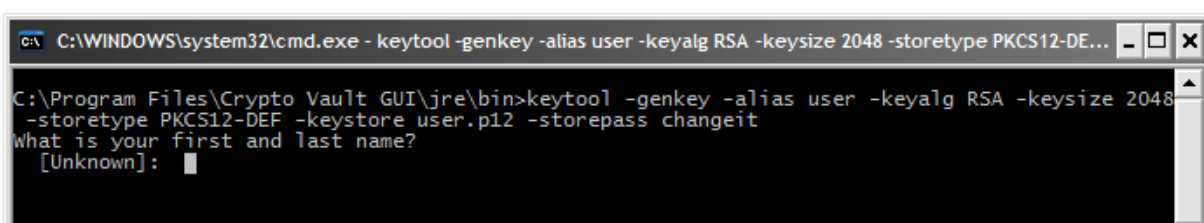
Para verificar que el ambiente Java está correctamente configurado, siga las instrucciones del **capítulo 3 "Instalando CryptoVault."**

3. Introduzca el siguiente comando:

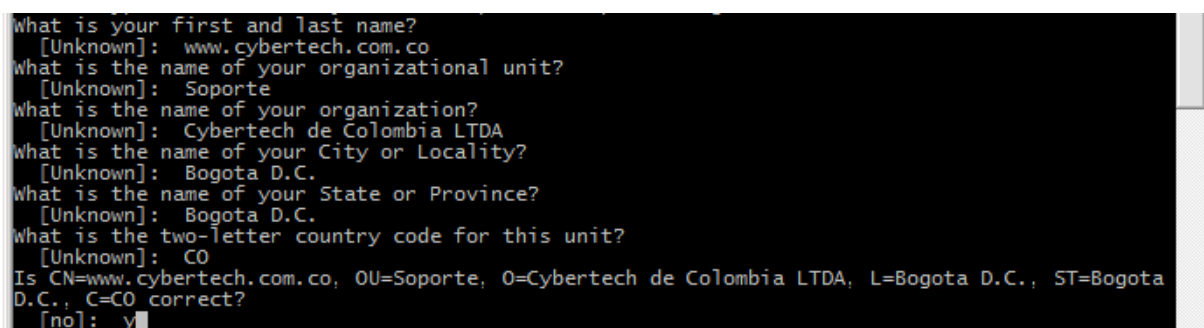
```
keytool -genkey -alias user -keyalg RSA -keysize 2048 -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
```

Este comando tiene varios componentes:

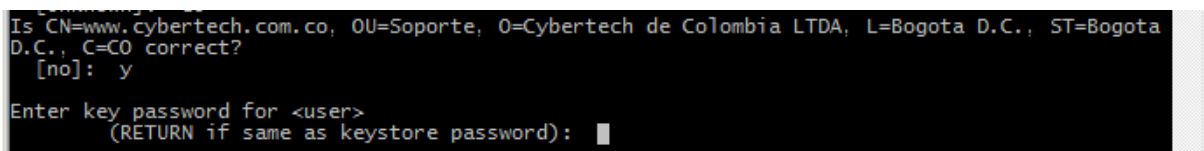
- “-alias user”: significa que la clave pública y privada van a quedar almacenadas en una entrada con nombre “user”.
- “-keyalg RSA”: significa que el algoritmo que se usa para las llaves públicas y privadas es RSA.
- “-keysize 2048”: significa que las llaves tienen tamaño de 2048 bits.
- “-storetype PKCS12-DEF”: significa que el tipo de almacén que se usa es PKCS12.
- “-keystore user.p12”: significa que el almacén de llaves creado va a ser un archivo que se llama “user.p12”.
- “-storepass changeit”: significa que la contraseña con la que se va a crear el almacén de llaves es “changeit”. Esta contraseña puede ser cualquier otra.



