

BORRANDO ARCHIVOS

Conceptos básicos sobre la dinámica del funcionamiento de los sistemas de archivo

Introducción

Uno de los activos más importantes de las organizaciones es la información y como tal, éstas deben establecer diferentes estrategias para enfrentar la variedad estados y posibilidades que exhibe la inseguridad informática. En este sentido, estrategias como backups, servidores espejo, sitios alternos y redundancia paralela, son parte fundamental de la administración de los repositorios de datos de las empresas actualmente.

Sin embargo, la susceptibilidad o inevitabilidad de la falla es una constante que en cualquier momento, sugiere un reto para los administradores de los sistemas o tecnologías de información. Estas situaciones por lo general, involucran información en tránsito o almacenada, que dada su criticidad y alto valor estratégico y táctico, se hace necesario contar con ella de manera inmediata para resolver alguna situación particular, o posteriormente cuando ha ocurrido un posible siniestro y se requiera recabar información sobre el hecho, para tratar de identificar los posibles autores.

En este sentido, estudiar y revisar los sistemas de archivo, (o en inglés *file systems*) estructuras donde se almacena la información, resulta de interés tanto para los cuerpos directivos como para los niveles técnicos, pues conociendo cuáles son sus ventajas y limitaciones propias, pueden establecer medidas de seguridad y control que disminuyan los riesgos de pérdidas de datos que podrían llegar a ser irreparables.

En consecuencia, este breve documento plantea una introducción a los sistemas de archivos y algunas características de los generalmente utilizados como FAT (File Allocation Table), NTFS (New Technology File System) y EXT2 (basado en Unix File System). Así mismo, ofrece una revisión básica de características de seguridad y recuperación que sugieran opciones ante posibles fallas en sus estructuras. Es importante anotar, que este artículo no busca profundizar en las características propias y detalles técnicos de cada sistema de archivos, sino ilustrar de manera práctica sus características y posibilidades ante situaciones que puedan comprometer los datos.

¿Qué es un sistema de archivos?

Los sistemas de archivos proveen mecanismos a los usuarios para almacenar datos de manera jerárquica: archivos y directorios. De esta forma un sistema de archivo consiste de estructuras y datos de usuarios que son organizados de tal forma que la máquina o computadora conoce donde encontrarlos. En este sentido, los sistemas de archivos tienen procedimientos específicos y estructuras que pueden ser utilizadas bien sea para almacenar un archivo en un diskette o diez mil archivos en un arreglo de discos. [Carrier 2005, pág.173-174].

Siguiendo el modelo referencial planteado por Carrier[2005, pág.174-175] existen cinco categorías de datos de análisis para profundizar y comparar los diferentes tipos de sistemas de archivos: sistema de archivos, contenido, metadatos, nombre de archivos y aplicación.

La categoría *sistema de archivo* contiene la información general del sistema de archivos. Dicha información hace referencia a estructuras de datos disponibles, unidad de almacenamiento de datos, versión del sistema de archivos, fecha de creación, en pocas palabras, la información base para usuario de dicho sistema. Por ejemplo para los sistemas FAT, esta información se puede encontrar en el sector de arranque (o *boot sector*). Este sector esta localizado en el primer sector del disco y es parte de un área reservada por el sistema de archivos. [idem]

La categoría *contenido* incluye la localización donde se ubican los archivos, los bloques o sectores asignados (unidades de almacenamiento) en el disco. Estas unidades de almacenamiento pueden tener dos estados: asignado (*allocated*) y no asignado (*nonallocated*).[idem]

Cuando se habla de *metadatos*, estamos hablando de la información que describe los archivos; es decir, datos que describen los datos. Esta categoría contiene información como: dónde esta ubicado el contenido del archivo, qué tan grande es el archivo, la hora y fechas en que el archivo fue leído o modificado y el control de acceso al mismo. Ejemplos de estructuras de datos en esta categoría son las entradas de la FAT, el *Master File Table* de NTFS y los *i*-nodos de los sistemas Ext2y Ext3. [idem]

La categoría *nombre de archivos* considera los nombres mismos de los archivos y la dirección donde se encuentra los metadatos. La importancia de esta categoría reside en poder ubicar el directorio raíz donde se encuentran los archivos, para a partir de allí revisar los metadatos e identificar la ubicación del mismo y proceder con su recuperación, si es del caso. [idem]

Finalmente, la categoría *aplicación* contiene estructuras de datos que establecen características especiales adicionales en los sistemas de archivos. Dentro de los ejemplos en esta categoría tenemos características como *user quota statistics* y *file system journal*. La primera como la capacidad de sistema de archivos de asignar y restringir el uso de espacio en disco y la segunda como la capacidad del sistema de archivos de guardar registros de eventos ocurridos en él, con el propósito de contar con información para reconstrucción ante eventualidades. [idem]

