REPÚBLICA DE CUBA
Ministerio del Interior

Universidad de las Ciencias Informáticas

Título: "Recuperación residual de ficheros JPEG y PSD a través de sus parámetros estructurales invariantes."

Trabajo de diploma para optar por el título de Ingeniero en Ciencias Informáticas.

Autores
Félix Daniel Batista Diñeiro
José Ramón Sera Concepción

Tutores
Lic. Ernesto Pico Abello
Lic. Ramsés Dupuy Mercader

"Todavía no se han levantado las vallas que le digan al talento: ¡De aquí no pasarás!"

Albert Einstein

DECLARACIÓN DE AUTORÍA

Declaramos ser autores de la presente tesis, y reconocemos a la Universidad de las Ciencias Informáticas y a la Dirección Informática y Comunicaciones del Ministerio del Interior los derechos patrimoniales de la misma, con carácter exclusivo.

Para que así conste, firmo la presente a los ____ días del mes de _____ del año _____.

_____	_____
José Ramón Sera Concepción	Félix Daniel Batista Diñeiro
Firma del Autor	**Firma del Autor**
_____	_____
My. Lic. Ernesto Pico Abello	Cap. Lic. Ramsés Dupuy Mercader
Firma del Tutor	**Firma del Tutor**

DATOS DE CONTACTO

Tutor: My. Lic. Ernesto Pico Abello

Licenciado en Física de la Universidad de La Habana (1992), Especialista de postgrado en Informática Operativa (2006), ha cursado postgrados de computación, seguridad y programación. Primer Perito Criminalista con 15 años de experiencia profesional. Ha trabajado el tema de la computación forense durante los últimos 9 años y participado en cursos y eventos internacionales. Cuenta con 16 años de graduado.

Tutor: Cap. Lic. Ramsés Dupuy Mercader

Licenciado en Derecho del Instituto Superior del MININT (2001), cursa la especialidad de postgrado en Informática Operativa, ha cursado postgrados de computación, seguridad y programación. Primer Perito Criminalista con 16 años de experiencia profesional. Ha trabajado el tema de la computación forense durante los últimos 7 años y participado en cursos y eventos internacionales. Cuenta con 7 años de graduado.

AGRADECIMIENTOS

En estas primeras líneas va mi agradecimiento a Martí, el primero de todos mis maestros, a ti Fidel por tu coraje y ejemplo, a ti Raúl por tu fidelidad a la causa y tu entereza. A los hombres y mujeres que me entregaron e hicieron posible la Cuba de hoy. A Céspedes, Agramonte, Aguilera, Calixto, Maceo, Mariana, Gómez y Panchito. A Mella, Frank, Abel y José Antonio. A las eternas Vilma, Celia y Haydee.

A mis tutores Pico y Ransés, por toda su ayuda y por lo que de ellos aprendí.

A mis amigos Echenique, Félix, Elier, Dani, José Armando por estar siempre a mi lado y acompañarme en mis luchas. Por gratos recuerdos de estos años que no se olvidan.

A mis verdaderos maestros, a los que no solo me enseñaron desde la instrucción, sino con la fuerza de su ejemplo. A Irma, Abel, Alonso, Molina y Héctor. A Hordy, Odalys Batista, Magalis Soberats, Zonia, Armando Mengana y Leonardo mi profesor de Matemáticas.

A Nenita y Fernando, por su cariño y su apoyo, por sus consejos y sus críticas, por su tiempo. Por haberme permitido aprender tanto de su experiencia y sus muchas virtudes. Por ser más que jefes, siempre amigos, tíos, padres y todo rol que hizo falta desempeñar en estos tres años últimos años.

A la UCI, mi universidad del presente que camina hacia el futuro.

Al Ministerio del Interior, por mantener prendida la llama y permitirme estar cerca. Por todo lo que aprendí de sus virtuosos hombres y mujeres.

José Ramón

Siempre estaré agradecido por haber tenido la dicha de nacer en esta tierra, por haber contado con un Fidel, una Revolución y una historia que siempre han sabido señalar el camino correcto de manera magistral.

En segundo lugar, deseo agradecer a mis abuelos, arquitectos principales de una obra que no pudieron ejecutar por sí mismos, pero que han podido ver materializada por sus hijos y nieto.

A mis padres, quienes han sido mi principal soporte sentimental, ideológico y financiero durante los últimos 23 años de mi vida.

A mi novia – esposa, el otro cimiento monolítico que ha sostenido mi peso durante ya casi una década, y quien nunca ha dejado de confiar en mí ni un solo instante.

A mis suegros, por el apoyo emocional – gastronómico que me han suministrado siempre.

A mis tutores, porque supieron inyectarme la pasión por ese mundo en base 16, desconocido casi por completo para mí hasta hace unos 7 u 8 meses.

Tampoco puedo dejar de mencionar a mis amigos, porque me permitieron tener mucho más que el leopardo. Sin ellos el camino hubiera sido incomparablemente más difícil. Agradezco especialmente a Fernando, porque sé que hizo todo cuanto estuvo a su alcance para evitar que yo me graduara, aunque felizmente no lo consiguió.

A José Ramón, Alejandro y todos los muchachos(as) del Estado Mayor, quienes me hicieron sentir uno más entre ellos desde el primer día.

A todos aquellos quienes, de una forma u otra, han contribuido desde la escuela primaria a hacer de mí una mejor persona, porque también gracias a ellos he podido llegar hasta aquí.

Félix Daniel

DEDICATORIA

A Cuba.

A mis compañeros de luchas de todos los tiempos.

A mi Estado Mayor al que jamás podré olvidar y dentro de él a los que siempre me acompañaron en los más puros sueños de está efímera juventud. A Ale, Hamler, Fernando, Roberto, Eliska, Yadira, Monika (Cacha), May, Niurvis, Angel, Graña, Noelito, Asley, Calé, Yoandry, Yosley y Camilito.

A mi familia toda y dentro de ella a mis tíos Sandra y Luis, que sin lugar a dudas merecen el reconocimiento y el afecto que estas líneas encierran. A Luisi y Julito.

A mis hermanas y sobrinos. A mis padres queridos por sus desvelos y su entrega incondicional al afecto que desde antes de nacer ya me tenían.

Especialmente a la heroína de mi historia, a mi Mariana Grajales, a mi Doña Leonor, a mi Lady Laura. A esa a quien le debo todo cuanto soy y seré, a quien nunca podré pagar las extraordinarias demostraciones de cariño, moral, desinterés, entrega, amor, sacrificio, honestidad, perseverancia y humildad, que he recibido de ella. A mi adorada madre, razón fundamental de mi esfuerzo y de mis días.

Finalmente la dedico a ti…

José Ramón

Deseo dedicar este trabajo, en primer lugar, al Ministerio del Interior, el cual me ha dado la oportunidad de ser más útil desde otra trinchera.

A mis padres, abuelos y novia, mis más viejos y mejores amigos, porque a pesar de la distancia física, ni una sola noche he dejado de sentir su calor sobre la almohada.

A Gerardo, René, Antonio, Ramón y Fernando, colosos de la dignidad, quienes atraviesan ahora un momento crucial en la defensa de su caso ante la Corte Suprema, y gracias a los cuales muchos círculos infantiles en Cuba –como las demás conquistas de nuestro pueblo- permanecen aun en pie.

Al ejemplo inmortal de Kuznetsov, Merésiev, Panfílov y todos los héroes, conocidos y anónimos, que entregaron sus vidas para que las banderas del comunismo ondearan victoriosas sobre los restos calcinados del fascismo.

A la gloriosa historia de Cuba…

Félix Daniel

RESUMEN

La volatilidad que caracteriza a la información digital, determinada fundamentalmente por la relativa facilidad con que puede corromperse o alterarse, ya sea de manera accidental o intencional, ha generado la necesidad de buscar formas para recuperar datos que se hayan extraviado en un soporte determinado. Ello ha convertido a la recuperación de información en entornos digitales en una disciplina altamente valorada.

El Ministerio del Interior, como parte del proceso de modernización tecnológica que lleva a cabo, requiere de herramientas cada vez más eficaces para su enfrentamiento constante a las nuevas tendencias delictivas. Dentro de ese amplio espectro, la recuperación de información ocupa un lugar destacado.

El hecho de que un archivo permanezca físicamente en un dispositivo aun después de su eliminación del sistema de ficheros abre una puerta valiosísima para la localización de ficheros o restos de ellos. En esa dirección está orientado el presente trabajo: el estudio de las estructuras que caracterizan a 2 ficheros de interés criminalístico: las imágenes JPEG y los ficheros de Adobe Photoshop.

El resultado de esta investigación es la construcción de un script diseñado en el entorno de desarrollo de la herramienta de análisis forense EnCase Enterprise, que permite identificar metadatos y otras estructuras internas propias de esos 2 formatos de ficheros.

PALABRAS CLAVE:
Recuperación de información, metadatos, Informática Forense, EnCase.

ÍNDICE DE CONTENIDOS

ÍNDICE DE FIGURAS

INTRODUCCIÓN

Con el surgimiento y la posterior globalización de las tecnologías de la informática y las comunicaciones, la aparición de Internet y con él, el inicio de la era (edad) de la información como un importante hito en el desarrollo de la humanidad, se crearon también las condiciones para el surgimiento, incremento y perfeccionamiento de procederes y conductas deshonestas, alejadas de la moral y la legalidad.

La computación y la era de la red de redes provocaron la proliferación de nuevos tipos de delitos y formas nuevas de entorpecer el esclarecimiento de los ya conocidos. Surgió entonces la necesidad mundial de encarar, en los planos operativo y legal del enfrentamiento, la organización de una disciplina capaz de hacer frente a estas tendencias.

Cuba, como parte del mundo, no está exenta de ese tipo de conductas. El actual y creciente proceso de informatización de la sociedad cubana ha reclamado por parte de los órganos del Ministerio del Interior la estructuración coherente de un proceso de modernización tecnológica que le permitan cumplir con eficacia su misión dentro de la sociedad cubana: la garantía del orden interior y de la seguridad del estado.

Con el apoyo de herramientas forenses y basándose en la experiencia acumulada los criminalistas cubanos se enfrentan a diario a los nuevos desafíos que le imponen las tecnologías de la informática y las comunicaciones. Aún así muchos de estos procesos en la actualidad son realizados manualmente, muy a pesar de que en los análisis periciales, con independencia del caso en cuestión, se observa cierta regularidad en los procedimientos a emplear en la investigación.

Se conoce, fruto del trabajo de un grupo de especialistas del MININT, que en algunos casos los programas y herramientas forenses no están en condiciones de brindar un resultado totalmente acabado en la recopilación de evidencias digitales, puesto que los algoritmos que utilizan, no incorporan en sus procesos de búsqueda a la totalidad de las estructuras, de determinados ficheros informáticos.

La Sección de Informática Forense de la División de Criminalística del Ministerio del Interior no cuenta con un script que le permita recuperar ficheros JPEG y PSD eliminados o residuos de ellos, a través de la localización de sus parámetros estructurales invariantes con excepción de sus cabeceras y terminaciones.

Es por eso que en la búsqueda de soluciones a esta problemática se plantea el siguiente **problema científico**:
¿Cómo recuperar ficheros JPEG y PSD eliminados o residuos de ellos, a través de la localización de sus parámetros estructurales invariantes con excepción de sus cabeceras y terminaciones?

El **objeto de estudio** del presente trabajo de diploma son las herramientas informático-forenses. Siendo su **campo de acción** el desarrollo de herramientas para la Arqueología de datos en el MININT.

Como **objetivos investigativos** se encuentran:

- Determinar los parámetros estructurales invariantes en ficheros JPEG y PSD.
- Desarrollar un script que permita recuperar ficheros JPEG y PSD eliminados o residuos de ellos, a través de la localización de sus parámetros estructurales invariantes con excepción de sus cabeceras y terminaciones.

La **idea a defender** de este trabajo es que con el desarrollo de un script que le permita recuperar ficheros JPEG y PSD eliminados o residuos de ellos, a través de la localización de sus parámetros estructurales invariantes con excepción de sus cabeceras y terminaciones, la Sección de Informática Forense de la División de Criminalística elevará su capacidad y efectividad en el esclarecimiento de los hechos delictivos donde aparezcan comprometidos ficheros de estos formatos.

Las Tareas propuestas para dar cumplimiento a los objetivos:

1. Estudio del estado del arte de las herramientas que se emplean en los análisis forenses.
2. Asimilación de las experiencias y tecnologías del entorno de trabajo de la Sección de Informática Forense de la División de Criminalística del MININT.

3. Sistematización de las normas y estándares internacionales que rigen las estructuras físicas de los ficheros JPEG y PSD.
4. Caracterización de los parámetros invariantes en dichas estructuras físicas.
5. Implementación del script para la recuperación de ficheros JPEG y PSD o fragmentos de ellos, mediante la localización de sus parámetros estructurales invariantes, con excepción de sus cabeceras y terminaciones.
6. Validación de los resultados obtenidos con la implementación del script.

Los métodos teóricos que hemos utilizado son:

Método Análisis histórico - lógico: Utilizado para sistematizar la evolución y desarrollo de la Informática Forense y las diferentes herramientas con que cuenta la disciplina a nivel internacional. También en el estudio de la evolución de los estándares y estructuras de los ficheros informáticos.

Método Analítico - Sintético: Empleado para procesar toda la información que se consultó en el transcurso de esta investigación y que nos sirvió para arribar a conclusiones en la determinación de las actividades que debían estar presentes en el desarrollo de este trabajo de diploma.

Método Inductivo - deductivo: Utilizado para sistematizar las estructuras físicas de los ficheros informáticos, y poder establecer regularidades en su conformación y en los métodos de manipulación.

Los métodos empíricos que hemos utilizado son:

Método Experimento y el de la Observación Científica: Empleados para evaluar los resultados de la solución propuesta como conclusión de esta tesis de grado.

CAPÍTULO 1
MARCO TEÓRICO

1.1 - Introducción a la Informática Forense

La necesidad de la Informática Forense se desprende directamente de una serie de sucesos que han afectado a la globalizada sociedad de la información de fines del siglo XX y principios del XXI.

Caracterizada por la constante aparición de vulnerabilidades en los sistemas de información, esta etapa ofreció un escenario perfecto para la proliferación de tendencias relacionadas con los intrusos informáticos. Los mismos poseen diferentes motivaciones, alcances y estrategias que desconciertan a analistas, expertos y cuerpos de investigación, pues sus modalidades de comportamiento y ataque varían de un caso a otro.

Este escenario ha condicionado la especialización de la criminalística hacia el estudio y el análisis en ambientes digitales de lugares que han sido objeto de acciones criminales. Por ello se hizo preciso establecer un conjunto de herramientas y procedimientos que permitiera establecer en los medios informáticos la presencia de una evidencia digital que corroborara la ocurrencia de un hecho delictivo.

Surgió entonces la informática forense como una disciplina auxiliar de la justicia moderna para enfrentar los desafíos impuestos por los intrusos informáticos, y garantizar las prácticas adecuadas que establecen las leyes en la instrucción de los procesos penales.

Asumimos como válida la definición que da el FBI al conceptualizar la informática (o computación) forense como *"la ciencia de adquirir, preservar, obtener y presentar datos que han sido procesados electrónicamente y guardados en un medio computacional*[1]*.* Es también conocida como: Cómputo Forense, Computación Forense, Análisis o Examinación Forense Digital, Informática Forense así como *Digital Forensics* en idioma inglés.

[1] Óscar López , Haver Amaya, Ricardo León y Coautora: Beatriz Acosta, "INFORMÁTICA FORENSE : GENERALIDADES, ASPECTOS TÉCNICOS Y HERRAMIENTAS," http://www.criminalistaenred.com.ar/Informatica_F.html.

En el análisis de la documentación, se ha podido constatar que independientemente de la definición, la opinión internacional coincide en señalar que la disciplina debe procurar la identificación, preservación, extracción, análisis, interpretación, documentación y presentación de las pruebas conservando la confiabilidad e integridad de la evidencia.

De acuerdo a la bibliografía consultada, la Informática Forense es una ciencia relativamente nueva y no existen aun estándares universalmente aceptados, aunque sí hay varios proyectos en desarrollo, como el C4PDF (Código de Prácticas para Informática Forense), el *Open Source Computer Forensics Manual* (Manual Público de Computación Forense) y el *Training Standards and Knowledge Skills and Habilitees (Estándares para el entrenamiento de habilidades)*, de la *International Organization on Computer Evidence*.

Existen diferentes versiones sobre la fecha de surgimiento de las técnicas forenses en la informática, pero la mayoría de los estudiosos coinciden en señalar la década de los 80 como el momento histórico de su nacimiento. Por lo general se reconoce a Dan Farmer y Wietse Venema, creadores del Forensics Toolkit, como los pioneros de la informática forense.

1.2- Arqueología de Datos.

La Arqueología de Datos es la derivación de la Computación Forense, que se encarga fundamentalmente del trabajo con datos previamente eliminados, o residuos de ellos que persistan, aún parcialmente sobrescritos, en los dispositivos electrónicos-computacionales de almacenamiento.

En el desarrollo del presente trabajo de diploma, sus autores ante la poca información especializada se asumió la conceptualización de esta arista de la Informática Forense, definiéndola como la rama que se encarga de la recuperación de ficheros o fragmentos de ellos, que persisten en el espacio no asignado de un medio computacional de almacenamiento luego de ser eliminados.

1.3- Evidencia digital

Reconocida como el elemento principal dentro de una investigación una evidencia de acuerdo con la Revista colombiana de Cirugía no es más que una *"Certeza clara, manifiesta y tan perceptible, que nadie puede racionalmente dudar de ella".*[2]

Por tanto en el ámbito computacional forense una Evidencia Digital de acuerdo con el HB: 171 2003 *Guidelines for the Management of IT Evidence*, se asume que es "cualquier información, que sujeta a una intervención humana u otra semejante, ha sido extraída de un medio informático", entiéndase también cualquier registro generado por o almacenado en un sistema computacional que pueda ser utilizado como evidencia en un proceso legal.

El Dr. Luis Rafael Moreno González quien es Miembro Fundador, Ex presidente y Presidente Honorario Vitalicio de la Academia Mexicana de Criminalística menciona que: *"El manejo inadecuado de la evidencia física conduce a su contaminación, deterioro o destrucción, siendo esta última la causa más frecuente que impide su ulterior examen en el laboratorio. Por esta razón, cuando llegue el Momento de proceder a su levantamiento se realizará con la debida técnica a fin de evitar tan lamentables consecuencias."*[3]

El párrafo anterior advierte sobre la exigente labor que se precisa en el trabajo de los investigadores, tanto en procedimientos como en técnicas y herramientas tecnológicas que permitan obtener, custodiar, revisar, analizar y presentar la evidencia presente en una escena del delito.

Es por tanto necesario poseer un conocimiento detallado y actualizado de las normas y regulaciones legales asociadas con las pruebas y el derecho procesal, así como de las técnicas y procedimientos que permitan mantener la confiabilidad de los datos recogidos y la integridad de los mismos.

De acuerdo a la bibliografía consultada, se puede conocer que a nivel mundial el trabajo con la evidencia incluye aunque no se limita a las siguientes fases: (**Tomado "Análisis Forense Digital de Alberto Hornero"**.)

---

[2] Dr.Joaquín Silva, "CIRUGÍA,Glosario*, "http://encolombia.com/cirugia14399_glosario47.htm.

[3] MORENO GONZÁLEZ, Luis. (1990). *Manual de Introducción a la Criminalística*, México: Porrúa.

Identificación de la evidencia: [4] "La tarea inicial de una investigación es identificar la evidencia que es necesaria para el caso. Sin evidencia no existe mucho más que una opinión. Es obvio que cada caso es diferente, así es que se necesitan diferentes tipos de evidencia en base al caso. Conociendo que la evidencia necesaria es una parte integral de una investigación satisfactoria."

"La regla de oro es tomar todo. Desafortunadamente, existen temas legales y logísticos con este enfoque. Siendo más realistas, se debe tomar todo y cualquier cosa que pueda estar remotamente relacionada con el caso. Se debe de seguir religiosamente la cadena de custodia y las directrices de etiquetar todo lo que ha sido retirado."

Preservación de la evidencia: [5] "Antes de poder probar que se ha mantenido la integridad de los datos presentados como evidencia, se debe probar que se ha mantenido la integridad del Hardware que contiene los datos. Desde el inicio de la investigación, se deben de tomar las precauciones y documentar estas precauciones, para proteger, en este caso al Hardware."

"El máximo objetivo de la preservación de la evidencia es asegurarse de manera absoluta que no ha ocurrido cambio alguno desde que la evidencia fue recolectada. Se debe examinar los procedimientos de recolección y de manipulación."

