1.Sumario

En estos tiempos en los que las comunicaciones y las tecnologías de la información están siendo cada vez más importantes, combatir a los virus, hackers, escuchas y fraudes electrónicos, provoca que la seguridad tenga que tomarse muy en cuenta. Con el objetivo de proporcionar seguridad, existe la criptografía como herramienta principal y más importante con diferencia. A grandes rasgos, el uso de la criptografía ayuda a evitar el uso fraudulento de sistemas, a proteger información confidencial o importante, permitir comunicaciones seguras y posibilitar el comercio electrónico. Puede obtenerse más información sobre este tema consultando la dirección http://jcef.sourceforge.net/doc/introsecurity.pdf.

El objetivo primordial de este proyecto es aprender a utilizar los principales mecanismos criptográficos. Estos mecanismos son los de protección y autentificación. Como objetivo secundario, inicialmente se había planteado desarrollar un conjunto de programas representativos de las aplicaciones de la criptografía, pero tras el estudio preliminar, dicho objetivo fue eliminado y suplantado por otro mucho más interesante y más novedoso: desarrollar una librería criptográfica potente y sobre todo de fácil uso.

Siendo más precisos, la criptografía permite asegurar un objeto convirtiéndolo en otro objeto incomprensible y/o autentificable (un objeto autentificable es aquel en el que se puede comprobar si su origen es auténtico y/o sus propiedades son auténticas). Además, también permite obtener el objeto asegurado a partir de su versión segura. Para realizar estas transformaciones se utilizan algoritmos y parámetros criptográficos concretos; y es en la seguridad de estos elementos donde se basa la seguridad de la criptografía. Por otro lado, cabe destacar que un objeto puede ser cualquier cosa: información, recursos, datos, mensajes, ficheros, un objeto ya seguro incluso, etc... Más información sobre este tema en http://jcef.sourceforge.net/doc/oocryptography.pdf.

El resultado de este proyecto ha sido un conjunto de librerías Java. Entre ellas destaca la librería llamada JCEF (Java Cryptographic Extension Framework) cuya página web oficial y la de este proyecto se encuentra hospedada en el mayor repositorio de proyectos software de código abierto existente llamado SourceForge.net; la dirección de esta web es http://jcef.sourceforge.net.

Antes de la existencia de este proyecto fin de carrera ya existían unas librerías Java llamadas JCA y JCE encargadas de permitir a sus usuarios utilizar algoritmos criptográficos y algo más; pero su uso resulta y resultaba demasiado complicado. Es por ello por lo que se decidió realizar una librería Java para programar fácilmente sistemas que hagan uso de algoritmos criptográficos; en lugar de desarrollar un conjunto de programas representativo de las aplicaciones de la criptografía.

JCEF se soporta actualmente sobre JCA y JCE aunque no depende de dichas librerías. Eso sí, utilizar JCEF es mucho más sencillo que emplear JCA y JCE.

El cliente de esta nueva librería JCEF sería el desarrollador de software que necesita usar sistemas criptográficos de seguridad con suma facilidad sin las complejidades inherentes a JCA y JCE. JCEF es útil para programadores expertos en JCA y JCE y en especial para los nuevos programadores de sistemas criptográficos en Java.

Entrando en detalles, JCEF es un marco de trabajo para programadores que deseen desarrollar sistemas de seguridad basados en la criptografía a través del empleo de algoritmos criptográficos. Básicamente, JCEF define una especificación que permite definir y utilizar algoritmos criptográficos de protección y autentificación.

Para definir nuevos algoritmos criptográficos que cumplan la especificación JCEF se han implementado cinco especificaciones JCEF:

- Una para definir nuevos algoritmos criptográficos de una forma mucho más fácil de lo que permite JCA y JCE.
- Otra implementación permite adaptar fácilmente los algoritmos existentes con especificación JCA y JCE a la especificación JCEF.
- Una tercera implementación permite que los algoritmos implementados mediante la especificación JCEF sean compatibles con la especificación JCA y JCE.
- Para generar, convertir y comprobar automáticamente parámetros se ha realizado una cuarta implementación JCEF.
- Se ha realizado una última y quinta implementación JCEF para poder definir algoritmos criptográficos cuya implementación soporte las cuatro implementaciones anteriores como si fuera una única implementación, es decir, para definir algoritmos criptográficos de nueva implementación JCEF o adaptados de JCA y JCE y que además cumplan la especificación JCA y JCE y generen, conviertan y comprueben automáticamente parámetros que intervengan en dichos algoritmos.