A manera de resumen podemos observar en la siguiente tabla, las características por cada uno de los sistemas de archivos objeto de esta revisión: [Tomado de: Carrier 2005. Pág. 208]

	CATEGORÍAS				
	<i>Sistema de Archivos</i>	<i>Contenido</i>	<i>Metadatos</i>	<i>Nombre de archivo</i>	<i>Aplicación</i>
Ext X	Superbloque, descriptor de grupo	Bloques, bloques bitmap	Inodos, inodos bitmap, atributos extendidos	Entradas de directorios	Journal
FAT	Sector de arranque, FSINFO	Clusters, FAT	Entradas de directorio, FAT	Entradas de directorios	No Aplica
NTFS	\$Boot, \$Volume, \$AttrDef	Clusters, \$Bitmap	\$MFT, \$MFTMirr, \$DATA, \$ATTRIBUTE_LIST, \$SECURIY_DESCRIPTOR	\$FILE_NAME, \$IDX_ROOT, \$IDX_ALLOCATION, \$BITMAT	Disk quota, journal, change journal

Breve descripción de los sistemas de archivo: FAT, NTFS y Ext2

FAT

File Allocation Table o tabla de localización de archivos, dicho de otra manera un conjunto de entradas en una tabla que contiene el nombre del archivo, el tamaño en bytes, la dirección del sector de inicio del archivo en el cluster (conjunto de sectores físicos del disco) y otra información de metadatos. La estructura de FAT es utilizada para identificar el siguiente cluster en un archivo y establecer el estado de asignación de un cluster: disponible o asignado. A la fecha existen diferentes versiones de FAT: FAT12, FAT16 y FAT32. La mayor diferencia entre ellas es el tamaño de las entradas en la estructura de FAT.

La estructura de las entradas de la FAT, que se observa en el siguiente gráfico, está conformada por tres secciones físicas a saber: *área reservada*, que incluye datos que pueden estar en la categoría sistema de archivos; *área de FAT*, que contiene la estructura tanto primaria como de respaldo de la FAT y finalmente el *área de datos*, que contiene los clusters que serán asignados para almacenar los archivos y directorios.

Área reservada	Área de FAT	Área de datos
----------------	-------------	---------------

Estructura física de las entradas en la FAT [Tomado de: Carrier 2005, pág.213]

NTFS

New Technologies File System, diseñado por Microsoft con el fin de ofrecer mayor confiabilidad, seguridad y soporte a los usuarios de los sistemas windows NT, windows 2000, windows xp y windows server. Uno de los conceptos más importantes del diseño de NTFS es que los datos críticos son asignados a archivos. Esto incluye los datos administrativos del sistema de archivos que son típicamente ocultados por otros sistemas de archivos. El concepto central de NTFS es la tabla maestra de archivos o *Master File Table* – MFT, lugar donde se almacena toda la información acerca de los archivos y directorios en el volumen. Las entradas en la MFT son de 1kbyte, pero solamente los 42 primeros bytes tienen propósito definido. El resto de bytes almacena estructuras pequeñas de datos para fines específicos como: nombre del archivo, contenido del archivo.

La estructura de una entrada de la MFT, denominada *file record* – registro de archivo, que se muestra a continuación, se le otorga una dirección inicial basada en su localización en la tabla, iniciando por cero. La primera entrada en la tabla es \$MFT, donde se describe en el disco la localización de ella misma. La localización de la MFT es establecida por el boot sector, el cual siempre se ubica en el primer sector del sistema de archivos.

Encabezado de Entrada de MFT	Atributos	Atributos	Atributos	Espacio no utilizado
------------------------------	-----------	-----------	-----------	----------------------

Estructura física de las entradas en la MFT [Tomado de: Carrier 2005, pág.275]

Ext2

Este sistema de archivos toma su diseño de UFS (Unix File System), el cual fue diseñado para ser más rápido y confiable. Establece copias de importante estructuras de datos las cuales son almacenadas a lo largo del todo el sistema de archivos y toda la información asociada con archivos está localizada de tal forma que la cabeza de lectura del disco duro no se desplace mucho cuando requiera leerlos. La estructura de este sistema de archivos inicia con un área reservada, seguida de secciones llamadas grupos de bloques. Todos los grupos de bloques, excepto el último, contienen el mismo número de bloques, los cuales son usados para almacenar nombres de archivos, metadatos y contenidos de los mismos.