"Tomando todas las precauciones necesarias para proteger la evidencia recolectada de daño que pueda cambiar su estado. Una precaución significativa es la descarga de electricidad estática. Se debe de proporcionar protección estática a los dispositivos de la investigación. Se deben de utilizar y realizar anotaciones que expliquen los pasos que se toman para evitar daños imprevistos."

"Durante la investigación se abordarán muchos temas. No se debe de manipular la evidencia hasta estar absolutamente seguro de que legalmente se puede adquirir la evidencia y que los procesos de recolección y análisis no modifican la evidencia."

---

[4] Alberto Hornero, "Análisis Forense Digital, The Sleuth Kit. [1/2] « blog.ahornero.com," http://ahornero.wordpress.com/2009/05/25/analisis-forense-digital-the-sleuth-kit/
[5] Idem

Análisis de la evidencia: [6] "Antes de iniciar el examen del medio, se debe de crear un hash de la copia que se ha realizado del medio original. ¿Este hash generado concuerda con el hash del medio original? Si es así, se puede proceder. Si no lo es, se debe encontrar la razón."

"Puede darse el caso de que ocurrió algún tipo de escritura cuando se montó la copia. O tal vez, el proceso de copia tuvo alguna falla. En cualquier caso, no se puede iniciar el análisis hasta que se tenga una copia limpia y fiable."

"El proceso de análisis es una mezcla adecuada de ciencia y arte. Se tiene que desarrollar un sentido de donde buscar en primera instancia, y poseer conocimientos técnicos para extraer la información."

Documentación y presentación de resultados: [7] "Después de que el análisis se ha completado, es momento de presentar los resultados. El objetivo de cualquier caso es persuadir a la audiencia de utilizar la evidencia."

"La audiencia puede ser un juez, un jurado, o una junta de gerentes en una sala de conferencias. El objetivo es utilizar la evidencia que se ha recolectado para probar uno o más hechos. Incluso con una gran evidencia, el éxito del caso depende de la efectividad de la presentación."

1.4 – Ficheros

El desarrollo de esta investigación se encuentra estrechamente relacionado con los ficheros o archivos computacionales, sus estructuras binarias y estándares de implementación.

Es por esto que en aras de orientar de forma coherente el desarrollo de este trabajo de diploma, se dedica el siguiente epígrafe a todo lo relacionado con los ficheros informáticos, en especial en los aspectos que resultan de interés criminalístico dado el objetivo de esta investigación.

[6] Alberto Hornero, "Análisis Forense Digital, The Sleuth Kit. [1/2] « blog.ahornero.com," http://ahornero.wordpress.com/2009/05/25/analisis-forense-digital-the-sleuth-kit/
[7] Idem

Dichos aspectos de interés criminalístico no son más que los rasgos identificativos y/o de caracterización que puede contener un determinado fichero, que permita aportar indicios o pruebas concluyentes al trabajo de un investigador forense. Los conceptos de Ficheros sistematizados en el transcurso de la investigación no son lo suficientemente abarcadores, ni esclarecedores del fenómeno que describen.

Tomando como base las conceptualizaciones de un conjunto de investigadores, se ha definido como un Fichero Informático, a los efectos de esta investigación, al conjunto estructurado y organizado de información que es almacenado en un medio computacional de escritura y que puede ser manipulado por el Sistema Operativo de un Ordenador.

Todos los ficheros informáticos son identificados por un nombre y una extensión. Su información varía y va desde una información de texto, hasta cualquier tipo de ejecutables o archivos de sistemas, contenidos multimedia y gráficos, entre otros. Son agrupados en directorios dentro del sistema de archivos siendo el nombre y su extensión la identificación única en relación a los otros ficheros del mismo directorio.

Esta organización de datos en archivos y directorios es original del sistema operativo Unix y es ampliamente seguido por los sistemas operativos modernos. En algunos sistemas operativos los nombres de los archivos son sensibles a la capitalización en lo referido a la distinción entre mayúsculas y minúsculas como en Unix, sin embargo en DOS y Windows las mayúsculas y las minúsculas no tienen importancia a la hora de elegir el nombre para un archivo.

Los ficheros se utilizan cuando se desea almacenar datos de manera persistente. Dependiendo de cada sistema de archivos, los ficheros pueden tener atributos particulares como, por ejemplo, fecha de creación, fecha de última modificación, propietario y permisos de acceso. El tamaño de un archivo está limitado por una serie de factores, como la capacidad de memoria del ordenador y los límites impuestos por el sistema operativo o el sistema de archivos.

El tipo de un archivo es caracterizado por la organización de los datos contenidos y la interpretación que realiza el software que los escribe o los lee. En varios sistemas operativos como es el caso del DOS y el Windows de Microsoft, una extensión es imprescindible para el reconocimiento del tipo de archivo por las aplicaciones y SO.

En otros sistemas operativos, el tipo de archivo puede ser identificado por otros mecanismos.

1.4.1- Las Estructuras de los Ficheros.

Una de las clasificaciones que adquieren los ficheros informáticos es de acuerdo a su estructura. Atendiendo a sus especificaciones binarias, un fichero informático se divide en simples y compuestos, atendiendo a la forma en que organiza la información contenida, al almacenamiento de metadatos y a la presencia o no de marcadores internos en su estructura física, de acuerdo con las especificaciones que en aras de su estandarización su creador definió.

Los ficheros simples son aquellos que en una vista lógica de su estructura se puede apreciar que no cuentan con divisiones internas que le permitan organizar la información encapsulada, observándose únicamente los datos que el mismo contiene, ya sea fruto de su generación automática como es el caso de los ficheros de sistema, así como en los que son contenedores de informaciones entradas por un usuario. (**Ver Figura 1**)

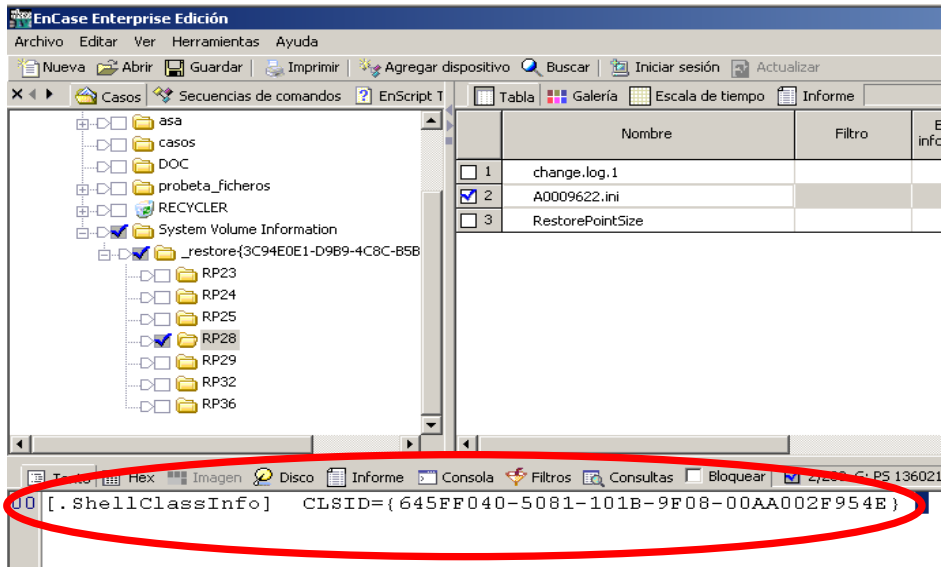


Fig. 1 – Vista lógica del contenido de un fichero simple. (*.ini) Se puede constatar que solo contiene en su interior la información que le consignó el usuario o el sistema.

Los compuestos son portadores de una estructura más compleja y organizada. Utilizando marcadores o identificadores en forma de esqueleto, cual etiquetas de un lenguaje de programación.

Los antes mencionados no solo guardan en su interior la información del usuario, sino que en su armazón recogen también datos del programa que le da vida e incluso del sistema que lo manipula. Ordenan la información haciendo uso de parámetros internos que le permiten guardar estilos, colores, fuentes, entre otros que hacen posible la interpretación del fichero por otros programas. (**Ver Figura 2**)

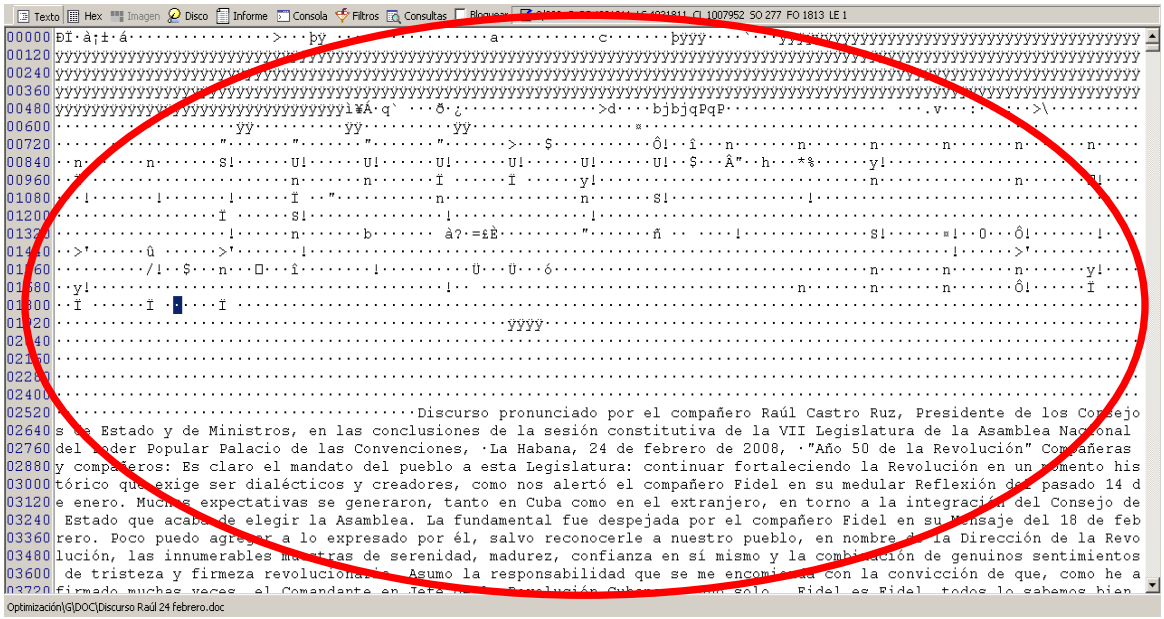


Fig. 2 – Vista lógica del contenido de un fichero compuesto. (*.doc) Se puede constatar qué en adición al texto (información consignada por el usuario), contiene otros caracteres alfanuméricos propios de la estructura binaria del estándar de ficheros.DOC

1.4.2- Estructuras compuestas. Su empleo en la recuperación de ficheros eliminados.

Como se ha explicado en los epígrafes anteriores, son precisamente los ficheros compuestos, los que hacen posible dada la estructuración de su contenido y la presencia de marcadores y metadatos ubicar ficheros eliminados o fragmentos de estos en el espacio no asignado de un disco. Cuando nos disponemos a buscar residuos de un fichero que resulta de interés para una investigación forense cualquiera, basta con conocer cuáles de esas estructuras internas con que cuenta, son parámetros invariantes en los estándares binarios de ese formato de archivo. (**Ver Figura 3**)

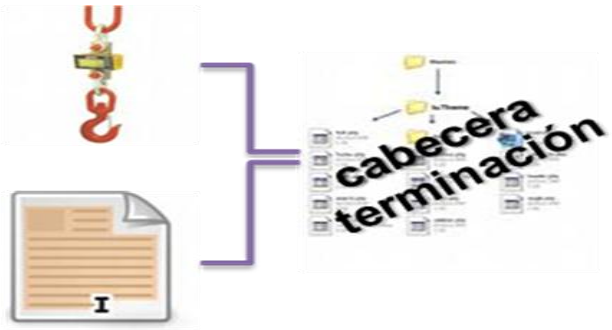


Fig. 3 – Representación de la recuperación de ficheros o residuos de estos, localizando en el espacio no asignado de un disco, una o varias estructuras propias e invariantes de ese tipo de formato.

Esas componentes invariantes de ser identificadas en el espacio no asignado nos permitiría identificar la presencia de un fichero informático o una parte de él. De la posibilidad de determinar las estructuras invariantes de un determinado formato dependerá la capacidad de la disciplina a la hora de localizar, ante un accidente o por la acción intencionada de un criminal, "el cuerpo" de quien en vida fuera un fichero informático.

1.5 - Los sistemas de archivos

Con el progresivo desarrollo de las tecnologías informáticas y el aumento exponencial de los volúmenes de información se hizo necesario desarrollar mecanismos para garantizar la persistencia y manipulación segura de los datos. Surgieron entonces los sistemas de ficheros, los cuales, según una breve definición del Observatorio Tecnológico del Ministerio de Educación, Política Social y Deporte de España, no son más que *"aquellas estructuras lógicas y sus correspondientes métodos que utiliza el propio sistema para organizar los ficheros en disco"*[8].

Actualmente existen múltiples sistemas de archivos, los cuales varían en dependencia del sistema operativo. Para dar soporte a su MS-DOS, Microsoft liberó inicialmente el FAT (File Allocation Table), formato muy básico que solía satisfacer las necesidades del momento.

[8] *Sistema de ficheros GNU/Linux*Observatorio Tecnológico del Ministerio de Educación, Política Social y Deporte de España, http://observatorio.cnice.mec.es/modules.php?op=modload&name=News&file=article&sid=549&mode=thread&order=0&thold=0&POSTNUKESID=71e64c9ba56ad2bfa28de218c2bcbc5c

Con el tiempo este sistema de archivos fue sufriendo cambios para adaptarlo a las condiciones y requerimientos de los nuevos sistemas, y surgieron así varias implementaciones del FAT:

- FAT 12: es el sistema de archivos del DOS, con el cual se formatean los disquetes. Fue muy utilizado en las primeras computadoras.

- FAT 16: este sistema de archivos tenía muchas limitaciones; si el disco duro era mayor de 2 GB era imposible particionarlo, además de que los nombres de archivos se limitaban a 8 caracteres.

- FAT 32: fue utilizado a partir de 1997 en Windows 98, pero a medida que el tamaño de los discos duros se incrementaba, surgieron nuevas limitaciones. Se le llamó así porque utiliza números de 32 bits para representar los clústeres en lugar de los 16 que empleaban sus predecesores.

El particionado y administración de discos duros con FAT se hacía difícil y a veces imposible por el incremento progresivo de las potencialidades del hardware (solo soporta particiones de hasta 2 GB sobre MS-DOS y hasta 4 en Windows NT), razón por la cual Microsoft decide crear el NTFS (New Technology File System) especialmente para Windows NT, el cual elimina las limitaciones de tamaño de los clústeres e incorpora funcionalidades de compresión y encriptación de datos.

A diferencia de lo que hizo con el FAT, el gigante transnacional no publicó las especificaciones del NTFS, pero gracias a las técnicas de ingeniería inversa se han podido desarrollar controladores estables para proporcionar soporte a otros sistemas operativos.

Microsoft desarrolló también, conjuntamente con IBM, el HPFS (High Performance File System), diseñado específicamente para el sistema operativo OS/2 con el objetivo de mejorar las limitaciones del FAT. Éste poseía también una tabla de archivos (como FAT), pero posicionada físicamente en el centro de la partición, lo cual proporcionaba menores tiempos de acceso a la hora de leerla y escribirla.

La aparición del sistema operativo GNU/Linux abrió las puertas a nuevos sistemas de ficheros. Los sistemas nativos de las distribuciones Linux son el ext2 (Second Extended Filesystem) y ext3 (Third Extended Filesystem). Éstos garantizan la compatibilidad con versiones anteriores, de modo que en actualizaciones posteriores del sistema de ficheros, no se perderán los datos almacenados previamente.

Existen además otros menos utilizados, como el ReiserFS, opción por defecto en algunas distribuciones (Linspire, SuSe). Normalmente, para ficheros de tamaño pequeño tiene mejor rendimiento que los extendidos 2 y 3. Otros sistemas de archivos conocidos son el XFS, sistema de 64-bits con rendimiento optimizado para ficheros de gran tamaño, y el JFS (Journaling File System) desarrollado por IBM para servidores que requieran un alto rendimiento.

Independientemente del sistema de archivos, el principio de funcionamiento es el mismo: cada cual posee la mencionada tabla de asignación de archivos (o de *inodos* en el caso de los sistemas UNIX), cuyo objetivo es almacenar un índice con información estructural por cada archivo existente en la unidad, lo cual permite al sistema operativo ubicarlo físicamente y crear la estructura arbolea completa del directorio.

1.6 - Sobre la Fragmentación de los Sistemas de Archivos

De acuerdo con el "Diccionario de la lengua española © 2005 Espasa-Calpe", el significado del término fragmentación es el siguiente:

> Fragmentación:
>
> 1. f. Fraccionamiento, división en partes o fragmentos:[9]

La fragmentación es una problemática que es consecuencia del ordenamiento interno de los datos de algunos sistemas de archivos. Encontrado con bastante regularidad en el sistema operativo Windows, también afecta a otras plataformas pero en una escala inferior. Se produce fragmentación también en la memoria RAM del ordenador cuando se asignan los procesos a los diferentes bloques de memoria.

[9] "fragmentación - Definición - WordReference.com," http://www.wordreference.com/definicion/fragmentaci%C3%B3n

La fragmentación ocurre cuando un trozo de datos en disco, es dividido en disímiles pedazos. (**Ver Figura 4**)

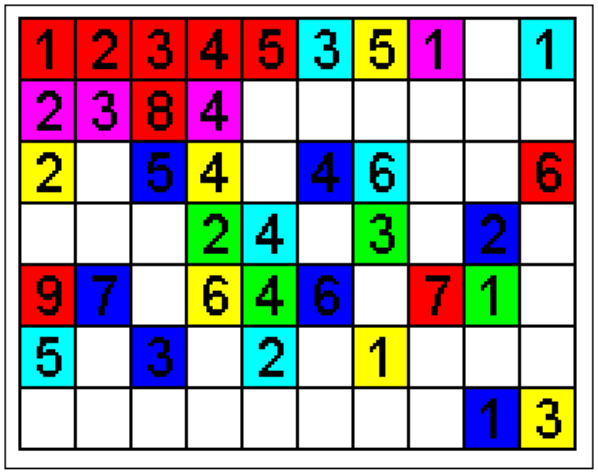


Fig. 4 - Representación del proceso de fragmentación del Sistema de Archivos. Los cuadros de colores representan fragmentos de un mismo archivo. (Asociar cada archivo según su color)

Existen dos tipos de fragmentación: interna y externa.

Fragmentación interna:

En un ordenador, el sistema de archivos organiza los medios de almacenamiento, como los discos duros, en estructuras por bloques comúnmente denominadas clúster y que tienen un tamaño especifico en bytes. El mismo varía según el sistema de archivos y el tamaño de la partición.

Por todo esto se recomienda no disponer en el trabajo de un tamaño de partición demasiado grande en los discos nuevos donde el factor capacidad siempre es muy importante. Por ejemplo si nuestro clúster en cuestión es de 300 KB por más que un archivo ocupe menos, en el dispositivo ocupará 300 KB. Esto acarrea una pérdida de espacio que el sistema de forma errónea informa utilizado. Ese sacrificio de espacio se denomina fragmentación interna, y no se corrige con el desfragmentador, sino disminuyendo el tamaño de la partición.

Fragmentación externa:

El tipo de fragmentación en cuestión, se genera como consecuencia de las diferentes normas de ajuste en bloques que tiene un sistema de ficheros, o como resultado de utilizar asignaciones dinámicas en el caso de la memoria. En el sistema de ficheros, la constante generación y eliminación de ficheros de diferentes tamaños puede conducir al aislamiento de los bloques libres de un disco.

La fragmentación se produce en la memoria del sistema cuando los procesos ordenados han ocupado posiciones no contiguas de memoria dejando excesivos bloques libres de escaso tamaño, en los que no se puede registrar nuevos procesos.

La desfragmentación en sistemas de ficheros, trata de dar solución a esta problemática, alineando los bloques de datos consecutivos y agrupando los bloques libres, obteniendo así intervalos mayores en la zona libre almacenamiento, que en el futuro será empleada para futuros ficheros. A nivel de memoria, este problema se soluciona comprimiendo los procesos tratando de que estos ocupen posiciones anexas y dejar los bloques libres juntos. Con la paginación de memoria, se le da solución también a este desafío.

¿Qué es Desfragmentación?