A continuación puede verse una imagen que representa la posición del proyecto en el mundo Java de la criptografía:

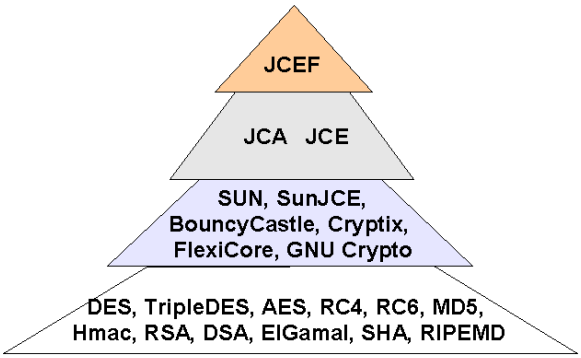

Ilustración 1: JCEF acerca la criptografía al usuario

JCEF mejora enormemente a JCA y JCE. Algunas de las deficiencias de JCA y JCE que han sido mejoradas son: JCE es muy difícil de utilizar, dificultando su uso y aprendizaje al proporcionar mecanismos heterogéneos y difíciles de emplear, requerir al usuario más conocimiento técnico del necesario y más líneas de código de las realmente necesarias para usuarios inexpertos con JCA y JCE. En definitiva, utilizar JCE requiere un gran esfuerzo en tiempo y dedicación al usuario del mismo.

Sin embargo, JCEF posee valores añadidos que hacen que valga la pena su uso. Algunos de estos valores añadidos más importantes son:

- Es amigable y fácil de utilizar al simplificar, eliminar y automatizar muchas operaciones de bajo nivel.
- Se aprende de forma muy sencilla y no requiere demasiado tiempo de aprendizaje ni grandes conocimientos técnicos.
- Y por si fuera poco viene con grandes posibilidades de extensión o mejora para el futuro y pruebas para comprobar el correcto funcionamiento de los algoritmos criptográficos y sus implementaciones.
- Permite además definir algoritmos criptográficos muy fácilmente.
- Además trae consigo un paquete inicial de 64 algoritmos criptográficos de todo tipo (algoritmos de protección asimétrica, protección simétrica de bloques y de flujo, protección basada en contraseña, algoritmos de autentificación mediante huellas digitales, sellos digitales, sellos digitales basados en contraseña, firmas digitales y generadores de fuentes de datos) y unas grandes colecciones de otros algoritmos de las mismas características, aunque estas colecciones no han sido probadas por razones de tiempo, como resultado de la traducción de todos los proveedores de algoritmos criptográficos Java encontrados.
- Y por último y más importante, JCEF permite asegurar objetos mediante la construcción de objetos seguros de una forma muy sencilla utilizando para ello cualquier tipo de algoritmo criptográfico y posibilitando obviamente recuperar estos objetos asegurados a partir de sus versiones seguras.

Por desgracia, JCEF no es perfecto y todavía puede y debe seguir creciendo. Algunas de las posibles mejoras que se le pueden hacer son:

- Implementar flujos de entrada/salida seguros para cualquier algoritmo criptográfico.
- Añadir soporte de almacenes seguros para cualquier tipo de objetos incluyendo claves y certificados digitales.
- Incluir protocolos de intercambio de claves.
- Gestionar certificados digitales.
- Configurar completamente los modos de operación y esquemas de relleno de los algoritmos criptográficos que los tengan.
- Realizar pruebas de implementación a los algoritmos mediante vectores de test con el objetivo de garantizar la correcta implementación de los algoritmos y aumentar el nivel de confianza de los usuarios de este proyecto.

- Corregir aquellos algoritmos que no pasen las pruebas de funcionalidad.
- Implementar algoritmos nuevos no implementados hasta ahora en Java.

Como un ejemplo del valor añadido de este proyecto, observe el código siguiente donde se muestra cómo se asegura un objeto e inmediatamente después se recupera el mismo; y todo ello de una forma super sencilla:

```
Object object = new String("my object");
CryptoAlgorithm secureAlgorithm = new AES_BlockSymmetricProtectionRREXKY();
SecureObject secureObject = new SecureObject(object, secureAlgorithm);
object = (String)secureObject.getObject(secureAlgorithm);
```

Tabla 1: Pequeño ejemplo de uno de los valores añadidos de JCEF

Si lo desea, también puede consultar la página web oficial del proyecto http://jcef.sourceforge.net y el recurso http://jcef.sourceforge.net/doc/project.pdf.

Sobre el diseño de este proyecto se puede decir que ha sido descompuesto en numerosas librerías. Además, cada librería tiene asociada otra que contiene las pruebas para comprobar el correcto funcionamiento de dicha librería. Por si fuera poco, todas las librerías han sido documentadas. Son un total de 38 librerías.

Las librerías de este proyecto son "JCEF" como librería principal, "JCEF Addons" como librería para extensiones futuras y el resto de librerías son proveedores criptográficos que cumplen al menos la especificación JCEF y librerías de pruebas para comprobar el correcto funcionamiento de todas las librerías. Concretamente se prueba el correcto funcionamiento de todos los métodos de todas las clases que componen el proyecto y se realizan pruebas de funcionalidad a cada algoritmo.