Área reservada	Grupo 0	Grupo 1	Grupo 2	Grupo 3	Grupo 4
<-Bloques por grupo ->					

Estructura física de Ext2 con cinco grupos [Tomado de: Carrier 2005, pág.400]

Como hemos podido observar, cada sistema de archivos contiene básicamente las estructuras planteadas por Carrier [2005, pág.174-175], unos con más detalles que otros, pero definitivamente con la complejidad y especificación que son inherentes a cada uno de ellos. De esta manera, cuando ocurre alguna situación crítica, pérdida de datos, corrupción de índices, entre otras, la recuperación y análisis de cada caso dependerá del grado de profundidad con que se conozca y maneje cada sistema de archivos.

Sin un conocimiento detallado y experiencia requerida para adelantar un análisis de sistemas de archivos, es altamente probable que los resultados de dicha revisión no sean los más adecuados. Por tanto, conocer las características propias de cada sistema de archivos es una competencia básica para los administradores de las plataformas de cómputo y para los investigadores y analistas forenses, pues es allí donde su pericia y cuidado deben hacerse presente para obtener el ciento por uno en su diligencia de recuperación y análisis.

Características y borrado en FAT, NTFS y Ext2

Con el conocimiento base de las estructuras y características base de los sistemas de archivo objeto de este documento, procedemos a revisar elementos propios de cada uno de ellos, con el fin de revisar estrategias para recuperación de datos ante borrados o fallas en el momento de generar los archivos. Las técnicas indicadas a continuación no son confiables ante eliminación física de datos sobre discos magnéticos, dado que se ha sufrido una alteración del medio físico.

Borrado de archivos en FAT

Cuando se elimina una referencia de un archivo de la tabla FAT o lo que comúnmente se llama borrado de archivos, se efectúan algunas operaciones que detallamos a continuación. [Adaptado de Carrier 2005, pág.246]

Paso 1: Se lee el sector de boot desde el sector 0 del disco, para localizar las estructuras de FAT, el área de datos y el directorio raíz.

Paso 2: Se localiza el archivo en las entradas del directorio raíz, buscando la que corresponde con el nombre del archivo a borrar.

Paso 3: Se procesa la entrada y se ubica el cluster donde inicia el archivo.

Paso 4: Se utiliza la estructura de la FAT para determinar el encadenamiento de clusters para el archivo.

Paso 5: Se actualizan las entradas de la FAT registradas para el archivo en cero.

Paso 6: Finalmente se retira la asignación de la entrada del directorio para el archivo cambiando el primer byte por el valor 0xe5.

Cumplidos los pasos comentados no existe referencia formal en la estructura de la FAT sobre el archivo eliminado. Sin embargo, los datos continúan disponibles en el área de datos, hasta que el sistema operacional, conociendo que es espacio disponible lo utilice para sus labores administrativas, o que cuando se guarde un nuevo archivo éste utilice algunas de las entradas disponibles como las recientemente ajustada a cero.

Borrado de archivos en NTFS

En este sistema de archivos el borrado es una operación más delicada pues implica la modificación de múltiple estructuras del sistema de archivo, dada las características de confiabilidad y seguridad que se implementaron en NTFS. A continuación un resumen de las operaciones que ocurren en el borrado de un archivo en NTFS: [Adaptado de Carrier 2005, pág.346]

Paso 1: Se lee el primer sector del sistema de archivos y se procesa el sector de boot para determinar el tamaño del cluster, la dirección de inicio de la MFT y el tamaño de las entradas de la MFT.

Paso 2: Se revisa la primera entrada de la tabla MFT, donde se encuentra la referencia a ella misma, la cual es procesada y así determinar la configuración de la MFT, la cual se encuentra en atributo \$DATA.

Paso 3: Se procede a buscar en el directorio raíz, a través de dos atributos \$INDEX_ROOT e \$INDEX_ALLOCATION, hasta encontrar el archivo a eliminar. Allí se actualiza la fecha de último acceso al archivo.

Paso 4: Se procesa el resultado obtenido en la búsqueda para el atributo \$INDEX_ROOT y se ubica la dirección dentro de la MFT del archivo.

Paso 5: Se remueve la entrada del índice (que se actualiza con cada nueva entrada en la MFT) y las entradas en el nodo de creación del archivo son removidas y sobre escritas como estaban originalmente (disponibles). Se actualiza las fechas de última escritura, modificación y acceso para el archivo.

Paso 6: Se desasigna la entrada en la MFT, eliminando la bandera de indicación: “en uso”. Así mismo se procesa el atributo \$BITMAP del archivo en cuestión y se le asigna el valor 0.