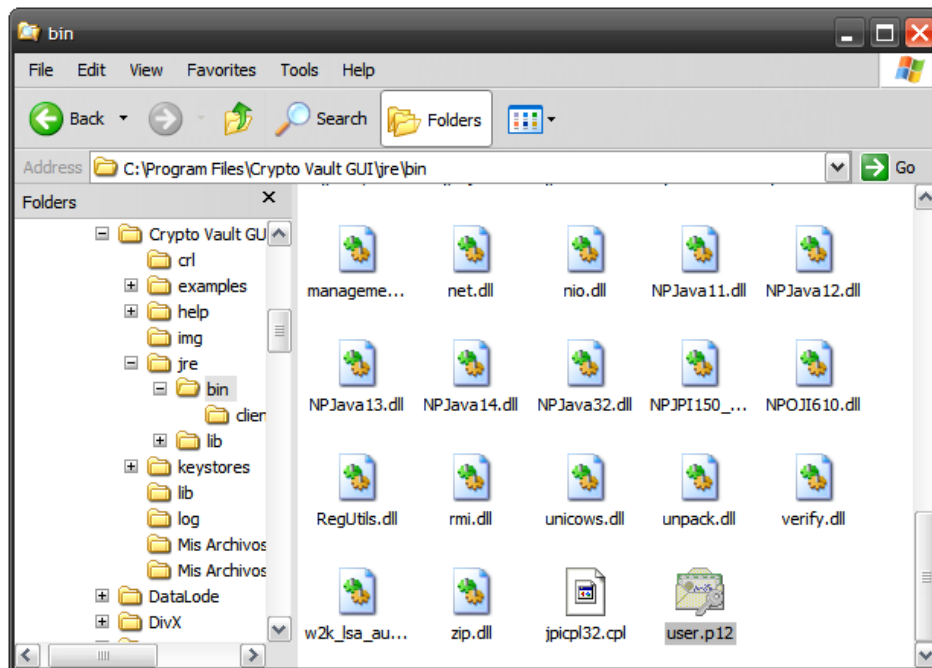
A continuación la herramienta pedirá la información necesaria para generar las claves:



Cuando se pida verificación de los datos ingresados, escriba “Y” y <ENTER>.



Cuando se pida una contraseña adicional, digite <ENTER> para que quede la misma del almacén de llaves.



Cuando el comando se haya completado se generará un archivo de almacén de llaves (en este caso “user.p12”) con las llaves públicas y privadas propias.

4. Para verificar el contenido de este almacén de llaves ejecute el siguiente comando:

```
keytool -list -v -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
```

```
C:\WINDOWS\system32\cmd.exe

C:\Program Files\Crypto Vault GUI\jre\bin>keytool -list -v -storetype PKCS12-DEF -keystore user.p12 -storepass changeit

Keystore type: PKCS12-DEF
Keystore provider: BC

Your keystore contains 1 entry

Alias name: user
Creation date: Mar 29, 2008
Entry type: keyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=www.cybertech.com.co, OU=Soporte, O=Cybertech de Colombia LTDA, L=Bogota D.C., ST=Bogota D.C., C=CO
Issuer: CN=www.cybertech.com.co, OU=Soporte, O=Cybertech de Colombia LTDA, L=Bogota D.C., ST=Bogota D.C., C=CO
Serial number: 47ee8cc5
Valid from: Sat Mar 29 13:39:01 COT 2008 until: Fri Jun 27 13:39:01 COT 2008
Certificate fingerprints:
    MD5: C2:66:85:CC:F4:0E:09:E3:A7:E2:EB:22:2E:55:06:17
    SHA1: 0A:56:95:87:E8:FD:D9:D4:42:50:97:56:88:88:1D:DA:88:92:1A:B7

*****
*****

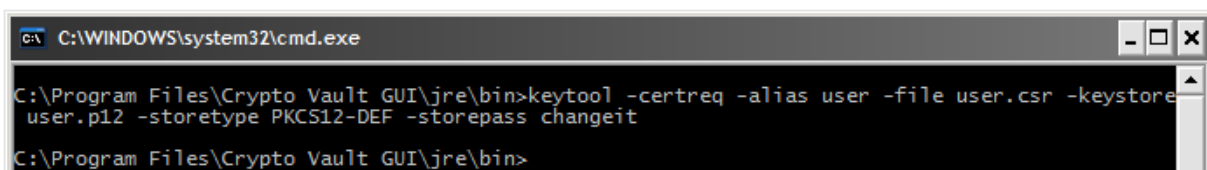
C:\Program Files\Crypto Vault GUI\jre\bin>
```

Debe haber solamente una entrada ("Your keystore contains 1 entry"), de tipo "keyEntry" ("Entry type: keyEntry"), con la información subministrada anteriormente.