"Este proceso consta de ordenar los trozos de información distribuida a través de todo el disco, para mejorar la velocidad de acceso y distribuir de mejor forma el espacio libre del dispositivo. Como este proceso consta en la reorganización de partes de archivos, requiere de suficiente memoria para realizar los movimientos de los trozos de información. Al mover en forma física la información, la estructura lógica no sufre alteraciones".[10]

Por lo general los sistemas operativos incorporan sus propios softwares de desfragmentación. Se conoce de aplicaciones externas, las cuales poseen opciones más avanzadas que las propuestas por los fabricantes del sistema operativo. El más conocido desfragmentador es el Defrag, usado en MS-DOS y las plataformas de Windows en estas últimas bajo el nombre de "Desfragmentador de disco", apareciendo usualmente en las propiedades de los discos duros accesible por el módulo Mi PC.

[10] "Desfragmentación - Wikipedia, la enciclopedia libre," http://es.wikipedia.org/wiki/Desfragmentaci%C3%B3n

1.7 – Manipulación de ficheros

Cuando el gestor de ficheros elimina un archivo, los bytes que lo componen no son destruidos ni blanqueados, sino se marcan como “disponibles” en el esquema lógico del disco y se elimina su direccionamiento correspondiente en la tabla de asignación, con lo cual el fichero original se vuelve “invisible” para el sistema operativo, el cual asume que en esa dirección de memoria ya no existen datos. (**Ver Figura 5**)

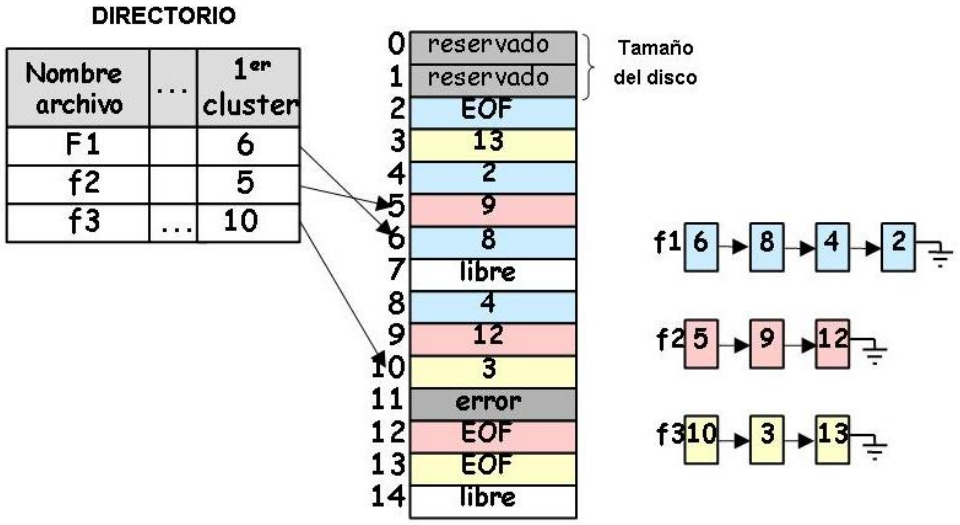


Fig. 5 - Representación de la Tabla de Asignación de los Sistema Operativos, donde se puede apreciar cómo se representan las direcciones en disco de los ficheros de un ordenador.

Luego el sistema utiliza justamente los espacios marcados como “disponibles” para almacenar nuevos datos sobrescribiendo los que, a pesar de ser no-direccionables por el Sistema Operativo, aún persistían físicamente en el dispositivo.

Esos espacios disponibles se van utilizando en el acopio de nueva información, y el archivo que aun existe físicamente comienza a ser sobrescrito progresivamente hasta quedar totalmente remplazado. Es por ello que la primera medida a adoptar tras un borrado accidental de datos es no utilizar el disco duro, extraerlo y conectarlo como disco secundario a otra máquina.

1.8 – Sobre la recuperación de ficheros eliminados

Existen versiones comerciales de sistemas de restauración de datos y programas especializados en la Computación Forense que identifican evidencias digitales dentro de un dispositivo de almacenamiento comprometido. (Memoria flash, disco duro, reproductor mp3, etc.) Los mismos en su generalidad utilizan algoritmos computacionales basados en la búsqueda de las estructuras físicas con que cuenta todo fichero informático.

Algunos de esos parámetros o estructuras son invariantes; es decir, se puede garantizar que siempre estarán presentes en ese tipo de archivos independientemente de la versión, fabricante y software que le dio vida.

Esta última condición, la invariabilidad, constituye una propiedad imprescindible para basar el funcionamiento de cualquier algoritmo dirigido a localizar, en cualquier contexto, algún fichero o residuo del él. Los programas forenses y comerciales de recuperación intentan reconstruir trozos de archivos típicos probando opciones predeterminadas.

Normalmente, si ya se ha mutilado la información de cabecera del archivo, la recuperación es imposible. Por ello la necesidad de determinar otros parámetros identificativos en los ficheros que permitan su recuperación aunque solo persistan residuos de su contenido original.

Las concepciones comerciales se encuentran bastante alejadas del trabajo pericial criminalístico, sus algoritmos utilizan técnicas invasivas que no garantizan la integridad de la evidencia, y por tanto son descartables en un proceso penal.

En cambio las herramientas forenses en su generalidad emplean en sus análisis imágenes, simulan copias tanto lógicas como físicas de los dispositivos sujetos a un proceso investigativo, por lo que la evidencia nunca se verá comprometida.

1.9 - Herramientas frecuentes en la adquisición de evidencias en discos

Las herramientas informáticas, son la base esencial en los análisis de las evidencias digitales en los medios informáticos. Sin embargo, es preciso comentar que éstas requieren de una formalidad adicional que permita validar tanto la confiabilidad de los resultados de la aplicación de las mismas, como la formación y conocimiento del investigador que las utiliza. Estos dos elementos hacen del uso de las herramientas, una constante reflexión y cuestionamiento por parte de la comunidad científica y práctica de la informática forense en el mundo.

Estrictamente hablando, en todo el mundo Análisis Forense se refiere a la recopilación legal de evidencias que puedan servir como prueba judicial. Es por ello que la mayor parte de las técnicas se basan en la recuperación de información de dispositivos de almacenamiento.

Dentro de las herramientas frecuentemente utilizadas en procedimientos forenses en informática detallamos algunas para conocimiento general de los lectores, que son aplicaciones que tratan de cubrir todo el proceso en la investigación forense en informática:

1.9.1 – Forensic Toolkit

Página Web oficial: *http://www.accessdata.com/forensictoolkit.html*

Forensic Toolkit® (FTK®) está reconocido como uno de los entornos más confiables para la investigación forense. Esta plataforma validada por numerosas cortes provee análisis internos, desencriptación y crackeo de contraseñas con una interfaz personalizable y bastante amistosa.

Además, posee la capacidad de conectarse a bases de datos para manipular grandes cantidades de datos. Posee otras funcionalidades como análisis de tráfico de red, de correo electrónico y presentación personalizada de datos que lo hacen más efectivo en entornos pequeños o medianos.

Es una solución integrada que permite en la realización de una investigación, crear imágenes, analizar registros, descifrar archivos, romper contraseñas, y generar un informe con una única solución. Con el Forensic Toolkit se puede recuperar contraseñas de más de 80 aplicaciones, aprovechar los ordenadores ociosos a través de la red para descifrar los archivos y realizar ataques de diccionario robusto. Posee una biblioteca con 45 millones de hashes.

Su vinculación con Oracle permite soportar las más grandes y complejas Bases de datos, hace casi imposible perder la información almacenada como consecuencia de un accidente. Adquiere además la Capacidad para realizar copias de seguridad y archivar los casos pudiendo contar con un Potente motor de búsqueda y una completa funcionalidad del motor de búsqueda de expresiones regulares binarias.

Garantiza un verdadero multi-procesamiento donde se aprovechan los avances del hardware, garantizando potencia y velocidad en el proceso. El perfeccionamiento del Pre- y post-procesamiento le permite controlar cómo se procesan las imágenes, reduciendo el tiempo de procesamiento. Asegura una avanzada talla de datos en el motor que le permite reducir la cantidad de datos irrelevantes a tener en cuenta, especificando los criterios, tales como el tamaño del archivo, tipo de datos y tamaño en píxeles. Cuenta con verificación Hash y la capacidad de filtrar los datos utilizando MD5, SHA1, SHA256.

Tiene una Interfaz intuitiva y funcional. Es fácil de entender y de usar a través de opciones predefinidas y personalizables, filtrado avanzado y categorización automática de datos. Múltiples puntos de vista de los datos le permite a los usuarios analizar un archivo dado de diferentes maneras. Soporta populares tecnologías de cifrado, como Credant, SafeBoot y UTIMACO. Un Rico y poderoso motor de la presentación de informes le permite crear informes detallados y de salida en formato nativo, HTML, PDF, XML, RTF, y muchos más con vínculos que le permiten regresar a las pruebas originales.

1.9.2 – X Ways Forensics

Página Web oficial: http://www.x-ways.net/forensics/index-m.html

X-Ways Forensics es un entorno de desarrollo avanzado para investigadores forenses diseñado para ejecutarse sobre sistemas operativos Windows 2000/XP/2003/Vista/2008.

En comparación con sus competidores X-Ways Forensic suele ser muy eficaz ya que una de sus principales ventajas es que no consume demasiados recursos en la máquina donde se ejecute.
Está basado en el popular editor hexadecimal WinHex.Ofrece muchas características de las que otros carecen, por ejemplo, consta de un modelo de trabajo en colaboración que permite a los investigadores compartir datos y colaborar entre sí. X Ways Forensics encierra todas las características del WinHex como son la clonación de discos y la obtención de imágenes, soporte nativo para FAT, NTFS, Ext2/3/4, Incorpora la interpretación de RAID 0 y RAID 5 y sistemas dinámicos de discos.

Garantiza el Acceso completo a discos, RAID, y a imágenes de más de 2 Terabytes de tamaño así como diversas técnicas de recuperación de datos y archivos. Para el cálculo de hash emplea algoritmos certificados para ese fin. (CRC32, MD4, ed2k, MD5, SHA-1, SHA-256)

Adicionalmente implementa la gestión completa de casos, registros automatizados de actividad (logs de auditoría), capacidad de análisis a distancia para que las unidades de red se pueden añadir opcionalmente. Cuenta con protección contra escritura de datos para garantizar la autenticidad de la evidencia sometida a análisis. Realiza un seguimiento de los archivos que ya fueron vistos durante la investigación y permite la generación de informes que pueden ser importados y posteriormente procesados por cualquier otra aplicación que entienda el formato HTML o MS Word.

1.9.3 – The Sleuth Kit & Autopsy

Página Web oficial: *http://www.sleuthkit.org*

The Sleuth Kit & Autopsy Browser son herramientas de código abierto diseñadas para la investigación digital. Pueden correr tanto sobre los sistemas operativos de la familia Microsoft Windows como sobre sistemas Unix (Linux, OS X, Cygwin, FreeBSD, OpenBSD, y Solaris). Posee soporte para diferentes sistemas de ficheros, como NTFS, FAT, Ext2, Ext3, UFS1, and UFS2 así como para distintos tipos de volúmenes y dispositivos.

Originalmente solo existía The Sleuth Kit, un intérprete de línea de comandos; luego se desarrolló y acopló con él la interfaz Autopsy, quien le permite más fácilmente llevar a cabo una investigación. Autopsia provee manejo de casos, la integridad de las imágenes, la búsqueda de palabras claves y otras operaciones automatizadas.

Dentro de las bondades que esta herramienta forense brinda se encuentran la de Mostrar los datos del sistema de archivos y la meta-estructura de información del fichero. The Sleuth Kit está programado en C y Perl y usa un poco del código y el diseño de The Coroner's Toolkit.

1.9.4 – The Coroner Toolkit

Página Web oficial: http://www.porcupine.org/forensics/tct.html

La primera versión liberada alrededor de agosto de 1999. The Coroner Toolkit (TCT) es una colección de programas hecha por los desarrolladores Dan Farmer y Wietse Venema para realizar análisis post-mortem de sistemas UNIX luego de la ejecución de acciones indebidas o criminales. Fue presentado por primera vez en agosto de 1999.

Es un programa que está relativamente sin pulir en comparación con el software que Dan y Wietse generalmente producen. The Coroner Toolkit suele pasar mucho tiempo en la recopilación de los datos.

El Forense es un campo en el que la brecha entre los datos brutos y la información significativa hace la diferencia. Este programa automatiza el proceso de recogida de la evidencia, eliminando en un cierto margen la probabilidad de un error humano. TCT es fácil de instalar y configurar.

1.9.5 – HELIX

Página Web oficial: *http://www.e-fense.com/helix/*

Hélix es un entorno de respuesta y protección ante incidentes. Tiene la capacidad de responder ante diferentes peligros que amenacen a unidades o imágenes de discos o a la memoria RAM del sistema. Recolecta además información sobre el tráfico y la actividad de red de los usuarios. Detecta, identifica, analiza y preserva la evidencia necesaria para revelar la verdad de un evento o acción no autorizado.

Hélix3 Enterprise se controla a través de una interfaz gráfica fácil de utilizar, que funciona con cualquier sistema operativo. Es tan fácil que no requiere de ningún entrenamiento. La instalación se hace rápidamente y puede ser ejecutada a través de su red usando las herramientas de su instalación existente.

Con un simple clic se puede realizar una captura de pantalla o una llave de registro en cualquier sistema en su red. Puede seleccionar el sistema en el menú para obtener lo que hay en la pantalla en un momento dado.

Hélix Enterprise fue desarrollado por un equipo de expertos forenses e investigadores de crímenes cibernéticos. Garantiza la Presentación de informes siendo esta una parte importante de cualquier aplicación forense. Puede generar informes concisos sobre las auditorías realizadas sobre la base de diferentes criterios.

1.9.6 – EnCase Enterprise

Página Web oficial: http://www.guidancesoftware.com

Desde su lanzamiento a finales de la década de los 90 del pasado siglo, EnCase se convirtió casi de manera automática en el entorno preferido para el análisis forense en todo el mundo. Su seguridad y confiabilidad para manipular evidencia electrónica y sus potencialidades lo han convertido en el estándar mundial en este sentido.

Cuenta con un lenguaje de programación propio orientado a objetos llamado EnScript que reúne elementos de C++ y Java, y permite a los investigadores personalizar los scripts predeterminados o escribir los suyos propios.
Es el estándar a nivel mundial en computación forense: Utilizado por más de 12.000 investigadores y profesionales de la seguridad.

Ofrece una solución revolucionaria integrando en una sola herramienta todo el proceso forense. EnCase proporciona las herramientas más avanzadas para el Análisis Forense de Sistemas e Investigaciones Digitales.

Con un entorno gráfico intuitivo, flexible y un rendimiento sin igual, proporciona a los investigadores todo lo necesario para realizar análisis a gran escala en investigaciones complejas con precisión y seguridad.

Una solución premiada que garantiza por completo la integridad de la información tratada, permitiendo a los analistas gestionar con facilidad grandes volúmenes de evidencias digitales incluso en ficheros borrados, en áreas de su estructura física, zonas de paginación y clústers sin asignar.

La aplicación proporciona las adquisiciones de medios con más validaciones de la industria. Crea un duplicado binario exacto de la unidad o disco original y luego lo verifica al generar valores hash MD5 para archivos de imágenes relacionados. Además asigna valores de control de redundancia cíclica (CRC, por su sigla en inglés) a los datos a fin de revelar las instancias en que las pruebas se modificaron forzosamente o se alteraron de alguna manera.

Este enfoque fue validado por el Instituto Nacional de Normas y Tecnología (National Institute for Standards and Technology, NIST) y resistió numerosos cuestionamientos por parte de los tribunales de justicia.

Los productos EnCase incluyen la programación EnScript™, un macro lenguaje de programación que permite a los usuarios crear scripts personalizados para automatizar tareas de investigación prolongadas, como la búsqueda y el análisis de tipos de documentos específicos. La sintaxis de la programación EnScript® está basada en JAVA y C++.

EnCase Enterprise actúa como plataforma común para otros productos de Guidance Software como EnCase eDiscovery y EnCase Information Assurance, los cuales forman, de conjunto, una completa *suite* para la investigación digital y la protección de los datos que incluye análisis de actividad de redes, auditoría de datos y aplicación de políticas de seguridad dirigidos a corporaciones y agencias gubernamentales o legales. Desarrollado por expertos en análisis forense penal, EnCase Forensic cuenta con la aprobación de tribunales de justicia de todo el mundo.

El software sigue siendo la herramienta elegida por el FBI, el Departamento de Seguridad Nacional de Estados Unidos (US Department of Homeland Security), el Departamento de Defensa de Estados Unidos (US Department of Defense), New Scotland Yard y miles de laboratorios criminalistas y agencias encargadas del cumplimiento de la ley en todo el mundo.

Otras agencias con departamentos especializados en Informática Forense que utilizan EnCase como herramienta fundamental para sus peritajes informáticos son la Brigada de Investigación Tecnológica del Cuerpo Nacional de Policía de España, la National Hi-Tech Crime Unit de Inglaterra, la INTERPOL y otras homólogas en países latinoamericanos como Argentina, Colombia, Chile y Perú.

Ventajas de la tecnología EnCase:

- Total visibilidad en toda la red empresarial, revelando los posibles riesgos operacionales y mitigando las amenazas

- Respuesta automática a las alertas de seguridad, iniciando una investigación y evitando que los datos potenciales se pierdan o se corrompan

- Reducción de los costes e incremento de la productividad al realizar las investigaciones y la recogida de pruebas desde una ubicación central y de forma distribuida

Vista panorámica sobre Guidance Software

Guidance Software, la firma desarrolladora de las versiones de EnCase, fue fundada en 1997 con el objetivo de desarrollar soluciones capaces de buscar, identificar y recuperar información digital de manera efectiva y poco costosa. Han desarrollado aplicaciones no solo para el análisis de dispositivos físicos, sino para el monitoreo de redes en entornos empresariales o reducidos.

El paquete de aplicaciones de EnCase® provee aplicaciones para investigaciones internas que permite a corporaciones, entidades gubernamentales y agencias de seguridad, conducir de manera efectiva sus pesquisas digitales y tomar decisiones acertadas ante eventos o acciones dañinos.

Guidance Software ha desarrollado múltiples relaciones con otras firmas reconocidas mundialmente en el ámbito de la investigación digital como Etek, corporación multinacional fundada en 1974 que brinda soluciones integrales de Seguridad de la Información, para incrementar el radio de acción de sus aplicaciones.

Actualmente cuenta con más de 20 mil usuarios oficiales de su tecnología en todo el mundo, y otros 4 mil se acogen anualmente a los programas de entrenamiento de Guidance Software. Sus principales clientes son departamentos de policía, agencias gubernamentales y de seguridad.

Guidance Software ha sido patrocinadora y organizadora de la Conferencia de Investigaciones Digitales Empresariales (*Computer Enterprise Investigations Conference, CEIC*), que reúne anualmente a los más eminentes investigadores del mundo para discutir los nuevos avances en materia forense y las prácticas más recomendadas.

Guidance Software también fue acreedora del máximo galardón en el *Gartner's MarketScope for E-Discovery and Litigation Support Vendors* del 2007, evento que selecciona anualmente los mejores productos en el ámbito legal del mercado digital. El reporte completo se encuentra disponible en la Web oficial *www.guidancesoftware.com*.

1.9.7 – Opinión pública valorativa

El sitio hispano http://www.forensic-es.org posee, en su página principal, una encuesta bajo la interrogante: *¿Qué herramienta usas habitualmente para obtener las imágenes de las evidencias?* Hasta el momento, el 37% de los encuestados coincidió en señalar a EnCase como la herramienta predilecta. (**Ver figura 6**)

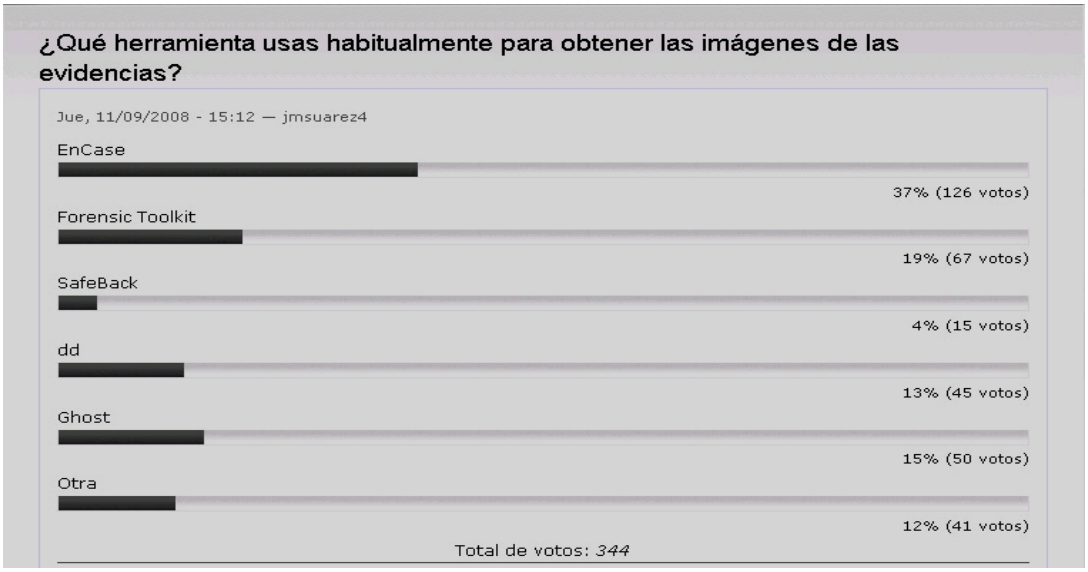


Fig. 6 – Encuesta publicada por el Sitio http://www.forensic-es.org bajo la interrogante ¿Qué herramienta usas habitualmente para obtener las imágenes de las evidencias?