El diseño de este proyecto sigue los principios básicos de los Java Beans y otra consideración a tener en cuenta es que se ha intentado al máximo que el número de líneas de código por método fuera el menor posible.

Puede obtenerse más información sobre este tema en la web http://jcef.sourceforge.net y en la dirección http://jcef.sourceforge.net/doc/project.pdf que contiene información detallada del proyecto.

Además, se proporcionan una serie de distribuciones para que todo el mundo tenga acceso a este proyecto. Estas distribuciones se podrán encontrar en http://jcef.sourceforge.net y http://sourceforge.net/projects/jcef. Existen 6 distribuciones: una con lo básico para usuarios, otra para desarrolladores con todo lo necesario, otra con todo pero sólo para usuarios, otra que contiene todo el código fuente, otra distribución con toda la documentación del proyecto y por último una distribución que contiene la página web.

La página web del proyecto http://jcef.sourceforge.net permite el acceso del mundo al proyecto de forma libre y gratuita. En esta web se podrá encontrar todo lo relacionado con el proyecto: documentación, código fuente, presentación, las últimas noticias y muchos otros enlaces.

El manual de usuario de este proyecto, sin considerar este documento, se encuentra en http://jcef.sourceforge.net/api/org/rrexky/security/crypto/jcef/UserGuide.html Para obtener más información es importante consultar la página web de este proyecto http://jcef.sourceforge.net y la página principal online de la documentación de JCEF que se encuentra en http://jcef.sourceforge.net/api/index.html.

La dimensión del proyecto es considerable, ya que de ahí el gran número de librerías, debido principalmente a que se ha querido proporcionar el mayor número de proveedores y algoritmos criptográficos posible realizando adaptaciones de proveedores y algoritmos criptográficos ya existentes que cumplan con la especificación JCA y JCE u otras.

Para que el lector pueda hacerse una idea de la dimensión de este proyecto, basta con decir que este proyecto contiene 3142 clases, 315 campos, 2914 métodos, 73959 líneas de código y 22941 líneas de comentarios.

Otra métrica para imaginarse el coste de este proyecto es el tiempo de desarrollo y realización del mismo; el cual se estima entre un mínimo de 2000 horas de trabajo y un máximo de 3000 horas. Este proyecto ha tenido pocas pausas durante su desarrollo, el cual comenzó a finales de febrero de 2005 y terminó a finales de mayo de 2006, es decir, más de un año de desarrollo.

Sobre el coste económico del proyecto hay que decir que su coste económico es cero debido a que para el desarrollo del mismo se han utilizado únicamente recursos y programas totalmente gratuitos de licencia Open Source o Freeware.

También se ha documentado un conjunto de posibles futuros proyectos fin de carrera que podrán realizarse como continuación de éste para mejorarlo y ampliarlo. La descripción de estos futuros proyectos se ha realizado mediante el formato exigido a una propuesta de proyecto de fin de carrera por la Universidad de Las Palmas de Gran Canaria, con el objetivo de facilitar la elección de estos proyectos por alumnos y tutores de dicha universidad y dar así continuidad a este proyecto. Los 7 futuros proyectos propuestos se titulan: «Ampliaciones de JCEF», «Pruebas sobre algoritmos JCEF», «Certificados Digitales con JCEF», «Archivos seguros con JCEF», «Proveedor Criptográfico JCEF», «Almacén de objetos seguros con JCEF», «Metaimplementación de "Aplicaciones Criptográficas Java"».

Se puede obtener más información sobre el proyecto en http://jcef.sourceforge.net y sobre los futuros proyectos solamente en la dirección web http://jcef.sourceforge.net/doc/futureprojects.pdf.

Para la realización del proyecto se han utilizado numerosos recursos totalmente gratuitos de licencia Open Source o Freeware, tales como bibliografía, recursos web, programas, etc... A continuación se muestra un sumario de los recursos empleados para el desarrollo del proyecto:

- Para obtener información: Bibliografía y numerosas páginas y recursos web.
- Para dar valor al proyecto: Se han utilizado todos los proveedores criptográficos existentes de algoritmos criptográficos para Java que se han encontrado. Un total de 16 proveedores.
- Como apoyo al proyecto: se han reutilizado algunas herramientas y librerías Java.
- Para el desarrollo del proyecto: Se ha utilizado principalmente el entorno de desarrollo para Java *"Eclipse"* y también algunos de sus *plugins*.