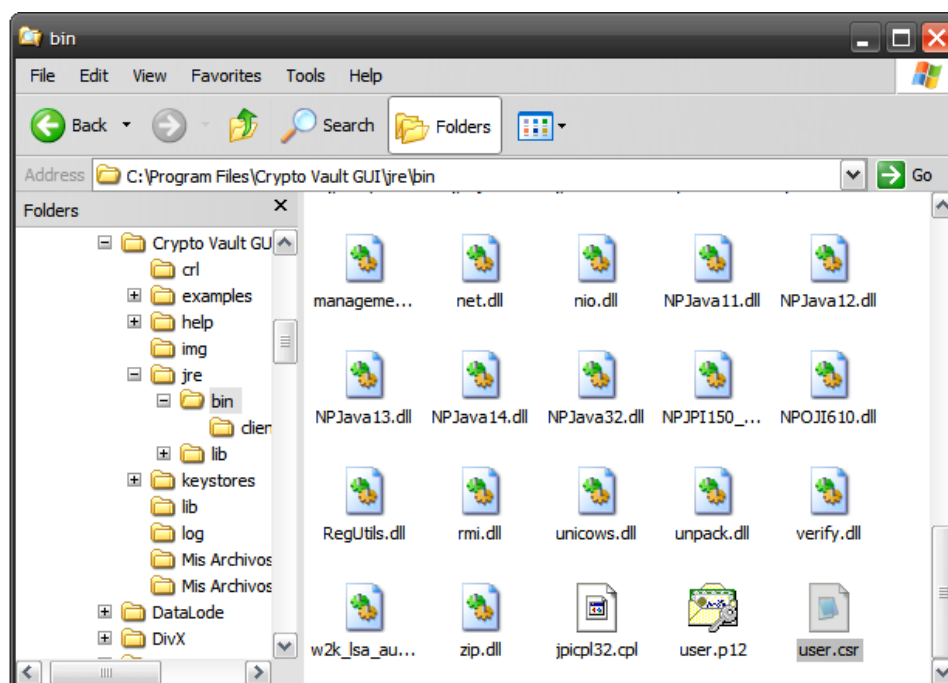
Para poder utilizar este almacén de llaves con Crypto Vault, es necesario pedir un certificado a una Autoridad Certificadora reconocida. Por ejemplo Certicámara o Verisign. Para esto siga los pasos a continuación.

5. Ejecute el siguiente comando:

```
keytool -certreq -alias user -file user.csr -keystore user.p12 -storetype PKCS12-DEF -storepass changeit
```



Este comando generará un archivo de petición de certificado CSR, que se llama "user.csr" en este caso:

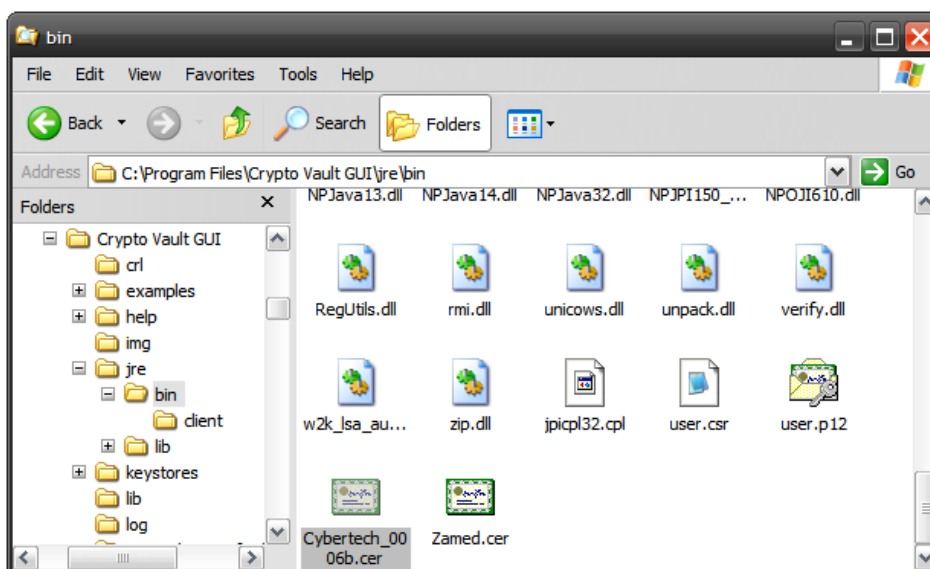


Este archivo CSR contiene únicamente la clave pública propia.

6. Envíe el archivo CSR a la Autoridad Certificadora de su elección. Dependiendo de la Autoridad Certificadora elegida, será necesario seguir un procedimiento para poder obtener este certificado. Dicho procedimiento, dependiendo del caso **puede tardar incluso algunos días**.