La firma TransUnion, líder global en gestión de informaciones y créditos con más de 500 millones de clientes en todo el mundo, utiliza EnCase desde el año 2007 para recolectar, preservar y analizar la información electrónica que maneja en sus redes.[11]

EnCase también recibió el año pasado 2 galardones de bronce en los premios *Law Technology News* (LTN) para la recolección y procesamiento de información electrónica. El proceso de selección de los premiados se basa en los votos emitidos por los más de 40,000 suscriptores de LTN.[12]

EnCase ha recibido múltiples premios y reconocimientos por su efectividad y confiabilidad en el análisis de datos. La prestigiosa revista SC Magazine le otorgó durante 2 años consecutivos, en 2001 y 2002, la categoría de producto 5 estrellas en su tradicional entrega de reconocimientos.[13]

[11] "Guidance Software, Inc. - TransUnion Selects Guidance Software's EnCase Enterprise to Conduct Its Network-wide Digital Investigations," http://investors.guidancesoftware.com/releasedetail.cfm?ReleaseID=270731
[12] "Law Technology News - Home," http://www.lawtechnews.com/r5/showkiosk.asp?listing_id=2003877
[13] "ENCASE FORENSIC 6.1, presente en BOLIVIA," http://www.nobosti.com/spip.php?article497

EnCase está validado por numerosas cortes, departamentos, agencias policiales y organizaciones legales alrededor del mundo. Con frecuencia recibe reconocimientos de importantes revistas como eWeek, Network Computing y SC Magazine. En el año 2008 recibió el premio dorado en la encuesta tecnológica alemana Socha-Gelbmann´s.[14]

1.10 - Conclusiones

En el desarrollo de este capítulo hemos analizado los conceptos básicos de la disciplina de Informática Forense y de Evidencia Digital. Abordamos aspectos importantes en la manipulación de los ficheros y el estado del arte de las herramientas forenses más exitosas en el mercado internacional.

A pesar de la amplia gama de productos enfocados al análisis y recolección de evidencias, el líder del mercado en entornos forenses de discos es EnCase, que puede realizar duplicados exactos del contenido de un disco, incluso de forma remota. EnCase ha sido aceptado internacionalmente como el estándar para la manipulación de evidencias digitales y el análisis forense.

La propuesta de solución se basa entonces en el desarrollo sobre la herramienta forense EnCase Enterprise v4.20, tomando en consideración que sus productos incluyen la programación EnScript™, un macro lenguaje de programación que permite a los usuarios crear scripts personalizados para automatizar tareas de investigación prolongadas, como la búsqueda y el análisis de tipos de ficheros específicos.

A su vez, coincide con la herramienta forense utilizada y el conocimiento que existe en la Sección de Informática Forense del Ministerio del Interior, entidad a la que están dirigidos los aportes que se derivarán de este trabajo de diploma.

[14] "EnCase Search Technology Validated by Federal Court in Contested Electronic Discovery Ruling | Reuters," http://www.reuters.com/article/pressRelease/idUS117764+23-Mar-2009+BW20090323

CAPÍTULO 2
SOBRE LAS ESTRUCTURAS DE LOS FICHEROS

2.1 - Introducción

Como se explicaba en el capítulo anterior, los ficheros informáticos modernos generalmente cuentan con estructuras compuestas. Con el objetivo de lograr una caracterización formal de las estructuras internas de los diferentes tipos de ficheros que son objeto de este trabajo de diploma, se realizó un estudio de las normas oficiales que rigen las especificaciones binarias de los formatos de archivos JPEG y PSD.

2.2 – Estandarización de los tipos de ficheros

Para todos los tipos de ficheros conocidos existe documentación técnica conocida como 'especificación de formato de fichero', la cual describe la estructura interna del formato para que pueda ser utilizado por los programadores a la hora de escribir sus aplicaciones, y así poder estandarizar la lectura y comprensión de los diferentes formatos de archivos.

Por tal motivo, al comienzo de esta investigación se procedió a obtener los documentos oficiales con las normas para la construcción de algunos archivos específicos. Esos documentos se nombran a continuación:

<u>Imágenes JPEG</u>:

- ISO IEC IS 10918-1: *Information technology - Digital compression and coding of continuous-tone still images: Requirements and guidelines.*
- *JPEG File Interchange Format. 1992, C-Cube Mycrosystems.*

<u>Archivos de Adobe Photoshop</u>:

- *Adobe Photoshop® 5.5 File Formats Specification. Copyright © 1991–1998 Adobe Systems Incorporated.*
- *Adobe Photoshop® CS File Formats Specification. Copyright © 1991–2003 Adobe Systems Incorporated.*

2.3 - Metadatos

Los metadatos son información que describen características o propiedades de un documento, y son distinguibles dentro del contenido principal del mismo. Por ejemplo, en el caso de un documento de texto, el contenido incluye el texto escrito por el usuario, las tablas, imágenes insertadas, el formato de ese texto (fuente, color, alineación…), etc., mientras los metadatos recogen información como el autor, las fechas de creación y modificación, los derechos de autor y otros.

Pueden existir áreas dudosas en las cuales la misma información pudiera ser tratada como contenido o como metadatos. De manera general, los metadatos deben almacenar información, independientemente del contenido del archivo. Por ejemplo, una lista de todas las fuentes empleadas en un documento pudiera ser un metadato útil, mientras la información relativa a un tipo de fuente específico utilizado en un párrafo determinado sería manipulada como contenido.

Los metadatos permiten a usuarios y aplicaciones manipular más eficientemente los documentos. Las aplicaciones pueden hacer varios usos de los metadatos de un fichero, incluso si no comprenden el formato nativo del documento.

Los sistemas de ficheros suelen proveer metadatos tales como fechas de modificación de los archivos y sus tamaños. Metadatos adicionales pueden ser suministrados por los usuarios o por las aplicaciones específicas que manipulen el fichero. Los metadatos pueden o no ser almacenados como parte del archivo al cual están asociados.

2.4 - Caracterización de las estructuras internas

2.4.1 - El formato de ficheros de Adobe Photoshop

Introducción

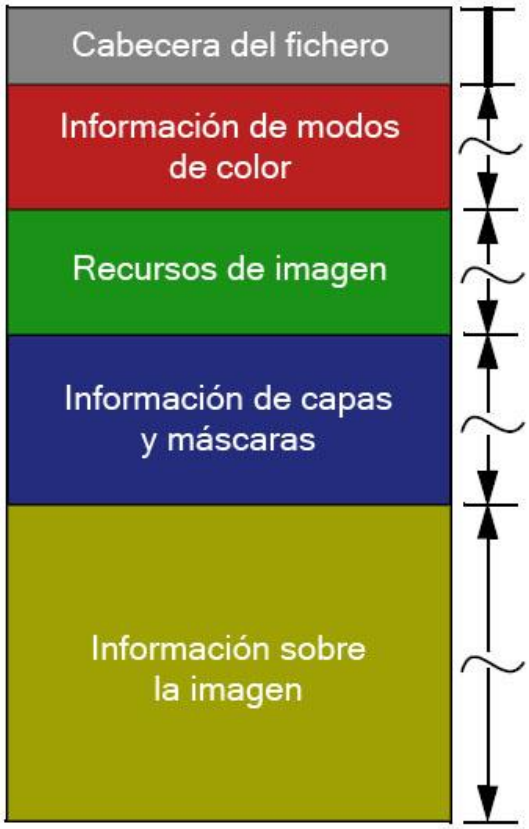
Adobe Photoshop se ha mantenido durante más de una década como la herramienta de edición preferida por diseñadores gráficos y usuarios en general, dadas las amplias posibilidades que ofrece a partir del amplio espectro de funcionalidades que incluye.

El desarrollo gradual de los sistemas operativos de código abierto generó la necesidad de crear aplicaciones homólogas en esos entornos, que fueran capaces de emular con sus similares de la plataforma Windows. En el ámbito del diseño gráfico en 2 dimensiones, la aplicación GIMP ha logrado igualar muchas de las características de Photoshop, aunque la segunda continúa en la preferencia general.

Estas aplicaciones, al igual que las demás que permiten manipular ficheros nativos de Adobe Photoshop (.psd), conocen la estructura interna de ese formato de los ficheros, lo cual les permite interpretar los diferentes segmentos de datos que lo componen y modificar el contenido del archivo de manera inteligible.

Especificaciones del formato binario de Adobe Photoshop

Cada fichero de Adobe Photoshop está compuesto por 5 grandes bloques de datos, los cuales son: (**Ver Figura 7**)



1. Cabecera del archivo.
2. Información sobre los modos de color.
3. Recursos de la imagen.
4. Información sobre capas y máscaras.
5. Información de la imagen (píxeles).

Fig. 7 – Representación de los bloques lógicos de datos en los que se encuentra seccionado el estándar PSD.

1 – Segmento de cabecera del fichero

Este segmento almacena las propiedades básicas de la imagen. Es el único de los 5 que posee una longitud fija, equivalente a 26 bytes, los cuales están asignados de la siguiente manera (en orden continuo, comenzando por el primero):

Longitud (bytes)	Descripción
4	Identificador de cabecera: equivale a la secuencia ‘8BPS’. Si no se corresponde con este valor, el fichero puede estar dañado o corrupto.
2	Número de versión: siempre igual a 1. Si no se corresponde, puede ser un indicio de que el fichero está corrupto o dañado.
6	Bytes reservados: deben ser igual a 0.
2	Número de canales en la imagen. Oscila entre 1 y 56.
4	Altura de la imagen en píxeles. Oscila entre 1 y 30 000.
4	Ancho de la imagen en píxeles. Oscila entre 1 y 30 000.
2	Profundidad: número de bits por canal (1, 8 ó 16).
2	Modo de color de la imagen *(ver descripciones debajo)*: - 0 = mapa de bits - 1 = escala de grises - 2 = color indexado - 3 = RGB - 4 = CMYK - 7 = multicanal - 8 = tono dual - 9 = lab

Mapa de bits: fichero de imagen estructurado en una rejilla rectangular llamada *"raster"* que almacena los píxeles o puntos de color. Es decir, almacena una rejilla con la información de color para cada píxel. Este formato matricial es el más usado en la fotografía digital.

Escala de grises: escala de colores que se emplea para representar la información de color de una imagen, en la cual cada píxel almacena información equivalente a una graduación de gris. Las intensidades varían entre negro intenso, blanco y 254 tonos de gris.

Color indexado: modo en el cual se lee la información de color de cada píxel de una tabla, llamada *mapa* o *paleta de colores*, que almacena los índices de los 256 posibles colores.

Esquema RGB: hace referencia a la composición del color por la intensidad de la combinación de los 3 colores primarios (rojo, verde y azul). Se basa en la síntesis aditiva por adición de estos 3 colores.

Esquema CMYK: modelo de colores que se basa en la mezcla de los pigmentos de cian, magenta, amarillo y negro. Es un modelo sustractivo, pues la impresión de los 3 primeros sobre blanco resulta en color negro.

Multicanal: modo que emplea 3 canales alfa y 256 colores, similar a la escala de grises. Se emplea en tareas especializadas de impresión.

Tono dual: modo disponible solo para imágenes en escala de grises. Se comporta igual que éste, salvo porque es posible especificar cualquier color de sustitución para cambiar por el gris.

Lab: modo que controla 3 canales de imagen llamados *chrome A*, *chrome B* e iluminación. *Chrome A* controla 128 colores que van del verde al rojo*; chrome B* se encarga de otros 128 entre el azul y el amarillo; e iluminación gestiona el grado de luminosidad de la imagen.

2 – Segmento de modo de color

Esta sección comienza inmediatamente después del último byte de la cabecera –o sea, en el 27-. Su estructura es la siguiente:

Longitud	Descripción
4	Longitud del bloque de información de color que le continúa.
Variable	Información de color

Este bloque solamente almacena datos para los modos de color **indexado** y **tono dual**. En el caso de los restantes 6 modos, esta sección se compone únicamente por los 4 bytes que especifican la longitud con valor 0.

En el caso del color indexado, la longitud de la información de color es de 768. El segmento contiene una paleta de 256 colores, de la cual se extraen los índices de los colores de los píxeles.

Para los tonos duales, el segmento de información contiene las especificaciones propias del esquema de color, las cuales no han sido documentadas públicamente por Adobe. Se presume que contenga información de visualización y otros parámetros relacionados. Por lo general, los programadores manipulan su información como escala de grises y la mantienen inalterable.

3 – Segmento de recursos de datos

Este bloque almacena recursos que describen características no asociadas a los píxeles de la imagen. Es decir, las estructuras que se encuentran dentro de él no describen la información específica de los píxeles, sino que hacen referencia a otros atributos de la imagen, tales como metadatos del fichero, firmas de seguridad, escalas y banderas de impresión, guías y reglas, etc.

La sección comienza con un campo que almacena la longitud total del conjunto de recursos, seguido por la información relativa a cada recurso.

Longitud	Descripción
4	Longitud de la sección de recursos de imagen.
Variable	Recursos de la imagen (ver “Bloques de recursos”)

Recursos

Los recursos de una imagen son la unidad básica para construir diferentes formatos gráficos, no solo el de Photoshop, tales como el JPEG *(Joint Photographic Experts Group)* y el TIFF *(Tag Image File Format).* Los recursos de imagen, como se decía anteriormente, se utilizan para almacenar información adicional (no relativa a los píxeles) de una imagen.

Todos los recursos de Adobe Photoshop poseen la estructura siguiente:

Longitud	Descripción
2	Delimitador de recurso: siempre igual a ‘8BIM’.
2	Identificador único del recurso (ver “Identificadores para bloques de recursos”).
Variable	Nombre: cadena en formato Pascal. Si el campo está vacío, consiste en 2 bytes iguales a 0.
4	Longitud de la información del recurso a continuación
Variable	Información del recurso, descrita en las secciones de los tipos de recursos individuales. (ver “Bloques de recursos comunes”)

Identificadores para bloques de recursos

Los 2 bytes siguientes al delimitador de recurso ‘8BIM’ identifican el tipo de recurso en cuestión y, como consecuencia, el tipo de información que almacena en sí. A continuación se muestra la lista de identificadores publicada por Adobe Systems Incorporated.

Adobe Systems Incorporated no especifica en su documentación técnica el orden de aparición de estos recursos de imagen. No obstante, los experimentos realizados sobre una colección aleatoria de ficheros .psd de diversos orígenes y creados con distintas versiones de Adobe Photoshop, ha permitido establecer el siguiente orden:

ID Hex	Descripción
0x0404	Estructura IPTC-NAA. Estándar para metadatos. Aparece en aproximadamente el 27% de la muestra. Para el resto, el primer recurso es el '0x0425'.
0x0425	CaptionDigest. Hash de seguridad en RSA o MD5.
0x0424	Metadatos del fichero en formato XMP *(ver epígrafe "2.4.1.2.3.1 – XMP: la plataforma de metadatos de Adobe Systems Incorporated")*.
0x03ED	Estructura 'ResolutionInfo'. Almacena información sobre la resolución de la imagen.
0x0426	Escala de impresión.

A continuación, pueden o no aparecer los siguientes 4 recursos, en dependencia de que haya existido o no manipulación de los canales alfa de la imagen:

ID Hex	Descripción
0x03EE	Nombres de los canales alfa
0x0415	Nombres alfa en Unicote
0x03EF	Estructura 'DisplayInfo'
0x041D	Identificadores alfa

Si los 4 recursos anteriores no se encuentran presentes, aparecerían inmediatamente después del recurso 'Escala de impresión' los listados a continuación:

ID Hex	Descripción
0x040D	Ángulo de vista global
0x0419	Altitud global de la imagen
0x03F3	Banderas de impresión
0x040A	Información de copyright
0x2710	Información sobre banderas de impresión
0x03F5	Información de tonalidades en modo color
0x03F8	Funciones de transferencia de color
0x0400	Información de estado de capas
0x0402	Información sobre los grupos de capas
0x0430	No documentado oficialmente por Adobe
0x042D	No documentado oficialmente por Adobe
0x0408	Información de guías y reglas

0x041E	Lista de URLs utilizadas
0x041A	Slices
0x0428	Relación de aspecto
0x040F	Perfil ICC
0x0414	Número base para generar IDs de capas
0x040C	Vista previa (incluido a partir de la versión 5)
0x0421	Versión del manipulador de metadatos
0x0422	Información EXIF 1

Los demás identificadores de recursos que se muestran en la siguiente tabla son opcionales (dependen de la manipulación específica de la imagen) u obsoletos (anteriores a la versión 5 de Adobe Photoshop):

ID Hex	**Descripción**
0x03E8	Obsoleto. Almacena número de canales, filas, columnas y modo de color de la imagen.
0x03E9	Información de impresión impresión en Macintosh.
0x03EB	Obsoleto. Tabla de color indexada.
0x03F0	Caption de la imagen.
0x03F1	Información del borde de la imagen.
0x03F2	Color de fondo de la imagen.
0x03F4	Información sobre escalas de grises y multicanales.
0x03F5	Información de tonalidades medias de color.
0x03F6	Información de tonalidades medias del modo ‘duotone’.
0x03F7	Función de transferencia de escala de grises y multicanales.
0x03F9	Funciones de transferencia de tonos duales
0x03FA	Información de imagen en tono dual
0x03FB	Información referente a escala de grises (blanco y negro)
0x03FC	Identificador obsoleto
0x03FD	Opciones EPS
0x03FE	Información de máscara rápida
0x03FF	Identificador obsoleto
0x0401	Ruta de trabajo (sin salvar)
0x0403	Identificador obsoleto
0x0404	Registros IPTC

0x0405	Modo de imagen para formatos en bruto (imágenes raw)
0x0406	Calidad JPEG
0x0409	Vista previa de la imagen (solo para Photoshop 4)
0x040B	URL
0x040E	Muestreadores de color
0x0410	Marca de agua
0x0412	Efectos de visualización de capa
0x0413	Sombra de tono medio (spot halftone)
0x0416	Contador de la tabla de color indexado
0x0417	Índice de transparencia
0x041B	URL del flujo de trabajo
0x041C	Salto a segmento XPEP
0x0423	Información EXIF 2
0x0429	Composición de capa
0x042A	Colores alternos en tono dual
0x042	Colores alternativos (spot colors)
0x0BB7	Nombre de la ruta adjunta

Existe además una serie de identificadores descubiertos durante la realización de pruebas, validaciones y experimentos, que no se encuentran contemplados en la documentación oficial de Adobe. Algunos de ellos son considerados obsoletos por pertenecer a versiones previas a Adobe Photoshop 5. A los efectos de esta investigación, esos recursos serán tratados como **no documentados**. Ellos son:

ID Hex	**Descripción**
0x0FA0	No documentado por Adobe
0x0FA1	No documentado por Adobe
0x416E	No documentado por Adobe
0x526F	No documentado por Adobe
0x0FA1	No documentado por Adobe
0x6E6F	No documentado por Adobe
0x6C66	No documentado por Adobe
0x6C75	No documentado por Adobe
0x6C6e	No documentado por Adobe
0x6C79	No documentado por Adobe

0x636C	No documentado por Adobe
0x636C	No documentado por Adobe
0x696E	No documentado por Adobe
0x6B6E	No documentado por Adobe

De igual manera, los experimentos arrojaron la existencia de recursos adicionales en ficheros creados con las versiones CS3 y CS4 de Photoshop, de los cuales tampoco se tiene docuementación disponible; por tanto, en el transcurso de esta investigación se les tratará como **Recursos de caja negra de Photoshop CS3 o superior**:

ID Hex	**Descripción**
0x0436	Recurso de Photoshop CS3 o superior
0x0433	Recurso de Photoshop CS3 o superior
0x0434	Recurso de Photoshop CS3 o superior
0x0FA2	Recurso de Photoshop CS3 o superior
0x0FA3	Recurso de Photoshop CS3 o superior
0x416E	Recurso de Photoshop CS3 o superior
0x526F	Recurso de Photoshop CS3 o superior
0x0FA4	Recurso de Photoshop CS3 o superior
0x6E6F	Recurso de Photoshop CS3 o superior
0x6C75	Recurso de Photoshop CS3 o superior
0x6C6E	Recurso de Photoshop CS3 o superior
0x6C79	Recurso de Photoshop CS3 o superior
0x636C	Recurso de Photoshop CS3 o superior
0x696E	Recurso de Photoshop CS3 o superior
0x6B6E	Recurso de Photoshop CS3 o superior
0x6C73	Recurso de Photoshop CS3 o superior
0x6C63	Recurso de Photoshop CS3 o superior
0x6C66	Recurso de Photoshop CS3 o superior

XMP: la plataforma de metadatos de Adobe Systems Incorporated

El recurso de imagen ‘0x0424’ identifica el bloque de metadatos en un fichero de Adobe Photoshop. Los principales productos de Adobe Systems, tales como Illustrator, Acrobat y Framemaker, así como otras herramientas ajenas a la compañía, emplean una plataforma común para la incrustación de metadatos en sus respectivos ficheros, denominada XMP (*eXtensible Metadata Platform* o Plataforma Extensible de Metadatos).