- Como herramientas de documentación: se utilizó la suite de ofimática *"OpenOffice"*, *la utilidad de generación de documentación para código Java "Javadoc"*, programa de diseño UML llamado *"ArgoUML"*, y otras llamadas *"PDFCreator" y "Java2HTML"*.
- Para diseñar la página web del proyecto: *"Nvu"* y una plantilla para dicha página web.
- Como navegador web para Internet: *"Mozilla Firefox"*.
- Para publicar el proyecto en Internet: *"WinSCP"*, *"FileZilla"* y *"SourceForge.net"*.
- Como lenguajes de desarrollo: *"Java"*, *"HTML"*, *"XML"*, *"UML"* y *"CSS"*.
- Y otros recursos como *"7-zip"*, *"AlZip"*, "CDBurnerXP Pro", *"JEdit"*, *"Notepad++"*, *"GIMP"*, *"Irfanview"*, *"Paint.NET"*, *"Directrices de desarrollo"*, *"NotesHolder Lite"*, *"JSmooth"* y *"Cobian Backup"*.

Para mayor información sobre este proyecto se puede consultar la web http://jcef.sourceforge.net y también puede encontrar información completa sobre los recursos utilizados en http://jcef.sourceforge.net/doc/resources.pdf.

También se podrá encontrar en este documento una sección llamada «Preguntas frecuentes» que incluye las preguntas más frecuentes que el lector de este proyecto puede hacerse.

Quizás, la pregunta más importante podría ser: "¿Por qué no se ha desarrollado un conjunto de aplicaciones?" Es decir, "¿porque no se ha cumplido uno de los dos objetivos de este proyecto?". Respondiendo de forma corta, se podría podría decir que: "Porque tras realizar el análisis surgió una idea mejor". La respuesta más larga sería: "La razón de existencia del segundo objetivo previsto «Mostrar aplicaciones de las técnicas criptográficas mediante software desarrollado en el proyecto» simplemente era el de probar que realmente se había alcanzado el primer objetivo: «Aprender a utilizar técnicas criptográficas». Tras finalizar el estudio preliminar, se consideró que era mucho mejor desarrollar algo nuevo, antes que realizar un conjunto de aplicaciones representativas de la criptografía, lo cual no es nada novedoso. Además, ya existen herramientas muy buenas como "CrypTool" (http://www.cryptool.com/), TrueCrypt (http://www.truecrypt.org/) y AxCrypt (http://axcrypt.sourceforge.net) de código abierto y otras freeware tales como "EncryptOnClick" y "FingerPrint" (http://www.2brightsparks.com/).".

También son interesantes las siguientes cuestiones que se responderán aquí de forma breve:

- ¿Cuál es el tamaño del proyecto? Su tamaño es bastante grande ya que está compuesto por 3142 clases, 315 campos, 2914 métodos, 73959 líneas de código y 22941 líneas de comentarios.
- ¿Cuál ha sido el coste del proyecto? Económico ninguno al utilizar herramientas gratuitas pero el coste de tiempo se estima entre 2000 y 3000 horas; más de un año de trabajo sin pausa.
- ¿Qué metodología se ha utilizado? Se ha utilizado la clásica (análisis, diseño, implementación y pruebas) con retroalimentaciones.

Para mayor información sobre el proyecto puede consultar la web http://jcef.sourceforge.net y si desea consultar más preguntas frecuentes puede acceder directamente a ellas dirigiéndose a la dirección http://jcef.sourceforge.net/doc/faqs.pdf.

En resumidas cuentas, las conclusiones más importantes sobre el proyecto en sí son los principales aspectos valorables positivamente:

1. Haber desarrollado un trabajo novedoso en lugar de lo que se tenía previsto.
2. El uso 100% de herramientas Open Source o Freeware.
3. Haber aplicado el conocimiento de varias áreas: Ingeniería del software, Gráficos, Programación, Ofimática, Programación web, etc...
4. El enfoque orientado a objetos de la criptografía.
5. Publicación del proyecto en una página web (http://jcef.sourceforge.net).
6. El diseño de imágenes propias originales.
7. Haber sido capaz de trabajar en el proyecto sin prisas y sin pausas durante algo más de un año.
8. Las propuestas realizadas sobre posibles futuros proyectos fin de carrera.
9. La sección de preguntas frecuentes como ayuda adicional.

También es importante conocer algunos puntos que podrían considerarse como negativos. Éstos son los siguientes:

1. El tiempo empleado en el proyecto podría considerarse excesivo.
2. No haber desarrollado exactamente lo que estaba previsto desde un principio.

En las siguientes secciones se podrá ver información más detallada sobre el proyecto, comenzando por un par de capítulos de introducción tanto a la seguridad como a la criptografía, luego otro capítulo en el que se habla del proyecto en sí; seguidamente se continúa con una sección titulada “Futuros proyectos”, para posteriormente continuar con el capítulo donde se detallan los recursos utilizados en este proyecto; y finalmente un capítulo donde se colocan las preguntas frecuentes que puede hacerse el lector del proyecto.

Recuerde el lector que toda la información sobre este proyecto y más se puede encontrar en la dirección web http://jcef.sourceforge.net.