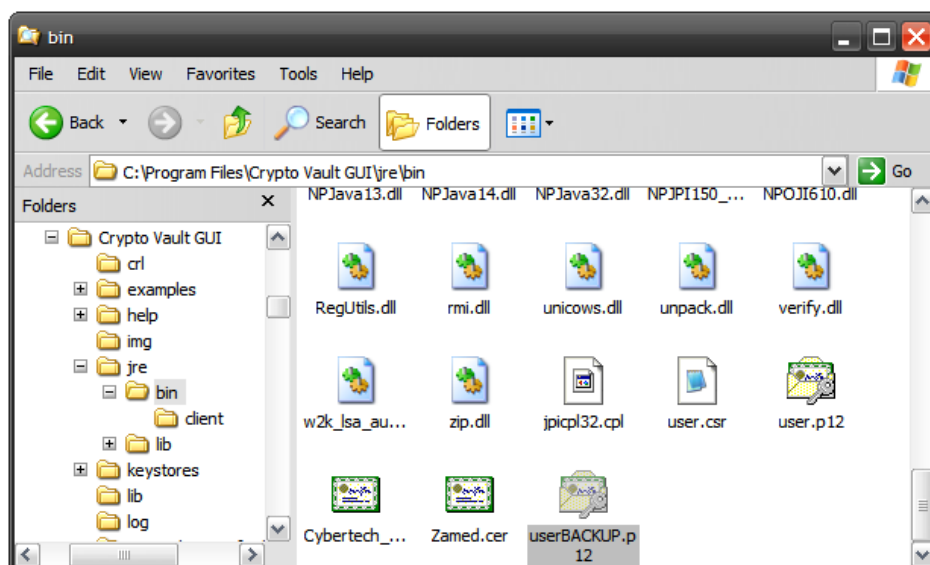
Como respuesta de la Autoridad certificadora se deben recibir los siguientes archivos:

- **Certificado digital propio:** Un archivo .cer firmado por la Autoridad Certificadora. Este archivo debe contener la información suministrada al crear el almacén de llaves en el paso 3. (En este ejemplo “Cybertech_0006b.cer”)
- **Certificado digital de la Autoridad Certificadora:** Un archivo .cer con la información de la Autoridad Certificadora. (En este ejemplo es una autoridad de pruebas llamada “ZAMED”).



Para facilitar los comandos los vamos a copiar en la misma carpeta donde generamos el almacén de llaves.

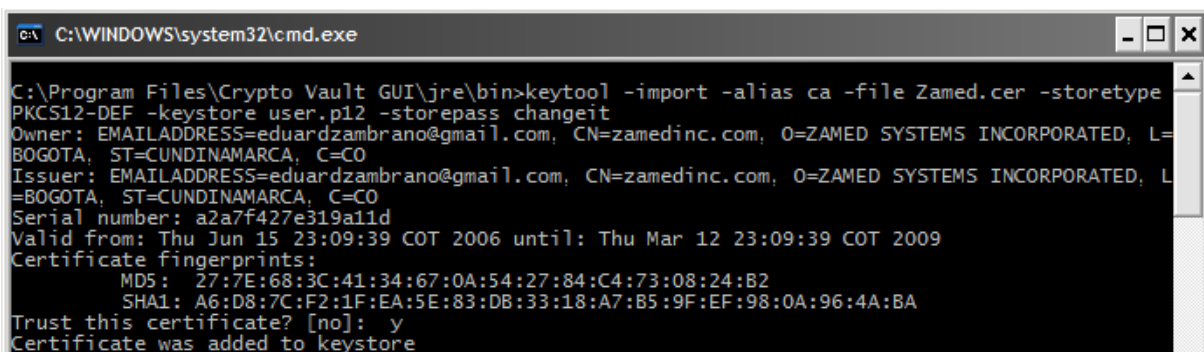
7. **Haga una copia de respaldo** del almacén de llaves (en este caso “user.p12”). Esta es una medida de precaución por si llega a haber un error en los pasos siguientes:



8. Ejecute el siguiente comando:

```
keytool -import -alias ca -file Zamed.cer -storetype PKCS12-DEF -  
keystore user.p12 -storepass changeit
```

Donde “Zamed.cer” debe ser reemplazado por el nombre del certificado de la Autoridad Certificadora que usted haya recibido (Ej: “Certicamara.cer”).



```
C:\WINDOWS\system32\cmd.exe  
C:\Program Files\Crypto Vault GUI\jre\bin>keytool -import -alias ca -file Zamed.cer -storetype  
PKCS12-DEF -keystore user.p12 -storepass changeit  
Owner: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=  
BOGOTA, ST=CUNDINAMARCA, C=CO  
Issuer: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=  
=BOGOTA, ST=CUNDINAMARCA, C=CO  
Serial number: a2a7f427e319a11d  
Valid from: Thu Jun 15 23:09:39 COT 2006 until: Thu Mar 12 23:09:39 COT 2009  
Certificate fingerprints:  
MD5: 27:7E:68:3C:41:34:67:0A:54:27:84:C4:73:08:24:B2  
SHA1: A6:D8:7C:F2:1F:EA:5E:83:DB:33:18:A7:B5:9F:EF:98:0A:96:4A:BA  
Trust this certificate? [no]: y  
Certificate was added to keystore
```

Cuando se pida verificación de la información del certificado, escriba “y” y digite <ENTER>.