El XMP emplea una sintaxis similar a la definida por el lenguaje XML *(eXtensible Markup Language)* para registrar la información. Se basa además en la especificación RDF *(Resource Description Framework* o Entorno de Descripción de recursos*)* del World Wide Web Consortium, un lenguaje de descripciones estandarizado por dicha organización.

En 1994, Adobe definió una especificación para la adición de información descriptiva a las imágenes digitales, denominada “encabezados IPTC”. Posteriormente, en 2001, la firma introdujo el XMP en la mayoría de las estructuras de metadatos de sus aplicaciones.

La estructura interna de un recurso de metadatos de Photoshop se ajusta al Modelo de Datos y Serialización incluido en la propia especificación del XMP. Photoshop incrusta un paquete XMP con los metadatos del fichero dentro del recurso ‘0x0424’.

Un paquete XMP posee la siguiente estructura:

- Instrucciones de procesamiento y encabezado del paquete.
- Elemento superior de tipo ‘x:xmpmeta’, el cual contiene a su vez un único elemento ‘rdf:RDF’.
- Componentes del elemento ‘rdf:RDF’, de tipo ‘rdf:Description’.
- Los ‘rdf:Description’ contienen, a su vez, una o más propiedades XMP.

Estructura de un segmento XMP en un fichero .psd

Encabezado

El paquete XMP está compuesto por un encabezado estándar con el siguiente formato:

```
<?xpacket begin="ï»¿" id="W5M0MpCehiHzreSzNTczkc9d"?>
```

A continuación aparece el elemento 'xmpmeta', el cual resulta muy útil para ubicar los segmentos XML en una secuencia de datos. Su formato es:

```
<x:xmpmeta xmlns:x='adobe:ns:meta/'>
      <!--Metadatos serializados -->
</x:xmpmeta>
```

La declaración del xmpmeta lleva un atributo adicional cuya sintaxis es 'x:xmptk', el cual almacena el número de versión del manipulador de metadatos utilizado por Photoshop.

```
<x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="<!-- Versión-->">
```

Inmediatamente después aparece el elemento 'rdf:RDF', que tiene la siguiente estructura:

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
      <!--Metadatos serializados -->
</rdf>RDF>
```

Elementos 'rdf:Description'

El elemento 'rdf:RDF' puede contener cero o más elementos de tipo 'rdf:Description'. A continuación se muestra un 'rdf:RDF' con un único 'rdf:Description':

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
      <rdf:Description rdf:about=""
            xmlns:dc="http://purl.org/dc/elements/1.1/">
                  <!--...-->
      </rdf:Description>
</rdf:RDF>
```

A continuación se ofrece un ejemplo del cuerpo completo de la estructura de un segmento XMP: (**Ver figura 8**)


```
<?xpacket begin="ï»¿" id="W5M0MpCehiHzreSzNTczkc9d"?>
- <x:xmpmeta xmlns:x="adobe:ns:meta/" x:xmptk="3.1.1-111">
  - <rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
    - <rdf:Description rdf:about="" xmlns:dc="http://purl.org/dc/elements/1.1/">
        <dc:format>application/vnd.adobe.photoshop</dc:format>
      </rdf:Description>
    - <rdf:Description rdf:about="" xmlns:xap="http://ns.adobe.com/xap/1.0/">
        <xap:CreatorTool>Adobe Photoshop CS2 Windows</xap:CreatorTool>
        <xap:CreateDate>2008-04-15T16:22:24-04:00</xap:CreateDate>
        <xap:ModifyDate>2008-04-15T16:31:25-04:00</xap:ModifyDate>
        <xap:MetadataDate>2008-04-15T16:31:25-04:00</xap:MetadataDate>
      </rdf:Description>
    - <rdf:Description rdf:about="" xmlns:xapMM="http://ns.adobe.com/xap/1.0/mm/">
        <xapMM:DocumentID>uuid:2CE58F29290BDD11835BF722A298F35C</xapMM:DocumentID>
        <xapMM:InstanceID>uuid:2EE58F29290BDD11835BF722A298F35C</xapMM:InstanceID>
      </rdf:Description>
    - <rdf:Description rdf:about="" xmlns:tiff="http://ns.dobe.com/tiff/1.0/">
        <tiff:Orientation>1</tiff:Orientation>
        <tiff:XResolution>720000/10000</tiff:XResolution>
        <tiff:YResolution>720000/10000</tiff:YResolution>
        <tiff:ResolutionUnit>2</tiff:ResolutionUnit>
        <tiff:NativeDigest>256,257,258,259,262,274,277,284,530,531,282,283,296,301,318,319,529,
          532,306,270,271,272,305,315,33432;A19A05E1C08C408EB96AE5A053A64C8D</tiff:NativeDigest>
      </rdf:Description>
    - <rdf:Description rdf:about="" xmlns:exif="http://ns.adobe.com/exif/1.0/">
        <exif:PixelXDimension>14</exif:PixelXDimension>
        <exif:PixelYDimension>15</exif:PixelYDimension>
        <exif:ColorSpace>1</exif:ColorSpace>

          <exif:NativeDigest>36864,40960,40961,37121,37122,40962,40963,37510,40964,36867,36868,33434
          37380,37381,37382,37383,37384,37385,37386,37396,41483,41484,41486,41487,41488,41492,41
          41986,41987,41988,41989,41990,41991,41992,41993,41994,41995,41996,42016,0,2,4,5,6,7,8,9,1
          23,24,25,26,27,28,30;A6E55BE5922351E596C91DC118053725</exif:NativeDigest>
      </rdf:Description>
    - <rdf:Description rdf:about="" xmlns:photoshop="http://ns.adobe.com/photoshop/1.0/">
        <photoshop:ColorMode>3</photoshop:ColorMode>
        <photoshop:ICCProfile>sRGB IEC61966-2.1</photoshop:ICCProfile>
        <photoshop:History />
      </rdf:Description>
    </rdf:RDF>
  </x:xmpmeta>
  <?xpacket end="w"?>
```


Fig. 8 – Representación de la estructura de un segmento XMP. (Observar la presencia de metadatos referidos a la aplicación, hora de creación entre otros.)

4. – Segmento de información sobre capas y máscaras

La cuarta sección de un fichero nativo de Photoshop contiene información sobre las capas y máscaras de la imagen. Si no existen capas o máscaras en el fichero, esta sección se compone únicamente por el campo de longitud con sus 4 bytes establecidos en 0.

Longitud	Descripción
4	Longitud de la sección completa.
Variable	Información de capas
Variable	Información de la máscara de capa global

Las tablas siguientes muestran la organización de los segmentos de capas y máscaras, respectivamente:

Información de capas:

Longitud	Descripción
4	Longitud de la sección de información sobre las capas, redondeada a un múltiplo de 2.
2	Cantidad de capas. Su valor absoluto corresponde al número de capas.
Variable	Registros con información sobre cada capa.
Variable	Información sobre los canales de la imagen. Contiene uno o más registros por cada capa.

Información de máscara:

Longitud	Descripción
4	Longitud de la sección de información de máscara.
2	Espacio de color de superposición (sin documentar).
8	4 * 2 bytes: componentes de color.
2	Opacidad: 0 = transparente, 100 = opaco.
1	Tipo de máscara. 0 = Selección por color; 1 = Protección por color; 128 = valor almacenado por capa.
Variable	Relleno de ceros.

5 – Segmento de información de la imagen

La última sección de un archivo de Photoshop contiene los píxeles de la imagen. Los datos de la imagen se almacenan en orden lineal: primero toda la información del rojo, luego la del verde y finalmente la del azul. Cada plano se almacena en orden secuencial, sin bytes delimitadores.

Sección de información de la imagen

Longitud	**Descripción**
2	Método de compresión: *(ver debajo)*
	0 = Imagen en bruto
	1 = RLE
	2 = ZIP sin predicción
	3 = ZIP con predicción
Variable	Contenido de la imagen en orden secuencial (RRR GGG BBB)

El formato de **imágenes en bruto** almacena todos los datos de la imagen tal y como fueron captados por la lente digital, sin que la cámara le aplique filtros ni manipule sus atributos. Generalmente se asocia al método de compresión sin pérdidas, por lo cual las imágenes suelen ocupar bastante más espacio en disco que las comprimidas con el formato JPEG.

RLE o RLC son las siglas de *Run Length Encoding* o *Run Length Coding* (codificación por longitud del desplazamiento) es un método bastante simple de compresión de datos, que consiste en almacenar las secuencias repetidas como un único valor más la cantidad de repeticiones a realizar.

Conclusiones parciales sobre la sistematización del Adobe Photoshop

Como se había explicado al comienzo de este epígrafe, el formato de fichero de Adobe Photoshop está dividido en 5 grandes bloques de información, los cuales fueron detallados en sub epígrafes posteriores.

El estudio de este formato ha permitido conocer cuáles son los parámetros que se encuentran presentes de manera invariante en esos segmentos, así como caracterizarlos.

Se ha podido establecer que dichos parámetros siempre aparecen enunciados por el delimitador 8BIM el cual precede la aparición de dos bytes destinados al identificador único del recurso. Ello permite validar su legitimidad dentro de las especificaciones del formato, así como conocer la función que dentro de la estructura del estándar este juega.

Otras informaciones asociadas a la longitud del segmento y su información, permite conocer incluso la distancia a la que debemos esperar la aparición de un nuevo marcador.

Dada la organización de los bloques de datos, se ha decidido centrar la atención de esta investigación en los siguientes segmentos:

- Información de cabecera
- Recursos de imagen
- Información de capas y máscaras

En los restantes 2 segmentos (“Información del modo de color” e “Información de la imagen”), por su estructura, resulta difícil establecer patrones comunes para los demás archivos de Photoshop, pues la información del modo de color varía sustancialmente en dependencia del modo, y la información de la imagen es específica de los píxeles del propio archivo.

2.4.2 – Especificaciones de formato para ficheros JPEG

Introducción al JPEG

Como se ha mencionado anteriormente durante el transcurso de esta investigación, las siglas JPEG identifican al *Joint Photographic Experts Group*, nombre con que se identifica la comisión que creó el estándar de mismo nombre. Por lo general se suele asociar el término JPEG a un formato específico de imagen, cuando en realidad hace referencia a la norma estandarizada para la construcción de esas imágenes.

Por tal razón, JPEG no define un formato o extensión determinado para el intercambio de imágenes; para ello se han desarrollado diferentes implementaciones partiendo de la especificación nativa recogida en el ISO IEC IS 10918-1, la norma estandarizada internacionalmente. Las 2 más comunes son el JFIF *(JPEG File Interchange Format,* Formato de Intercambio de Archivos JPEG*)* y el Exif *(Exchangeable Image File Format,* Formato de Archivo de Imágenes Intercambiables), aunque recientemente el propio comité de mantenimiento de la norma ISO, propuso una nueva especificación denominada SPIFF (*Still Picture Interchange File Format,* Formato de Intercambio de Imágenes Estáticas).

Uno de los acápites de la norma JPEG es el método de compresión para las imágenes digitales. El JFIF se utiliza, por lo general, para almacenar, intercambiar y transmitir fotos a través de Internet, mientras el Exif es más comúnmente utilizado en cámaras digitales y otros dispositivos de captura de imágenes.

El algoritmo de compresión JPEG clasifica entre los denominados ‘con pérdida’, lo cual significa que la imagen obtenida después de la compresión no posee la misma calidad que la original. No obstante, una de sus principales ventajas es que permite ajustar el nivel de compresión, el cual es inversamente proporcional a la calidad de la imagen resultante.

El método para la codificación de imágenes en formato JPEG puede incluir varios procedimientos. A pesar de que su análisis no constituye objetivo de esta investigación, se hará una descripción elemental de los mismos, para que el lector conozca a qué se

refiere el texto cuando se haga referencia a las estructuras internas del formato de fichero que manipulan esta información.

Transformación del espacio de color: este procedimiento consiste en convertir la imagen de su modelo RGB original a otro conocido como YUV. Como se puede apreciar, este espacio de color cuenta con 3 componentes: Y (luminosidad o información de brillo), U (saturación, también conocida como cantidad o pureza de color) y V (tono o nombre del color, propiamente dicho). En la imagen resultante, la luminosidad está separada de la información de color (U y V).

Submuestreo: el submuestreo basa su funcionamiento en la reducción de la información de color con respecto a la de brillo, puesto que el ojo humano es más sensible a los cambios de luminosidad de una imagen que a los cambios de color. Si este método no se aplica, la imagen mantiene su espacio de color YUV; y si la imagen está en escala de grises, se puede optimizar más eliminando por completo la información de color (4:0:0).

Transformación discreta del coseno (DCT): luego de aplicado el submuestreo, cada componente de la imagen queda dividido en bloques de 8x8 píxeles, a los cuales se les aplica una función llamada 'transformación discreta de coseno' bidimensional, conocida como DCT. Al final, los colores quedan ubicados en la imagen en matrices de 8x8 u 8x16 píxeles.

Cuantificación: la cuantificación aprovecha el fenómeno descrito sobre la facilidad de percepción por el ojo humano de los cambios de brillo en una imagen. Esto ocurre en áreas relativamente grandes, pero no cuando el brillo varía frecuentemente en espacios reducidos *(variación de alta frecuencia)*, lo cual permite eliminar esas altas frecuencias sin sacrificar demasiado la calidad visual. Finalmente, la información queda organizada en las llamadas 'tablas de cuantificación'.

Codificación de entropía: es una forma especial de la codificación denominada 'sin pérdida de datos'. Este método selecciona los elementos de la tabla de cuantificación en forma de zigzag, luego agrupa los elementos que poseen frecuencias de aparición similares, para finalmente insertar ceros de codificación en ellos y aplicar el método de codificación de Huffman o una codificación aritmética para los valores restantes.

Formato interno de los ficheros JPEG

El formato JPEG puro rara vez se utiliza, fundamentalmente por la dificultad de implementar codificadores y decodificadores capaces de soportar toda la información que especifica el estándar.

Por tal motivo se hizo necesario dictar nuevos estándares que fueran más flexibles, sin que perdieran la fidelidad al JPEG nativo. El primero de estos esfuerzos fue el JFIF, surgido en 1992, y más recientemente el Exif y el SPIFF.

Marcadores JPEG

Una imagen JPEG está compuesta por una secuencia ordenada de **marcadores**, cada uno de los cuales comienza con el identificador hexadecimal '0xFF' seguido por un byte que especifica el tipo de marcador. Los marcadores de inicio y fin del fichero poseen únicamente estos 2 bytes; los restantes van seguidos por 2 bytes que almacenan la longitud del segmento de datos que le continúa.

ID hex	**Longitud (bytes)**	**Descripción**
0xFF	1	Delimitador del marcador
ID (ver lista)	1	Identificador del marcador
-	2	Longitud del segmento de información que le continúa (incluye los 2 bytes que la almacenan)
-	Variable	Información contenida en el marcador

El estándar ISO para JPEG:

La mencionada norma *'ISO IEC IS 10918-1: Information technology - Digital compression and coding of continuous-tone still images - Requirements and guidelines'* establece que un fichero JPEG puro posee la siguiente estructura interna: (**Ver figura 9**)

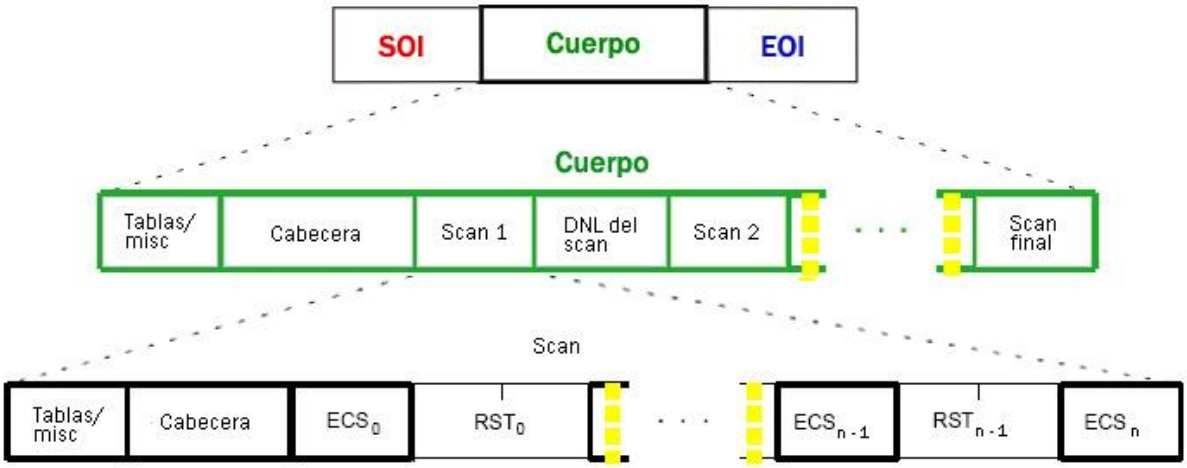


Fig. 9 – Representación de los bloques físicos de datos en los que se encuentra seccionado el estándar JPEG.

Según la estructura definida que se puede apreciar en la figura anterior, un archivo de intercambio gráfico en formato JPEG debe estar compuesto por un marcador de inicio de fichero (SOI, *Start Of Image*), un marco principal o cuerpo del fichero (*frame*) y un marcador que indica el final de la imagen comprimida (EOI, *End Of Image*). Dentro del cuerpo debe haber una o más estructuras de procesamiento de datos válidas (*scans*).

Cada uno de los *scans* del cuerpo posee a su vez una cabecera propia, uno o más segmentos de datos (ECS) y uno o más intervalos de reinicio (RST), en caso de que haya varias ocurrencias para el marcador. Las secciones de 'Tablas/misc' pueden contener información sobre tablas de cuantificación, compresión y otros datos.

El mismo estándar define los siguientes marcadores válidos para una imagen en formato JPEG:

ID hex	Nomenclatura	Descripción
Marcadores de inicio de cuerpo para especificar codificación Huffman (no diferenciales)		
0xFFC0	SOF_0	**Compresión DCT básica**
0xFFC1	SOF_1	**Compresión DCT extendida**
0xFFC2	SOF_2	**Compresión DCT progresiva**
0xFFC3	SOF_3	**Compresión sin pérdida (secuencial)**
Marcadores de inicio de cuerpo para especificar codificación Huffman (diferenciales)		
0xFFC5	SOF_5	**Compresión DCT secuencial (diferencial)**
0xFFC6	SOF_6	**Compresión DCT progresiva (diferencial)**
0xFFC7	SOF_7	**Compresión sin pérdida diferencial(secuencial)**
Marcadores de inicio de cuerpo para especificar codificación aritmética (no diferenciales)		
0xFFC8	JPG	**Reservado para extensiones JPG**
0xFFC9	SOF_9	**Compresión DCT secuencial extendida**
0xFFCA	SOF_{10}	**Compresión DCT progresiva**
0xFFCB	SOF_{11}	**Compresión sin pérdida (secuencial)**
Marcadores de inicio de cuerpo para especificar codificación aritmética (diferenciales)		
0XFFCD	SOF_{13}	**Compresión DCT secuencial (diferencial)**
0XFFCe	SOF_{14}	**Compresión DCT progresiva (diferencial)**
0xFFCF	SOF_{15}	**Compresión sin pérdida diferencial(secuencial)**
Especificación de tabla de Huffman		
0xFFC4	DHT	**Define las tablas de Huffman**
Condiciones para la codificación aritmética		
0xFFCC	DAC	**Define condiciones para la codificación aritmética**
Intervalo de reinicio		
0xFFD0 a 0xFFD7	RST_m *	**Cantidad *m* de reinicios en base a 8 (de 0 a 7)**

Otros marcadores		
0XFFD8	SOI*	**Inicio de la imagen (cabecera)**
0xFFD9	EOI*	**Fin de la imagen (terminación)**
0XFFDA	SOS	**Inicio de *scan***
0XFFDB	DQT	**Definición de tablas de cuantificación**
0xFFDC	DNL	**Definición del número de líneas**
0xFFDD	DRI	**Definición del intervalo de reinicio**
0xFFDE	DHP	**Definición de la progresión jerárquica**
0xFFDF	EXP	**Expansión de los componentes de referencia**
0xFFE0 a xFFEF	APP_n	**Reservados para segmentos de la aplicación**
0xFFF0 a xFFFD	JPG_n	**Reservados para extensiones JPEG**
0xFFFE	COM	**Comentario**
Marcadores reservados		
0xFF01	TEM*	**Para uso temporal durante la codificación aritmética**
0xFF02 a 0xFFBF	**RES**	**Reservados**

La familia de estándares JPEG: JFIF, Exif y SPIFF

Los formatos JFIF y Exif sustentan una estructura interna compatible con el ISO para el JPEG, pero más simple en su implementación. Las diferencias principales radican en la estructura del segmento principal de datos, del cual suelen omitirse elementos como los intervalos de reinicio y la reducción de la cantidad de *scans* o segmentos.