Este comando importa el certificado de la Autoridad Certificadora en el almacén de llaves.

9. Para verificar que haya quedado correctamente importado ejecute el mismo comando del paso 4:

```
keytool -list -v -storetype PKCS12-DEF -keystore user.p12 -storepass  
changeit
```

```
C:\WINDOWS\system32\cmd.exe

C:\Program Files\Crypto Vault GUI\jre\bin>keytool -list -v -storetype PKCS12-DEF -keystore user.p12 -storepass changeit

Keystore type: PKCS12-DEF
Keystore provider: BC

Your keystore contains 2 entries

Alias name: ca
Creation date: Mar 29, 2008
Entry type: trustedCertEntry

Owner: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=BOGOTA, ST=CUNDINAMARCA, C=CO
Issuer: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=BOGOTA, ST=CUNDINAMARCA, C=CO
Serial number: a2a7f427e319a11d
Valid from: Thu Jun 15 23:09:39 COT 2006 until: Thu Mar 12 23:09:39 COT 2009
Certificate fingerprints:
    MD5: 27:7E:68:3C:41:34:67:0A:54:27:84:C4:73:08:24:B2
    SHA1: A6:D8:7C:F2:1F:EA:5E:83:DB:33:18:A7:B5:9F:EF:98:0A:96:4A:BA
*****
Alias name: user
Creation date: Mar 29, 2008
Entry type: keyEntry
Certificate chain length: 1
Certificate[1]:
Owner: CN=www.cybertech.com.co, OU=Soporte, O=Cybertech de Colombia LTDA, L=Bogota D.C., ST=Bogota D.C., C=CO
Issuer: CN=www.cybertech.com.co, OU=Soporte, O=Cybertech de Colombia LTDA, L=Bogota D.C., ST=Bogota D.C., C=CO
Serial number: 47ee8cc5
Valid from: Sat Mar 29 13:39:01 COT 2008 until: Fri Jun 27 13:39:01 COT 2008
Certificate fingerprints:
    MD5: C2:66:85:CC:F4:0E:09:E3:A7:E2:EB:22:2E:55:06:17
    SHA1: 0A:56:95:87:E8:FD:D9:D4:42:50:97:56:88:88:1D:DA:88:92:1A:B7
*****
```

Debe haber una nueva entrada con alias “ca” de tipo “trustedCertEntry”, que corresponde al certificado de la Autoridad Certificadora.

10. Ejecute el siguiente comando

```
keytool -import -alias user -file Cybertech_0006b.cer -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
```

Donde “Cybertech_0006b.cer” debe ser reemplazado por el nombre del certificado propio (por ejemplo “BancoDelNorteSA.cer”)

```
C:\WINDOWS\system32\cmd.exe

C:\Program Files\Crypto Vault GUI\jre\bin>keytool -import -alias user -file Cybertech_0006b.cer -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
Certificate reply was installed in keystore
```

Debe aparecer el mensaje “Certificate reply was installed in keystore”.