Toda imagen JPEG comienza con el identificador hexadecimal ‘0xFFD8’ y termina con ‘0xFFD9’. Entre estos delimitadores (llamados cabecera y terminación, respectivamente) pueden existir otros marcadores que se emplean para señalizar el comienzo de los diferentes bloques de información que componen la imagen.

El identificador ‘0xFFD8’ se conoce como SOI (Start Of Image o Inicio De Imagen), mientras el ‘0xFFD9’ se denomina EOI (End Of Image o Terminación De Imagen).

Los píxeles que componen la imagen se almacenan de manera continua como una secuencia ordenada de bytes dentro de un segmento del cuerpo del fichero. Al comienzo de dicho segmento se coloca el marcador ‘0xFFDA’, conocido como SOS (*Start Of Stream* o Inicio de Segmento). Luego del SOS comienza la información de la imagen hasta llegar al EOI.

Nota: En las posteriores interpretaciones del estándar ISO – IEC 10981-1 se decidió suprimir la existencia de múltiples *scans* en el cuerpo del fichero JPEG por la existencia de un único identificador: el SOS. Esta es justamente una de las características distintivas de las variantes de implementación mencionadas con anterioridad.

La estructura básica de un marcador es la siguiente: (**Ver figura 10**)

0xFF + Identificador del marcador (1 byte) + Longitud del segmento de datos (2 bytes) + Segmento de datos (n bytes).

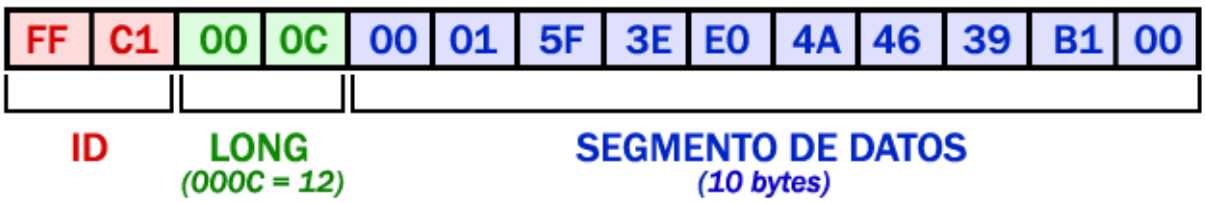


Fig. 10 – Representación de la estructura binaria de un marcador.

Si se tiene el marcador representado en la figura anterior, su identificador estaría compuesto por los bytes 0xFFC1 y la longitud del segmento de datos que contiene sería igual a 000C (cuyo valor decimal es 12). No obstante, esa longitud de 12 incluye los 2 bytes que la almacenan (o sea, 000C); por tanto, el segmento de datos posterior tiene una longitud de 10 bytes.

La estructura generalizada de los ficheros JPEG se describe en la siguiente imagen:
(Ver figura 11)

Fig. 11 – Representación de los bloques físicos de datos en los que se encuentran seccionadas las variantes de implementación del formato JPEG.

Luego del encabezado de la imagen (0xFFD8), el primer marcador que aparece es el denominado APP, empleado para identificar, entre otros parámetros, el tipo de formato que posee la imagen, seguido por 4 bytes que especifican de manera explícita dicho formato, como se expone a continuación:

Formato	Cabecera	Marcador APP	Longitud	Identificador
JFIF	0xFFD8	APP0 (0xFFE0)	2 bytes	0x4A464946 (“JFIF”)
Exif	0xFFD8	APP1 (0xFFE1)	2 bytes	0x45786966 (“Exif”)

De manera general, ni el JFIF ni el Exif difieren demasiado del JPEG nativo. Ambos están compuestos por marcadores para delimitar la información que contienen, y una miniatura almacenada de acuerdo a la especificación del JPEG. Por tanto, cualquier visor de ficheros JPEG podrá procesar imágenes en ambos formatos, indistintamente.

Las imágenes JFIF almacenan la miniatura (thumbnail) dentro del marcador APP0. El Exif, por su parte, la guarda embebida dentro del propio cuerpo de la imagen en un segmento JPEG adicional (desde el SOI hasta el EOI), de manera tal que, al diseccionar el cuerpo del fichero, aparecen en realidad 2 imágenes.

El formato Exif puede guardar esta miniatura en formato TIFF (Tag Image File Format) y no en JPEG. El TIFF fue diseñado por Adobe Systems Incoporated para ser usado fundamentalmente (pero no de manera exclusiva) en imágenes rasterizadas provenientes de cámaras, escáneres y otros dispositivos de captura digitales.

El estándar SPIFF

El SPIFF es, hasta ahora, la única implementación basada en la especificación original del JPEG que cuenta con un estándar reconocido. Forma parte de la tercera parte del ISO oficial del propio JPEG, y es el estándar que propone la comisión del mismo nombre para el almacenamiento de información JPEG.

Con la liberación del SPIFF, el Joint Photographic Experts Group pretende reemplazar al JFIF, actualmente el más utilizado en el mundo, a pesar de que posee varias limitaciones de formato.

El SPIFF introduce nuevas funcionalidades, tales como la adición de espacios de color adicionales y una forma nueva de introducir bloques de texto, entre otras. Además, proporciona compatibilidad con otras versiones, de manera tal que los decodificadores de imágenes JFIF puedan interpretar también el contenido de los ficheros SPIFF.

No obstante, de producirse el pretendido relevo del SPIFF, debe tardar aún cierto tiempo, porque el JFIF posee una enorme difusión en todo el mundo, además de que cubre las necesidades de la mayoría de los usuarios, es ligero y fácil de implementar.

Conclusiones parciales sobre la sistematización del JPEG

El formato de fichero de JPEG está dividido en 3 grandes bloques de información, como se detalla en este epígrafe.

Ha sido posible conocer, mediante el estudio de este formato, cuáles son los parámetros que se encuentran presentes en esos bloques y así caracterizarlo.

Se ha podido establecer que dichos parámetros siempre aparecen enunciados por el delimitador FF el cual precede la aparición de un byte destinado al identificador único del marcador. Ello permite validar su legitimidad dentro de las especificaciones del formato, así como conocer la función que dentro de la estructura del estándar este juega.

Otras informaciones asociadas a la longitud del segmento y su información, permite conocer incluso la distancia a la que debemos esperar la aparición de un nuevo marcador.

La validación de la unicidad en la aparición de determinados marcadores ha revestido un importante resultado en la recuperación de los restos de ficheros, aportándonos elementos de juicio a la hora de conocer el número de restos a los que nos estaríamos enfrentando.

Dada la organización de los bloques de datos, se ha decidido centrar la atención de esta investigación en esas mismas estructuras localizadas en todo el cuerpo de ese tipo de ficheros, conociendo de antemano el orden de aparición de muchos de estos marcadores.

Igualmente hemos podido sistematizar a la par de las definiciones binarias nativas de este tipo de fichero, las de sus dos variantes más utilizadas tanto en el mundo de la fotografía digital (Exif), como en el del diseño digital apoyado en herramientas informáticas (JFIF). Estos dos por su parte abordan las principales regularidades de este formato, que complementado con las normas oficiales del formato, han permitieron caracterizar el formato JPEG.

2.5 – Conclusiones del Capítulo

En el transcurso de la investigación, fue posible constatar que la caracterización del formato de un fichero, es una tarea complicada y que demanda de un serio y prolongado proceso de recopilación de información, experimentación y sistematización de normas y estándares internacionales.

Cada formato permite su especialización en versiones, donde los usuarios y proveedores de sistemas que los interpretan, pueden introducir variantes a la implementación general. Las estructuras pueden variar de una versión a otra del fichero o incluso de la aplicación que lo genera.

La identificación de ficheros JPEG y PSD puede, durante los procesos forenses de búsqueda, retornar resultados falsos positivos, pues cuando Photoshop trabaja con capas procedentes de archivos JPEG, incrusta dentro de su estructura la información binaria de esa imagen; y de la misma forma, al exportar el contenido de un fichero .psd a formato de imagen, Photoshop escribe varios recursos de imagen propios de manera embebida dentro del segmento de datos del JPEG generado.

Aunque ha quedado demostrado que la amplitud del fenómeno requiere de continuidad en su estudio, a fin de comprenderlo en su totalidad, finalmente se ha podido establecer una determinada generalidad e invariabilidad de las estructuras de los ficheros objeto de esta investigación, que garantizarán la implementación de un algoritmo que permita su recuperación.

Ninguna de las normas estudiadas durante el transcurso de este trabajo especifica el orden en que deben encontrarse los diferentes segmentos de datos dentro de los ficheros. No obstante, el estudio y la observación de diferentes poblaciones de estos ficheros, permitieron establecer de manera experimental un orden consecutivo de aparición para cada fichero.

En el caso de los ficheros JPEG, el orden propuesto puede presentar cambios entre un fichero y otro, en dependencia de la implementación del estándar que haya sido empleada en la codificación de la imagen **(ver anexos 1 y 2).**

CAPÍTULO 3
SOBRE LA PROPUESTA DE SOLUCIÓN TÉCNICA

3.1- Introducción

El estudio realizado sobre las diferentes herramientas existentes para el análisis informático forense permitió corroborar las potencialidades que hacen de EnCase la herramienta líder en este terreno a nivel mundial. Otro elemento que influye en la decisión de desarrollar la solución a este problema científico sobre EnCase es el hecho de la experiencia en su uso que acumula la Sección de Informática Forense.

De igual manera, los especialistas de la Sección de Informática Forense de la División de Criminalística realizaron una selección de los tipos de ficheros con mayores frecuencias de aparición en los hechos delictivos ocurridos hasta el momento. Esa selección es la que servirá de base al desarrollo del presente trabajo de diploma, y está compuesta por los siguientes tipos de archivos:

- Imágenes JPEG (*.jpg)
- Archivos de Adobe Photoshop (*.psd)

A continuación se describen las principales características de EnCase Enterprise 4.20, así como de la propuesta de solución técnica: el *script* **Arqueólogo**.

3.2 - El entorno de desarrollo de EnCase Enterprise: EnScript

La instalación predeterminada de EnCase Enterprise incluye un conjunto de *scripts* que ejecutan determinadas acciones sobre los dispositivos de evidencia. Su código puede ser modificado a conveniencia, de la misma manera en que se pueden crear nuevas secuencias de comandos.

La plataforma incluye una colección no muy extensa pero útil de bibliotecas de clases por defecto que sirven para cubrir las necesidades de los desarrolladores, muchas de las cuales pueden ser igualmente modificadas.

Cuenta con un compilador que interpreta el código y notifica si encuentra algún error sintáctico o semántico, e incorpora además varias clases para el manejo de interfaces gráficas de usuario (GUIs) a través de formularios con sus respectivos componentes visuales.

EnCase Enterprise 4.20 no necesita *frameworks* o plataformas de soporte. Es un entorno sumamente ágil, portable y apto para ser ejecutado sobre cualquier sistema operativo de la familia Windows 98/ME/2x/XP con un mínimo de 128 Mb de memoria RAM y 1.0 MHz de velocidad de procesamiento.

Su interfaz, pese a ser bastante intuitiva, requiere de cierto entrenamiento o familiarización previos que permitan al investigador hacer un uso correcto de la misma. Y para la modificación y/o construcción de *scripts*, es necesario tener conocimientos de programación orientada a objetos y del propio lenguaje EnScript, lo cual no debe ser problema para quienes estén familiarizados con la sintaxis de otros lenguajes de alto nivel como C++, Java o C#.

No obstante, EnScript 4 no soporta algunas de las principales fortalezas de la programación orientada a objetos, como los tipos genéricos, aunque el desarrollador puede escribir sus propias clases genéricas. Reconoce, por otro lado, los mismos tipos de datos nativos (enteros, cadenas, caracteres, números con coma flotante…)

3.5 - Componentes utilizados

El método de trabajo de EnCase se basa en el empleo de casos, los cuales funcionan de manera análoga al concepto de un proyecto. Cada vez que el examinador forense desee analizar un dispositivo o evidencia, debe tener un caso abierto o crear uno nuevo para montar dicho dispositivo.

Uno de los *scripts* predeterminados se llama **File Finder**, y proporciona una funcionalidad lejanamente similar a la propuesta de solución de este trabajo. Su código itera sobre la estructura física del fichero evidencia en busca de información de inicio y final de un grupo determinado de ficheros.

Tomando como base su principio de funcionamiento, se comenzó a desarrollar el nuevo *script*, sustituyendo la información de cabecera por algunos de los parámetros invariantes identificados.

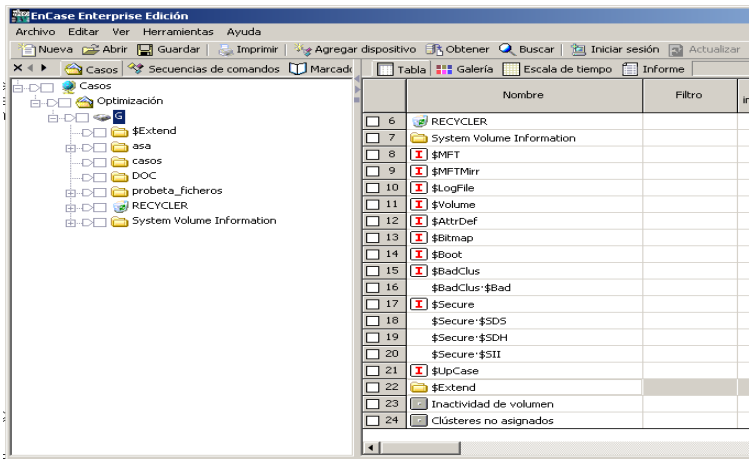
El entorno de desarrollo de EnCase incorpora de manera predeterminada una serie de clases y bibliotecas de clases cuyas funcionalidades resultan muy útiles a la hora de desarrollar un script.

Entre esas componentes sobresalen las clases EntryClass y FileClass, las cuales proporcionan un conjunto de procedimientos y métodos muy potentes para el análisis de la evidencia presente en un caso.

3.5.1 - La clase EntryClass

Todo lo que exista físicamente en un dispositivo abierto por EnCase se representa a través de la clase EntryClass, y se llaman **entradas**. El árbol de directorios, los subdirectorios y ficheros contenidos, así como el propio caso y el dispositivo en sí, son modelados por EnCase como instancias de la clase EntryClass.

Esta clase cuenta con métodos que permiten acceder tanto a la estructura lógica como física de la evidencia, analizar su contenido a nivel de bytes y consultar sus atributos (nombre, ruta original, hora de creación/modificación, ubicación física/lógica…).El espacio no asignado del disco también se representa a través de una entrada en EnCase. Lo que hace este entorno de trabajo es recopilar todos los clústeres no asignados del sistema de ficheros y mostrarlos como un único archivo concatenado para su análisis.



3.5.2 - La clase FileClass

La FileClass se encarga, por su parte, de la gestión de las operaciones de lectura/escritura en disco.Cuenta con procedimientos que permiten desplazarse dentro de la secuencia de datos de un fichero o de una entrada y realizar diferentes acciones sobre las mismas. En el caso de las entradas, las posibles acciones a ejecutar son de solo lectura, puesto que EnCase se asegura de que la manipulación de la evidencia sea totalmente legítima e inofensiva, en aras de garantizar su integridad y confiabilidad.

Posee un método muy útil definido con la siguiente sintaxis: **Find**(**const String** &secuencia, **int** longAcierto, **int** opciones), el cual devuelve un valor *booleano* (lógico) como resultado de la búsqueda en una posición determinada dentro de un fichero del sistema de archivos o de una entrada en el caso. Otra de sus potencialidades es la posibilidad de abrir un búfer de escritura en el disco e introducirle información en diferentes formatos y empleando distintos juegos de caracteres (codificaciones).

3.5.3 - El script FileFinder

EnCase incorpora de manera predeterminada un script diseñado para recuperar algunos tipos de ficheros en el espacio no asignado de un dispositivo. Pero su algoritmo de búsqueda se basa exclusivamente en la identificación de las cabeceras y terminaciones, por tanto, si algún fichero ya fue parcialmente sobrescrito, no será posible recuperarlo.

```
File Finder (v4)

/* File Finder (v4)  (c) 2004, Guidance Software, Inc.
*/

include "GSI_LogLib"
include "GSI_BulkCopyLib"

class FileFormatClass : NameListClass {
  String Name,
         Header,
         Footer,
         Extension;
  bool   IsPicture,
         IsCustom;
```

Fig. 12: Declaración de la clase 'FileFormat', empleada por el *script* FileFinder para manejar la información de los ficheros. Obsérvese cómo los únicos parámetros que se toman en cuenta son el inicio, la terminación y la extensión.

Por otra parte, se puede apreciar además en el fichero de configuración **FileSignatures.ini**, donde se almacenan las firmas que EnCase utiliza para la identificación de los diferentes tipos de ficheros, que la herramienta solo toma en consideración la estructura de las cabeceras de los archivos. **(Ver figura 13**)

```
Notepad++ - C:\Archivos de programa\EnCase4\FileSignatures.ini
File Edit Search View Format Language Settings Macro Run TextFX Plugins Window ?
FileSignatures.ini
117  1  0
118     Adobe PhotoShop 8BPS\x00\x01\x00\x00\x00\x00                    1          psd;pdd
119  1  0
120     Compound Document File  \xD0\xCF\x11\xE0\xA1\xB1\x1A\xE1\x00\x00                        1
121  1  0
122     Corel PhotoPAINT Tone Map   \x43\x4F\x52\x45\x4C\x20                        1          maj
123  1  0
124     Corel Presentation Exchange \x52\x49\x46\x46....\x43\x4D\x58                    1   1
125  1  0
126     CorelDRAW Graphic   \x52\x49\x46\x46....\x43\x44\x52                    1   1          cd:
127  1  0
128     Enhanced Metafile Graphic   \x01\x00\x00\x00\x58\x00\x00\x00                        1
129  1  0
130     GIF GIF8[79]a   1                   1   1          gif
131  1  0
132     Interchange File Format FORM    1                       1          iff
133  1  0
134     JPEG Image  \xFF\xD8\xFF[\xFE\xE0]\x00                    1   1          jpg;jpeg;jpe
135  1  0
Normal text file | nb char : 6304 | Ln : 1  Col : 1  Sel : 0 | UNIX | UCS-2 little Endian | INS
```

Fig. 13: Fragmento del contenido del fichero 'FileSignatures.ini', en el cual se almacenan las firmas de identificación para múltiples tipos de ficheros. Aquí se puede observar cómo se toma en cuenta solamente la información de cabecera del archivo.

3.5.4 - La clase Bookmark

Otra de las funcionalidades que incorpora EnCase es el almacenamiento de *bookmarks*, conocidos también como referencias o marcadores.

El funcionamiento de las marcas de EnCase es análogo al de los marcadores de páginas de un libro: cada vez que el **FileFinder** encuentra un archivo, agrega una referencia a la entrada que contiene el acierto de búsqueda, especificando además las posiciones exactas de inicio y final del resultado.