11. Para verificar que haya quedado correctamente importado, ejecute el comando del paso 4:

```
keytool -list -v -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
```

```

C:\WINDOWS\system32\cmd.exe

C:\Program Files\Crypto Vault GUI\jre\bin>keytool -list -v -storetype PKCS12-DEF -keystore user.p12 -storepass changeit

Keystore type: PKCS12-DEF
Keystore provider: BC

Your keystore contains 2 entries

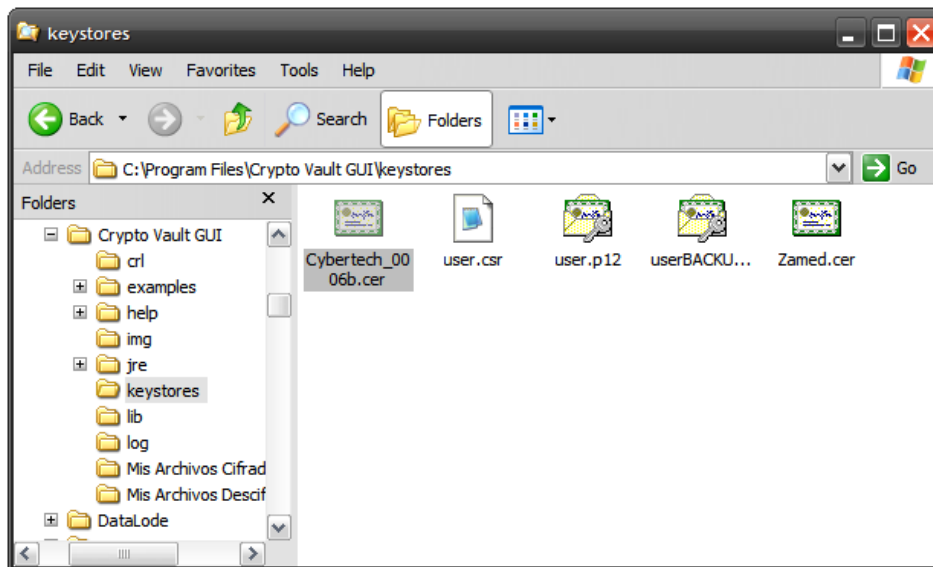
Alias name: ca
Creation date: Mar 29, 2008
Entry type: trustedCertEntry

Owner: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=BOGOTA, ST=CUNDINAMARCA, C=CO
Issuer: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=BOGOTA, ST=CUNDINAMARCA, C=CO
Serial number: a2a7f427e319a11d
Valid from: Thu Jun 15 23:09:39 COT 2006 until: Thu Mar 12 23:09:39 COT 2009
Certificate fingerprints:
    MD5: 27:7E:68:3C:41:34:67:0A:54:27:84:C4:73:08:24:B2
    SHA1: A6:D8:7C:F2:1F:EA:5E:83:DB:33:18:A7:B5:9F:EF:98:0A:96:4A:BA
*****
Alias name: user
Creation date: Mar 29, 2008
Entry type: keyEntry
Certificate chain length: 2
Certificate[1]:
Owner: CN=www.cybertech.com.co, OU=Soporte, O=Cybertech de Colombia LTDA, L=Bogota D.C., ST=Bogota D.C., C=CO
Issuer: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=BOGOTA, ST=CUNDINAMARCA, C=CO
Serial number: 6b
Valid from: Sat Mar 29 14:24:01 COT 2008 until: Tue Mar 24 14:24:01 COT 2009
Certificate fingerprints:
    MD5: 1E:72:A9:47:84:87:AB:17:FA:72:FD:45:85:F7:A8:0A
    SHA1: 0D:3C:D6:90:31:23:AD:07:37:7F:6C:BD:B4:4A:4A:9E:91:E8:51:04
Certificate[2]:
Owner: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=BOGOTA, ST=CUNDINAMARCA, C=CO
Issuer: EMAILADDRESS=eduardzambrano@gmail.com, CN=zamedinc.com, O=ZAMED SYSTEMS INCORPORATED, L=BOGOTA, ST=CUNDINAMARCA, C=CO
Serial number: a2a7f427e319a11d
Valid from: Thu Jun 15 23:09:39 COT 2006 until: Thu Mar 12 23:09:39 COT 2009
Certificate fingerprints:
    MD5: 27:7E:68:3C:41:34:67:0A:54:27:84:C4:73:08:24:B2
    SHA1: A6:D8:7C:F2:1F:EA:5E:83:DB:33:18:A7:B5:9F:EF:98:0A:96:4A:BA
*****

```

La entrada con alias “user” debe tener una cadena de longitud 2 (“Certificate chain length 2”), donde el segundo certificado (“Certificate[2]”) debe tener la información de la Autoridad Certificadora (Es igual al certificado que está en el alias “ca”).

12. En este momento este almacén de llaves está listo para utilizarse con Crypto Vault. Muévelo junto con los certificados a la carpeta “keystores” del directorio de instalación:



13. Configure su variedad de Crypto Vault para que utilice el almacén de llaves creado. Recuerde que **debe enviar el certificado propio** ("Cybertech_0006b.cer" en este ejemplo) a cualquier otra entidad con la cual deba intercambiar información cifrada.

Importar Certificados de Otras Autoridades Certificadoras

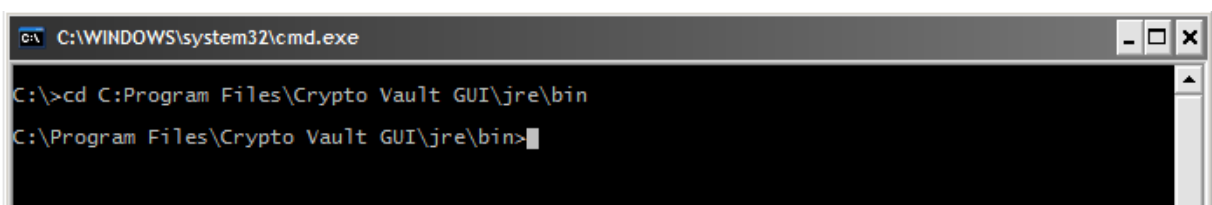
A veces es necesario intercambiar información cifrada con entidades que utilizan otra Autoridad Certificadora diferente a la escogida en el paso 6 de la generación de almacenes de llaves.