Ello proporciona una ayuda enorme al investigador, pues le permite remitirse inmediatamente al lugar donde comienza el acierto de búsqueda encontrado. De lo contrario, tendría que desplazarse manualmente por todo el espacio no asignado –que puede ocupar varios gigabytes de información- hasta llegar a la ubicación deseada. **(Ver figura 14**)

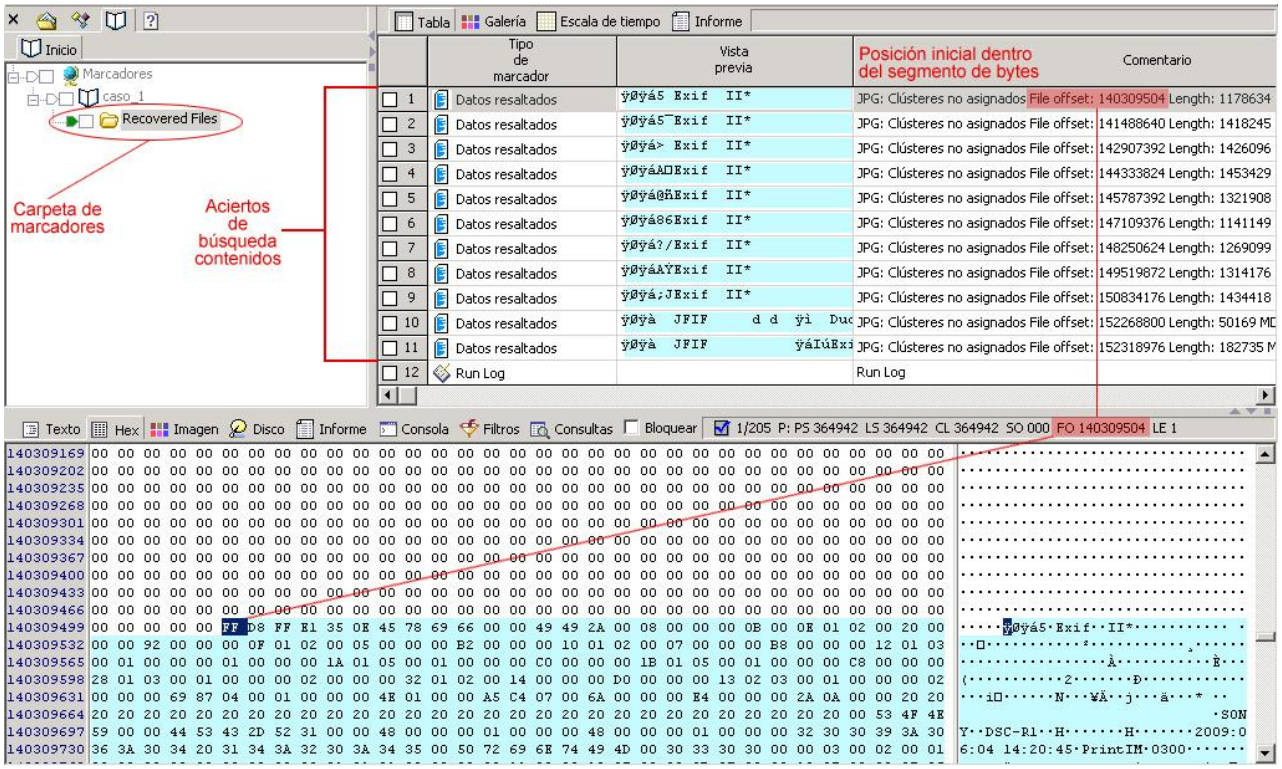


Fig. 14: Vista de las carpetas de marcadores de EnCase luego de ejecutado el *script* FileFinder. Se puede apreciar el resaltado del acierto de búsqueda en color azul celeste.

3.5.5 - Descripción del funcionamiento del script 'Arqueólogo'

Para el funcionamiento del Arqueólogo fue necesario diseñar un conjunto de clases de soporte que permitieran manipular los diferentes parámetros que necesita.

Por ese motivo se crearon las siguientes clases de datos auxiliares:

- **FormatoFichero**: posee un atributo '*descripcion'* que registra el tipo de fichero en cuestión.
- **FormatArray**: representa un arreglo de tipo 'FormatoFichero' para almacenar los formatos existentes (JPEG y PSD).
- **GestorFichero**: se encarga de las operaciones de lectura de los ficheros que contienen la información de los formatos.
- **GeneradorReportes**: se encarga de las operaciones de escritura en disco, fundamentalmente la creación de los reportes con los resultados de la búsqueda.
- **Conversor**: clase con miembros estáticos que permite realizar conversiones entre los sistemas numéricos binario, hexadecimal y decimal.

También fue necesario crear 2 tipos que heredan de la clase nativa de EnScript 'DialogClass' para presentar al usuario las interfaces necesarias para interactuar con las funcionalidades del *script*. Esas clases son:

- **Formulario**: contiene la interfaz principal que se levanta al iniciar el script, con todas las opciones de búsqueda.
- **VentanaAyuda**: muestra un diálogo con las instrucciones necesarias para realizar búsquedas personalizadas. Especifica cuál es la sintaxis que se debe emplear para que el algoritmo reconozca los parámetros introducidos.

Toda secuencia de comandos en EnScript necesita, para poder manipular los casos y dispositivos abiertos, una declaración de la clase 'MainClass' (clase principal), nativa del lenguaje de programación, y un método llamado 'Main' dentro de ella, que es el encargado de ejecutar las acciones iniciales.

La clase principal del Arqueólogo posee una instancia del tipo ‘FormatArray’ -descrito anteriormente- llamada “formatos”, y otra de tipo ‘GestorFichero’. Al iniciar el script, el objeto gestor lee automáticamente el contenido de los ficheros “marcadores_jpeg” y ”recursos_psd”, y utiliza esa información para inicializar los arreglos de formatos que almacena el objeto ‘formatos’. Si no se encuentran esos ficheros, se notifica al usuario sobre el evento ocurrido y se detiene la ejecución del código.

El script ‘Arqueólogo’ permite realizar 2 tipos de búsqueda: predeterminada y personalizada. En el primer caso, el funcionamiento es análogo para ambos tipos de ficheros (JPEG y PSD), por lo cual se describirán los métodos empleados de manera general. En el segundo caso, el algoritmo difiere ligeramente de los otros 2.

Algoritmos de búsqueda predeterminados (para ficheros JPEG y PSD)

JPEG

El método de recuperación de ficheros JPEG se basa en la identificación de los marcadores internos que los componen. Primeramente, el algoritmo selecciona las entradas a analizar en dependencia de los parámetros especificados por el usuario (clústeres asignados, no asignados, entradas seleccionadas y/o todo el contenido del caso). Luego comienza a desplazarse por el contenido del fichero buscando la existencia de algún marcador válido.

Para verificar la autenticidad de un marcador, el algoritmo invoca al método **PosMarcadorJPEG(String &marcador),** el cual recibe como parámetro la secuencia encontrada. Si dicha secuencia se encuentra en alguna posición del arreglo de marcadores JPEG, el método retorna esa posición; de lo contrario, devuelve el valor -1 y se reanuda la búsqueda del siguiente marcador.

En caso de tratarse de un marcador genuino, se realiza una llamada a la función **BuscarMarcadoresSigPorDesp()**, la cual procede a encontrar todos los marcadores válidos que aparezcan n bytes hacia delante a partir de la posición del marcador actual (el valor de n equivale a la longitud del marcador encontrado, y se obtiene de los 2 bytes siguientes al identificador del marcador, como ya se explicó).

Mientras existan marcadores una vez omitida la longitud n, el algoritmo calculará el tamaño del nuevo marcador, agregará la posición actual y el nombre del marcador válido a una lista de aciertos y desplazará el puntero interno del fichero n posiciones.

En caso de que en la posición esperada no se encuentre el próximo marcador válido, se invoca al método **BuscarMarcadoresSigSinDesp()**, el cual reanuda la búsqueda del siguiente marcador que aparezca, sin importar dónde se encuentre.

Fue necesario realizar esta validación para tomar en cuenta el posible nivel de fragmentación del sistema de archivos del dispositivo evidencia, porque en caso de que el fichero se encuentre fragmentado, sus segmentos no estarían ordenados uno a continuación del otro. En caso de encontrar otro marcador válido, se reanuda la ejecución del **BuscarMarcadoresSigPorDesp()**, y así hasta llegar al final del archivo.

Una vez analizado todo el contenido del fichero, se procesan los aciertos de búsqueda encontrados. Para ello se invoca al método **RetornarAciertos(EntryClass &entrada)**, el cual recorre la lista de aciertos y se extraen las secuencias de marcadores mientras exista consecutividad entre ellas.

Los marcadores JPEG se encuentran ubicados en el arreglo de formatos en el mismo orden en que deben aparecer dentro de un fichero. Ello permite recorrer la lista de aciertos y, mientras las secuencias almacenadas sean consecutivas dentro del arreglo, deberán formar parte del mismo fichero. Cuando se encuentre un marcador JPEG cuya posición en el arreglo no sea posterior a la del último encontrado, se asume como válida la secuencia encontrada hasta ese punto, y se adiciona un *bookmark* a la misma. Luego se continúa la extracción de secuencias consecutivas a partir de ese punto de ruptura, asumiendo que forman parte de otro fichero.

Al momento de agregar el marcador, se calcula el nivel de probabilidad de que sea realmente un fichero JPEG. Para ello se determina qué por ciento representa la cantidad de recursos JPEG encontrados del total de recursos existentes en el arreglo.Las imágenes Exif que almacenan su miniatura en formato JPEG incrustan esa vista previa como una imagen embebida en su propio segmento de datos.

Por esa razón, el algoritmo suele identificar 3 aciertos de búsqueda en lugar de uno: la secuencia que va desde el inicio de la imagen (0xFFD8) hasta el inicio de la miniatura (0xFFD8), la estructura JPEG completa de la miniatura hasta su final (0xFFD9) y finalmente el resto de la estructura interna de la imagen original. **(Ver figura 15**)

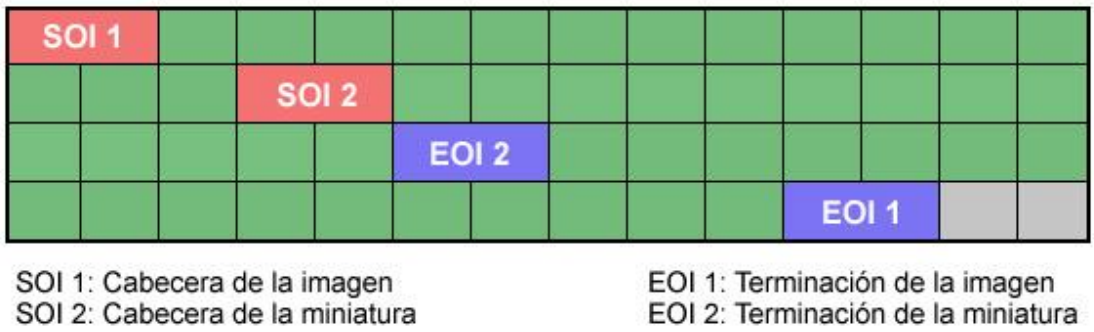


Fig. 15: Aproximación gráfica a la estructura de una imagen Exif con miniatura JPEG.

PSD

El algoritmo de búsqueda de documentos de Adobe Photoshop está basado en el mismo principio que emplea el algoritmo para las imágenes JPEG: la identificación de los delimitadores de recursos ‘8BIM’ en lugar de los marcadores ‘0xFFD8’ y el desplazamiento hacia delante buscando los siguientes recursos.

Cuando aparece una secuencia ‘8BIM’ en el contenido del fichero, se leen los 2 bytes siguientes donde se almacena el identificador único del recurso. Ese identificador se busca dentro del arreglo de recursos y, de ser válido, se aplica el mismo procedimiento descrito para las imágenes JPEG: se localizan los próximos recursos válidos que se encuentren en los desplazamientos esperados, y de no existir en esas posiciones, se localizan independientemente de su ubicación.

Al llegar al final del archivo, se realiza también una llamada a la función **RetornarAciertos(EntryClass &entrada)**, la cual genera los *bookmarks* correspondientes.

Algoritmo de búsquedas personalizadas

Para realizar búsquedas definidas por el investigador, fue necesario definir 2 palabras reservadas a manera de sintaxis para que el método pudiera extraer los parámetros que precisa.

La idea general del algoritmo se basa en la introducción de un determinado grupo de secuencias con los desplazamientos que deben existir entre ellas, para que el método sea capaz de identificarlas.

Las estructuras reservadas que se definieron son las siguientes:

- SEC: empleado para especificar un criterio de búsqueda
- DESP: empleado para especificar la cantidad de bytes que debe haber antes de encontrar la siguiente secuencia.

En el caso de las secuencias, se debe emplear la sintaxis de EnScript para el reconocimiento de expresiones regulares. Como la búsqueda se realizará sobre bytes, se debe colocar el token '\\x' delante de cada byte que componga la secuencia.

Por tanto, un criterio de búsqueda correctamente construido debería tener la siguiente estructura:

```
S: \\x4E\\x6F\\x72\\x6D\\x61\\x6C
D: 512
S: \\x78\\x2D\\x6E\\x73
D: 2048
S: \\x4D\\x53\\x57\\x6F\\x72\\x64\\x44\\x6F\\x63
D: 1036
S: \\x57\\x6F\\x72\\x64\\x2E\\x44\\x6F\\x63\\x75\\x6D\\x65\\x6E\\x74
```

Esos valores se almacenan en un arreglo; el algoritmo procede entonces a buscar cada elemento del arreglo dentro del contenido del fichero, comenzando por el primero. Cuando encuentra alguno, se desplaza hacia delante la cantidad de bytes especificada en la posición siguiente del arreglo, y verifica la existencia del próximo parámetro.Al llegar al final del fichero, se invoca al método **RetornarAciertos(EntryClass &entrada)**, cuyo funcionamiento fue descrito en secciones anteriores.

Algoritmo para generar reportes

Una vez concluida la búsqueda ejecutada por el usuario, se procede a generar el/los reporte(s) en los formatos seleccionados previamente.

Si solo se desea mostrar los resultados en consola, el objeto 'generadorReportes' recorre la carpeta de marcadores y muestra su contenido en la salida de la consola de EnCase. De lo contrario, el objeto generador crea una carpeta llamada "Reporte pericial" en la ruta seleccionada por el usuario, y coloca dentro el/los fichero(s). En caso de existir archivos de reportes generados previamente, éstos no serán sobrescritos, pues el método **GenerarFicheroReporte()** extrae la fecha y hora actuales del sistema y las adiciona al nombre del fichero generado.

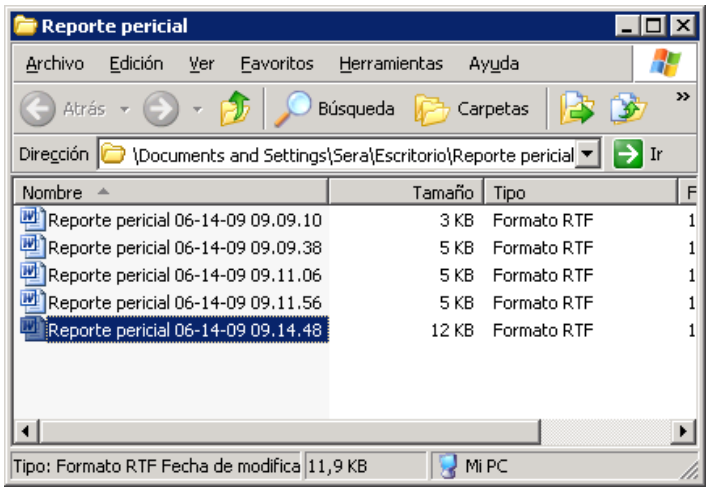
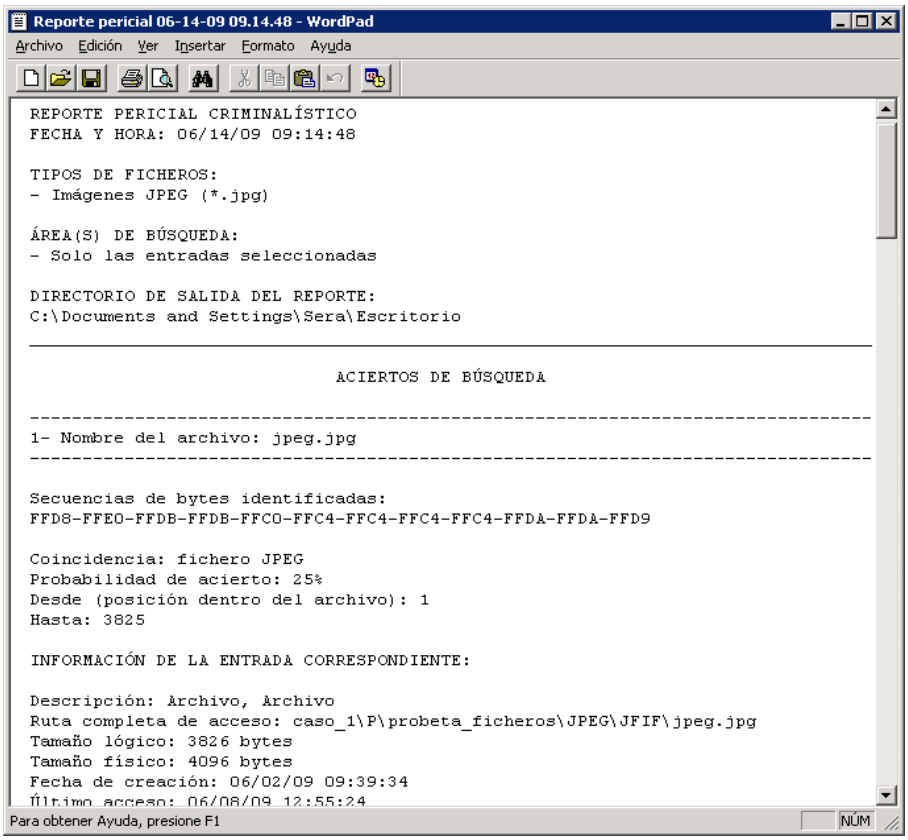


Fig. 16: Ejemplo de un directorio de salida para los reportes generados. Obsérvese cómo la inclusión de la fecha y la hora en el nombre del fichero previene que los ya existentes sean sobrescritos.

El fichero contiene un encabezado estándar, luego muestra la hora a la cual se ejecutó la búsqueda, los parámetros que se utilizaron (tipo de búsqueda, tipos de ficheros buscados, áreas de búsqueda…) y finalmente recorre la carpeta de marcadores que contiene todos los aciertos de búsqueda generados, para incluir en el fichero la información de cada *bookmark*.



Reporte pericial 06-14-09 09.14.48 - WordPad

Archivo Edición Ver Insertar Formato Ayuda

```
REPORTE PERICIAL CRIMINALÍSTICO
FECHA Y HORA: 06/14/09 09:14:48

TIPOS DE FICHEROS:
- Imágenes JPEG (*.jpg)

ÁREA(S) DE BÚSQUEDA:
- Solo las entradas seleccionadas

DIRECTORIO DE SALIDA DEL REPORTE:
C:\Documents and Settings\Sera\Escritorio
_____________________________________________________________________________

                              ACIERTOS DE BÚSQUEDA

-----------------------------------------------------------------------------
1- Nombre del archivo: jpeg.jpg
-----------------------------------------------------------------------------

Secuencias de bytes identificadas:
FFD8-FFE0-FFDB-FFDB-FFC0-FFC4-FFC4-FFC4-FFC4-FFDA-FFDA-FFD9

Coincidencia: fichero JPEG
Probabilidad de acierto: 25%
Desde (posición dentro del archivo): 1
Hasta: 3825

INFORMACIÓN DE LA ENTRADA CORRESPONDIENTE:

Descripción: Archivo, Archivo
Ruta completa de acceso: caso_1\P\probeta_ficheros\JPEG\JFIF\jpeg.jpg
Tamaño lógico: 3826 bytes
Tamaño físico: 4096 bytes
Fecha de creación: 06/02/09 09:39:34
Último acceso: 06/08/09 12:55:24
```

Para obtener Ayuda, presione F1 NÚM

Fig. 17: Captura de pantalla realizada a un fichero de reporte con extensión .rtf generado por 'Arqueólogo'.

3.6 – Conclusiones del Capítulo

En el largo proceso de sistematización y análisis de las características que debía reunir esta solución, se pudo conocer de los elevados niveles de certeza que necesita el trabajo criminalístico para llegar a buen término.

Utilizando las reducidas, pero no despreciables, posibilidades de desarrollo e implementación que brinda EnCase y su módulo EnScript, se modeló un algoritmo en capacidad de operar sobre las estructuras invariantes caracterizadas en el Capítulo # 2.

Aunque muchas interrogantes quedan por investigar y experimentar, algunas que no pudieron contemplarse en este trabajo de diploma, aun así se considera esta investigación como un avance en lo referido a la localización de ficheros eliminados, incluso los parcialmente sobrescritos, en la contraposición y verificación de residuos, candidatos a ser en medio de la investigación partes integrantes de un mismo fichero.