Cuando este caso se presente, es necesario importar el certificado de dicha Autoridad Certificadora en el almacén de llaves, de lo contrario surgirá el siguiente error al tratar de agregar un destinatario:

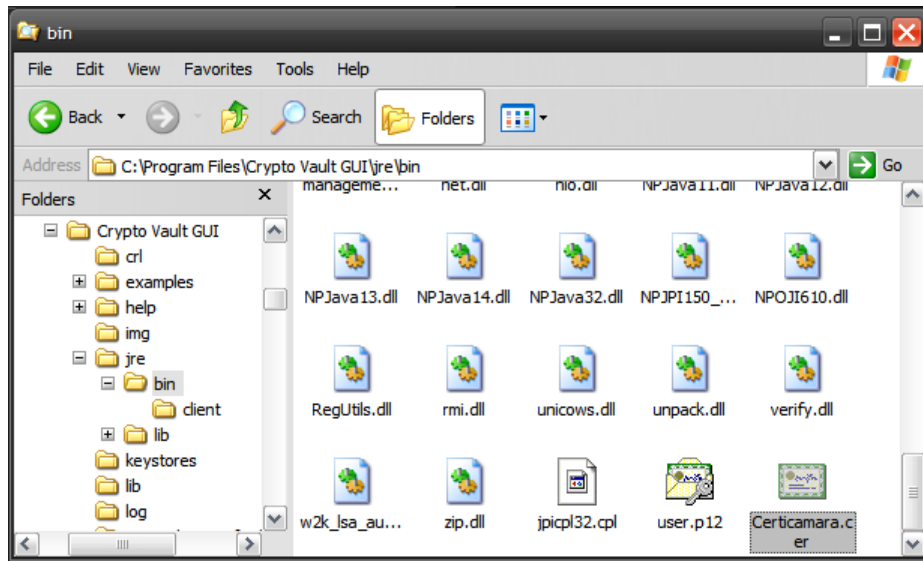
"La Autoridad Certificadora emisora del certificado del nuevo destinatario/remitente no coincide con ninguna de las Autoridades Certificadoras en su almacén de llaves."

Para hacer esta importación siga los pasos a continuación:

1. Abrir una ventana de línea de comandos.
2. Para facilitar los comandos, nos vamos a ubicar en la carpeta donde está el keytool.



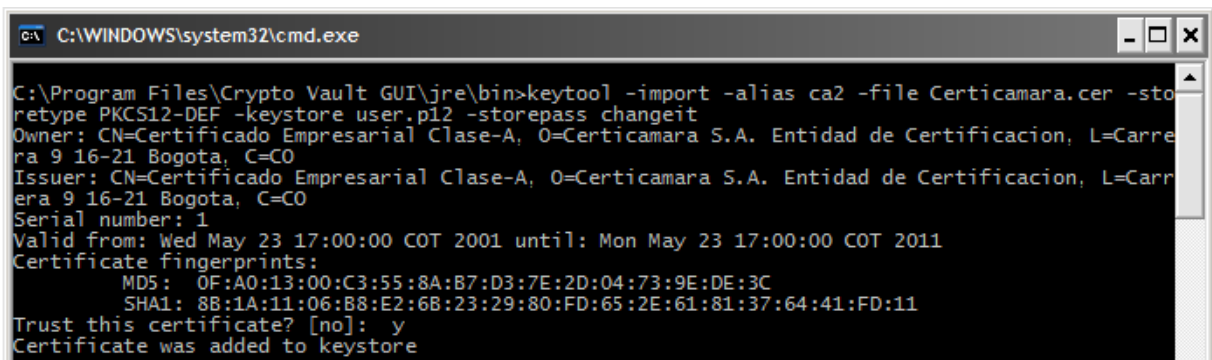
3. Para facilitar los comandos, vamos a mover el almacén de llaves (generalmente en la carpeta "keystores") y el certificado de la otra entidad certificadora a la carpeta donde está el keytool.



En este ejemplo el archivo “user.p12” es el almacén de llaves con el que trabaja Crypto Vault, y “Certicamara.cer” es el certificado de la Autoridad Certificadora adicional.