Se trabajo en el fenómeno de la fragmentación a la hora de localizar estructuras internas consecutivas, pero que debido a su dislocación y dispersión en el disco, pudieron no conservar físicamente el orden lógico descrito en sus especificaciones. A pesar de no cumplirse su consecutividad y orden, algún otro fragmento pudiera permanecer alojado en el espacio disponible de otro sector.

Finalmente se ha propuesto una solución que resuelve la localización de ficheros y fragmentos de ellos en los formatos JPEG y PSD mediante el empleo de parámetros estructurales que fueron sistematizados a lo largo de esta investigación. Se logró la generación de un Reporte Pericial Criminalístico como un resultado de consideración dado su impacto en el trabajo de los investigadores de la SIF.

RECOMENDACIONES

1- Continuar optimizando los tiempos de ejecución del *script*.
2- Probar el rendimiento en condiciones reales de trabajo y sometido a los volúmenes reales a procesar en los casos.
3- En lo adelante, incorporarle las funcionalidades para la ubicación y recuperación de otros formatos de ficheros de interés pericial, en especial para la familia de los Office.
   (**DOC, PPT, XLS**)
4- Recomendamos que por su contenido, esta tesis sea puesta a la disposición y tomada como material de estudio, para aquellos que pretendan implementar un script similar o una investigación sobre el tema abordado.

REFERENCIAS

Referencias bibliográficas:

1. Óscar López , Haver Amaya, Ricardo León y Coautora: Beatriz Acosta, "INFORMÁTICA FORENSE : GENERALIDADES, ASPECTOS TÉCNICOS Y HERRAMIENTAS, "http://www.criminalistaenred.com.ar/Informatica_F.html

2. Dr.Joaquín Silva, "CIRUGÍA,Glosario*, "http://encolombia.com/cirugia14399_glosario47.htm.

3. MORENO GONZÁLEZ, Luis. (1990). *Manual de Introducción a la Criminalística*, México: Porrúa.

4. Alberto Hornero, "Análisis Forense Digital, The Sleuth Kit. [1/2] « blog.ahornero.com," http://ahornero.wordpress.com/2009/05/25/analisis-forense-digital-the-sleuth-kit/

5. Ídem

6. Alberto Hornero, "Análisis Forense Digital, The Sleuth Kit. [1/2] « blog.ahornero.com," http://ahornero.wordpress.com/2009/05/25/analisis-forense-digital-the-sleuth-kit/

7. Ídem

8. *Sistema de ficheros GNU/Linux*Observatorio Tecnológico del Ministerio de Educación, Política Social y Deporte de España, http://observatorio.cnice.mec.es/modules.php?op=modload&name=News&file=article&sid=549&mode=thread&order=0&thold=0&POSTNUKESID=71e64c9ba56ad2bfa28de218c2bcbc5c

9. "fragmentación - Definición - WordReference.com," http://www.wordreference.com/definicion/fragmentaci%C3%B3n

10. "Desfragmentación - Wikipedia, la enciclopedia libre," http://es.wikipedia.org/wiki/Desfragmentaci%C3%B3n

11. "Guidance Software, Inc. - TransUnion Selects Guidance Software's EnCase Enterprise to Conduct Its Network-wide Digital Investigations," http://investors.guidancesoftware.com/releasedetail.cfm?ReleaseID=270731

12. "Law Technology News - Home," http://www.lawtechnews.com/r5/showkiosk.asp?listing_id=2003877

13. "ENCASE FORENSIC 6.1, presente en BOLIVIA," http://www.nobosti.com/spip.php?article497

14. "EnCase Search Technology Validated by Federal Court in Contested Electronic Discovery Ruling | Reuters," http://www.reuters.com/article/pressRelease/idUS117

BIBLIOGRAFÍA

Bibliografía:

1. "Análisis Forense de Sistemas GNU/Linux, Unix", http://www.loquefaltaba.com/documentacion/forense/index.html.

2. "Digital Evidence: Standards and Principles, by SWGDE and IOCE (Forensic Science Communications, April 2000)", http://www.fbi.gov/hq/lab/fsc/backissu/april2000/swgde.htm.

3. "Comportamiento de Virus en plataformas Windows - www.elhacker.net", http://www.elhacker.net/comportamiento-virus.htm.

4. "Electronic Crime Scene Investigation: A Guide for First Responders", http://www.ncjrs.gov/txtfiles1/nij/187736.txt.

5. "Computación forense", http://www.globalteksecurity.com/pdf/01%20-%20moduloi_computacionforense.pdf

6. "Computer forensics introduction to incident response and investigation", http://www.sans.org/reading_room/whitepapers/incident/computer_forensics_introduction_to_incident_response_and_investigation_of_windows_nt/2000_647?show=647.php&cat=incident.

7. "A case for forensics tools in crossdomain data transfers", http://www.sans.org/reading_room/whitepapers/forensics/a_case_for_forensics_tools_in_crossdomain_data_transfers_1126?show=1126.php&cat=forensics.

8. "Revista 96. Administración de sistemas de ficheros", http://www.acis.org.co/fileadmin/Revista_96/dos.pdf.

9. "Introducción a la informática forense en entornos Windows 1ª parte - www.elhacker.net", http://www.elhacker.net/InfoForenseWindows.htm.

10. "Law.com - Electronic Data Discovery Primer", http://www.law.com/jsp/article.jsp?id=1029171611801.

11. "Forensic Toolkit", http://www.accessdata.com/forensictoolkit.html.

12. "The Sleuth Kit", http://www.sleuthkit.org/sleuthkit/desc.php&rurl=translate.google.com.cu&usg=ALkJrhjUgRM0LqsdVxcdZcvyybMhUPerDw.

13. "The Sleuth Kit: Documents", http://www.sleuthkit.org/sleuthkit/docs.php

14. "The Sleuth Kit: History", http://www.sleuthkit.org/sleuthkit/history.php

15. "EnCase", http:// en.wikipedia.org/wiki/EnCase

16. “HELIX”, http://.e-fense.com%2Fh3-enterprise. php&sl=en&tl=es&hl=es&ie=UTF-8.

17. “Prevención, detección e investigación del fraude en entornos virtuales”, http://www.cybex.es/es/servicios_herramientas.htm.

18. “The Sleuth Kit”, http://www.sleuthkit.org/.

19. “X-Ways Software”, http://www.x-ways.net/forensics/.

20. “X-Ways Forensics: Integrated Computer Forensics Software”, http://www.x-ways.net/forensics/

ANEXOS

Anexo 1: Orden de aparición de los marcadores dentro del cuerpo de un fichero JPEG.

ID marcador	Descripción
ffd8	**Cabecera de imagen JPEG**
ffe0	Marcador APP0 de aplicación (JFIF)
ffe1	Marcador APP1 de aplicación (Exif)
ffe2	Marcador APP2 (reservado para la aplicación)
ffe3	Marcador APP3 (reservado para la aplicación)
ffe4	Marcador APP4 (reservado para la aplicación)
ffe5	Marcador APP5 (reservado para la aplicación)
ffe6	Marcador APP6 (reservado para la aplicación)
ffe7	Marcador APP7 (reservado para la aplicación)
ffe8	Marcador APP8 (reservado para la aplicación)
ffe9	Marcador APP9 (reservado para la aplicación)
ffea	Reservado para la aplicación
ffeb	Reservado para la aplicación
ffec	Reservado para la aplicación
ffed	Reservado para la aplicación
ffef	Reservado para la aplicación
ffdb	Tabla de cuantificación
ffc0	Compresión DCT básica
ffc4	Tabla de Huffman
ffdd	Definición del intervalo de reinicio
ffd0	Intervalo de reinicio 1
ffd1	Intervalo de reinicio 2
ffd2	Intervalo de reinicio 3
ffd3	Intervalo de reinicio 4
ffd4	Intervalo de reinicio 5
ffd5	Intervalo de reinicio 6

ffd6	Intervalo de reinicio 7
ffd7	Intervalo de reinicio 8
ffdc	Definición del número de líneas
ffcc	Tabla de codificación aritmética
ffc1	Compresión DCT extendida
ffc2	Compresión DCT progresiva
ffc3	Compresión sin pérdida (secuencial)
ffc5	Compresión DCT secuencial
ffc6	Compresión DCT progresiva
ffc7	Compresión sin pérdida diferencial (secuencial)
ffc8	Reservado para extensiones JPG
ffc9	Compresión DCT secuencial
ffca	Compresión DCT progresiva
ffcb	Compresión sin pérdida (secuencial)
ffcd	Compresión DCT secuencial
ffce	Compresión DCT progresiva
ffcf	Compresión sin pérdida (secuencial)
fffe	Comentario
fff0	Reservado para extensiones JPEG
fffd	Reservado para extensiones JPEG
ffde	Definición de la estructura jerárquica
ffdf	Componente(s) de referencia expandido(s)
ff01	Operaciones aritméticas temporales
ffda	Inicio de información de la imagen
ffd9	Terminación de imagen JPEG

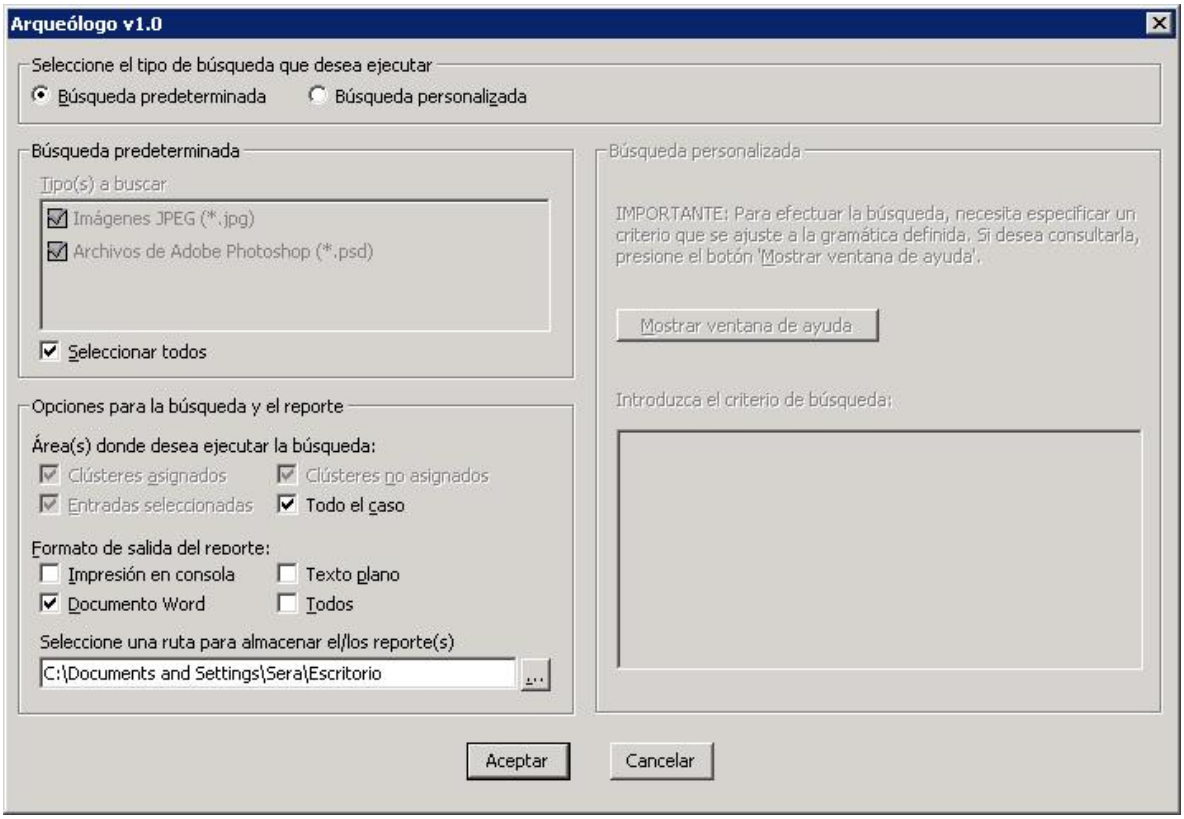
Anexo 2: Orden de aparición de los recursos dentro del cuerpo de un fichero PSD.

ID recurso	Descripción
38425053	Cabecera de fichero .psd
0425	CaptionDigest. Seguridad RSA o MD5
0424	Metadatos del fichero en formato XMP
03ED	Estructura 'ResolutionInfo'
0426	Escala de impresión
03EE	Nombres de los canales alfa
0415	Nombres alfa en Unicode
03EF	Estructura 'DisplayInfo'
041D	Identificadores alfa
040D	Ángulo de vista global
0419	Altitud global de la imagen
03F3	Banderas de impresión
040A	Información de copyright
2710	Información sobre banderas de impresión
03F5	Información de tonalidades en modo color
03F8	Funciones de transferencia de color
0400	Información de estado de capas
0402	Información sobre grupos de capas
0430	Recurso no documentado por Adobe
042D	Recurso no documentado por Adobe
0408	Información de guías y reglas
041E	Lista de URLs
041A	Slices
0428	Relación de aspecto
040F	Perfil ICC
0411	Perfil ICC sin etiqueta
0414	Número base para generar IDs de capas
040C	Vista previa (incluido a partir de la versión 5)
0421	Versión del manipulador de metadatos

0422	Información Exif 1
0FA0	Recurso no documentado por Adobe
0FA1	Recurso no documentado por Adobe
03F0	Caption de la imagen
03F1	Información del borde de la imagen
03F2	Color de fondo
03F4	Información de escala de grises y multicanales
03F6	Información de tonalidades en modo dual
03F7	Funciones de transferencia de escalas de grises y multicanales
03F9	Funciones de transferencia de tonos duales
03FA	Información de imagen en tono dual
03FB	Información referente a escala de grises (blanco y negro)
03FC	Identificador obsoleto
03FD	Opciones EPS
03FE	Información de máscara rápida
03FF	Identificador obsoleto
0401	Ruta de trabajo (sin salvar)
0403	Identificador obsoleto
0404	Registros IPTC
0405	Modo de imagen para formatos de imágenes en bruto
0406	Calidad JPEG
0409	Vista previa de la imagen (solo para Photoshop 4)
040B	URL
040E	Muestreadores de color
0410	Marca de agua
0412	Efectos de visualización de capa
0413	Vista de tonos medios
0416	Contador de la tabla de color indexado (paleta)
0417	Índice de transparencia
041B	URL del flujo de trabajo
041C	Salto a XPEP
0423	Información Exif 2
0429	Composición de capa

042A	Colores alternos en tono dual
042B	Colores alternativos de muestra
0BB7	Nombre de ruta adjunta

Anexo 3: Captura de pantalla del script Arqueólogo en tiempo de ejecución.



Anexo 4: Captura de pantalla de la ventana de ayuda del script Arqueólogo en tiempo de ejecución.

GLOSARIO DE TÉRMINOS

- **Acierto de búsqueda**: resultado positivo en la ejecución de un algoritmo de localización. Coincidencia favorable entre el criterio de búsqueda y el resultado obtenido.

- **Clúster**: Agrupación de sectores consecutivos que conforman la unidad mínima de asignación que realiza un sistema de archivos sobre un disco.

- **Espacio no asignado**: unidades de asignación de un disco (sectores o clústeres) que no se encuentren asignados a los archivos vigentes de un sistema de archivos. Puede contener archivos o fragmentos de ellos que todavía no se hayan sobrescrito.

- **Estándar puro o nativo**: especificación originaria para un determinado formato de ficheros. Conjunto original de normas establecidas que aseguran la comprensión de un patrón.

- **Fichero de interés criminalístico**: formato de fichero que adquiere relevancia dada la regularidad de su aparición en el trabajo de enfrentamiento de los criminalistas.

- **Fichero persistente**: que está vigente, que es direccionable al estar registrado en la tabla de asignación de un sistema operativo.

- **Fragmentación del sistema de archivos**: incapacidad del sistema de archivos para almacenar la información de un mismo fichero en sectores contiguos.

- **Identificador hexadecimal**: secuencia en notación hexadecimal, que permite delimitar un bloque de información en el cuerpo de un fichero.

- **Imagen rasterizada**: también llamada imagen de mapa de bits. Estructura o fichero de datos que representa una rejilla rectangular de pixeles o puntos de color, denominada raster, que permite dibujar la información de la imagen.

- **Invariabilidad**: Propiedad de los marcadores internos de un fichero que asegura la presencia del mismo en su estructura binaria, bajo cualquier implementación o estándar.

- **Marca de seguimiento/bookmark**: referencia o índice utilizado por el EnCase para devolver de forma organizada un acierto de búsqueda. Permite acceder con inmediatez a la ubicación precisa del segmento de datos del acierto.

- **Marcador JPEG/parámetro interno**: delimitación, bandera, identificador hexadecimal que enuncia la presencia de un bloque de información o de datos dentro de la estructura binaria de un fichero JPEG.

- **Metadato**: información que describe a otro dato o información. Atributos de un archivo.

- **Píxel**: acrónimo de la expresión anglosajona 'picture element'. Representa la unidad mínima de color que conforma a una imagen digital.

- **Recurso PSD/parámetro interno**: Delimitación, bandera, identificador hexadecimal que enuncia la presencia de un bloque de información o de datos dentro de la estructura binaria de un fichero PSD.

- **Sector**: Sección física de un disco duro que contiene generalmente 512 bytes de información más sus caracteres de formato y encabezado. Cuña, segmento comprendido entre dos líneas radiales del disco.

SIGLAS

- **C4PDF**: siglas en inglés de Code of practices for Digital Forensics. Código de Prácticas para Informática Forense.

- **CRC**: siglas en inglés de Cyclic Redundancy Check. Comprobación de redundancia cíclica.

- **DIVICO**: División de Investigaciones Criminales y Operaciones.

- **EOI**: siglas en inglés de End Of Image. Fin de la imagen.

- **EXIF**: siglas en inglés de Exchangeable Image File Format. Formato de Archivo de Imágenes Intercambiables.

- **ext2**: siglas en inglés de Second Extended Filesystem. Segundo sistema de archivos extendido.

- **ext3**: siglas en inglés de Third Extended Filesystem. Tercer sistema de archivos extendido.

- **FAT**: siglas en inglés de File Allocation Table. Tabla de Asignación de Ficheros.

- **FBI**: siglas en inglés de Federal Bureau of Investigation. Buró Federal de Investigaciones.

- **GUIs**: siglas en inglés de graphical user interface. Interfaz gráfica de usuario.

- **HPFS**: siglas en inglés de High Performance File System. Sistema de archivos de altas prestaciones.

- **INTERPOL**: siglas en inglés de International Criminal Police Organization. Organización Internacional de Policía Criminal.
-

- **ISO**: siglas en inglés de International Organization for Standardization. Organización Internacional para la Normalización.

- **IT**: siglas en inglés de Information Technology. Tecnologías de la Información.

- **JFIF**: siglas en inglés de JPEG File Interchange Format. Formato de Intercambio de Archivos JPEG.

- **JFS**: siglas en inglés de Journaling File System.

- **JPEG**: siglas en inglés de Joint Photographic Experts Group. Grupo de Expertos en Fotografía Digital.

- **KB**: siglas en inglés de KiloByte. Unidad de almacenamiento de información.

- **MD5**: siglas en inglés de Message-Digest Algorithm 5. Algoritmo de Resumen del Mensaje 5.

- **MS-DOS**: siglas en inglés de MicroSoft Disk Operating System. Sistema operativo de disco de Microsoft.

- **NTFS**: siglas en inglés de New Technology File System. Nueva tecnología de sistemas de archivos.

- **PSD**: siglas en inglés de PhotoShop Document. Documento de Photoshop.

- **RAID**: siglas en inglés de Redundant Array of Inexpensive Disks. Arreglo Redundante de discos baratos.

- **RAM**: siglas en ingles de Random Access Memory. Memoria de Acceso Aleatorio.

- **RDF**: siglas en inglés de Resource Description Framework. Entorno de Descripción de recursos.

- **RGB**: siglas en inglés de Red, Green, Blue. Esquema de color que utiliza como base los tres colores primarios (Rojo, Verde y Azul).

- **SIF**: Sección de Informática Forense

- **SO**: Sistema Operativo.

- **SOI**: siglas en inglés de Start Of Image. Inicio de la imagen.

- **SOS**: siglas en inglés de Start Of Stream. Inicio de Segmento.

- **SPIFF**: siglas en inglés de Still Picture Interchange File Format. Formato de Intercambio de Imágenes Estáticas.

- **TCT**: siglas en inglés de The Coroner Toolkit. Nombre de la herramienta forense.

- **TIFF**: siglas en inglés de Tag Image File Format. Formato de archivo de imágenes con etiquetas.

- **XML**: siglas en inglés de eXtensible. Markup Language. Lenguaje de marcas extensible.

- **XMP**: siglas en inglés de eXtensible Metadata Platform o Plataforma Extensible de Metadatos.