4. Ejecute el siguiente comando:

```
keytool -import -alias ca2 -file Certicamara.cer -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
```



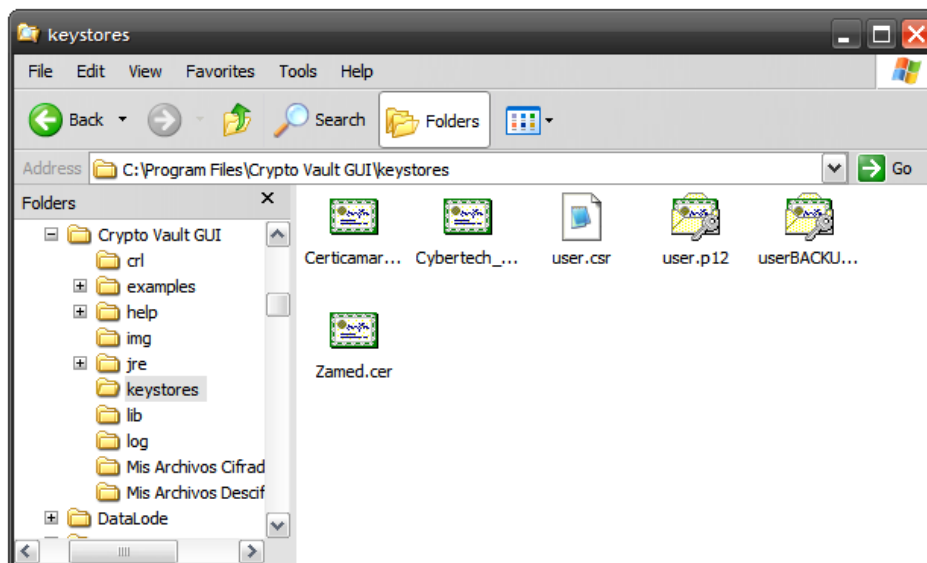
Cuando se pida verificación de la información del certificado, escriba “y” y digite <ENTER>.

5. Para verificar que haya quedado correctamente importado ejecute el mismo comando del paso 4:

```
keytool -list -v -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
```

Debe aparecer una nueva entrada tipo “trustedCertEntry” con la información del certificado de la nueva Autoridad Certificadora.

6. Devuelva el almacén de llaves y el certificado a su ubicación original:



Al usar este almacén de llaves, se pueden importar certificados de esta autoridad certificadora, e intercambiar información cifrada.

Renovación de Certificado

Las Autoridades Certificadoras suelen expedir un certificado con validez de un año. Cuando se finaliza el período de validez de un certificado es necesario pedir otro nuevamente, lo cual se puede hacer de cualquiera de las siguientes dos maneras:

- Se genera desde el principio un almacén de llaves y se realiza todo el procedimiento.
- Se renueva un certificado para un almacén de llaves que ya estaba en uso.

Se recomienda la **primera opción** porque disminuye el riesgo de que un almacén de llaves sea comprometido. En este caso simplemente haga una copia de respaldo del almacén de llaves anterior y genere uno nuevo desde el principio.

Para la **segunda opción** se debe seguir el siguiente procedimiento:

1. Generar un archivo CSR con el comando del paso 5 de la generación de almacenes de llaves:

```
keytool -certreq -alias user -file user.csr -keystore user.p12 -storetype PKCS12-DEF -storepass changeit
```

2. Enviar el CSR generado a la Autoridad Certificadora escogida. Esta debe responder con una renovación del certificado propio.
3. Importe el certificado renovado en el almacén de llaves con el comando del paso 10 de la generación de almacenes de llaves:

```
keytool -import -alias user -file CertificadoRenovado.cer -storetype PKCS12-DEF -keystore user.p12 -storepass changeit
```

Recomendaciones relacionadas a Almacenes de Llaves

Tenga en cuenta las siguientes recomendaciones al crear y usar almacenes de llaves:

- Los almacenes de llaves son **CONFIDENCIALES** porque contienen llaves privadas y públicas, NO deben ser enviados por correo electrónico ni se deben poner en lugares inseguros (carpetas compartidas, computadores públicos, etc.).
- Los archivos de certificados, CSR y CRL contienen únicamente llaves públicas, por lo que pueden ser transferidos o compartidos sin ningún problema.
- Se recomienda que los almacenes de llaves sean generados en la máquina en que se van a utilizar (donde esté instalado Crypto Vault). Sin embargo las implementaciones de keytool para algunas versiones de sistemas operativos tienen problemas al importar los certificados, por lo cual habría que generar el almacén de llaves en una máquina externa y después moverlos a la máquina en donde se van a utilizar.
- Es recomendable tener copias de respaldo de los almacenes de llaves, que estén guardadas en un lugar seguro. Esto facilita una potencial reinstalación en caso de alguna falla.
- Es buena práctica tener presentes las fechas de vencimiento de los certificados, y comenzar la gestión de petición de certificados nuevos con suficiente anterioridad, ya que este proceso puede tardar incluso algunos días.
- Si agrega varias Autoridades Certificadoras de confianza en su almacén de llaves para poder aceptar diferentes certificados, debe tener en cuenta que se deben mantener actualizadas todas las listas de certificados revocados (CRL) para cada autoridad certificadora.