Paso 7: Los atributos no residentes (nombre del archivo, entre otros) de la entrada en la MFT en cuestión son procesados y los correspondientes clusters son configurados con el estado “no asignados” en el atributo \$BITMAP del archivo.

Paso 8: En cada uno de los pasos anteriores se ha venido registrando los cambios en el *file system journal* (característica extendida de registro de acciones sobre el sistema de archivos) en el atributo de la MFT \$LogFile.

Como podemos observar el borrado en este sistema de archivos exige un detalle y conocimiento más elaborado para saber cómo se borran los archivos, dónde se efectúan los cambios e incluso cómo se van registrando los cambios ocurridos, para posibles

recuperaciones de los mismos en el futuro. Nuevamente y consistente con las características de seguridad de NTFS, no posible que existan dos objetos dentro del sistema con el mismo identificador, por lo que el registro de objetos ante el sistema operacional debe obedecer a una sincronización entre los atributos del sistema de archivos y las directivas del sistema operacional.

Borrado de archivos en ExtX

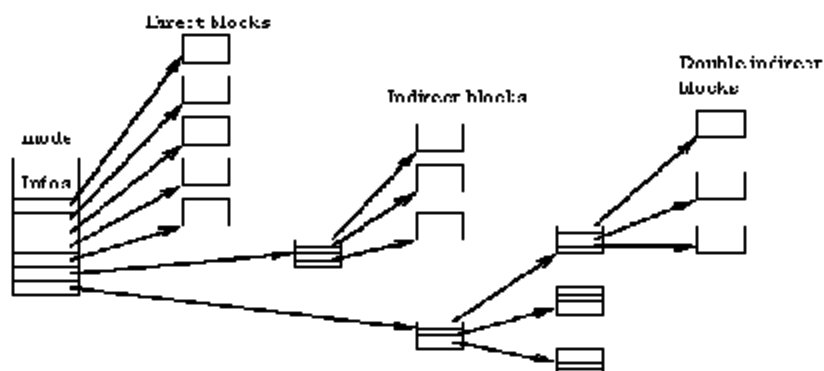
En los sistemas de archivo ExtX, el borrar un archivo exige una serie de pasos que modifican estructuras claves como los i-nodos (conjunto de datos relacionados con el archivo utilizados por el sistema de archivos para conocer las características del mismo), para mantener consistente la integridad del mencionado sistema. Es importante, aclarar que cuando se borra un archivo en sistemas Ext2, los valores de los i-nodos no desaparecen, luego los apuntadores a los bloques de los datos existen. Caso diferente cuando es Ext3, dado que los apuntadores no permanecen. [Carrier 2005, pág.446]

Con este contexto inicial, desaparecer un archivo en sistema ExtX se adelanta de la siguiente manera: [Adaptado de Carrier 2005, pág.443-444]

1. Se lee el superbloque (área reservada de 1024 bytes al inicio del sistema de archivos. Copias de esta estructura están típicamente almacenadas en el primer bloque de cada grupo de bloques).
2. Luego, se lee tabla de descriptores de grupo de bloques, ubicados en los bloques 2 y 3 del sistema de archivo. Esta tabla contiene la descripción de cada grupo de bloques (i-nodos y archivos)
3. Se localiza el archivo a eliminar en el directorio raíz y se procesa el i-nodo 2, que generalmente se asocia con el grupo de bloques 0.
4. Seguidamente se ubica en el grupo de bloques, la tabla de Inodos, para identificar dentro de la estructura de directorios, el bloque que le corresponde al archivo a eliminar.
5. Ahora se lee el contenido del directorio raíz para el número de bloque identificado e identificamos el valor del I-nodo del archivo.
6. Se calcula luego, donde esta ubicado el inodo del archivo, dentro del grupo de bloques.
7. Una vez ubicado el archivo dentro del grupo de bloque, se procede a desasignar y cambiar el estado del I-nodo, lo cual lleva a la actualización del MAC time (Modificación, Acceso y Creación) del archivo. Estos cambios se registran en el sistema de journaling del sistema (si aplica).
8. Finalmente se desasignan los bloques ocupados por el archivo, actualizando el bit en el bloque que indica su uso a 0 y el apuntador al bloque en el I-nodo es reiniciado.

Este proceso de borrado de archivos en sistemas ExtX, requiere un conocimiento detallado del direccionamiento directo, indirecto y doblemente indirecto que maneja el sistema de archivos de Linux, como se ilustra en la gráfica No.1.

Gráfica 1. Manejo de I-nodos en sistemas Ext. [Tomada de: Nelson et al (2004) Guide to computer forensics and investigations. Cap.4 Thomson. Course Technology]



En este sentido la recuperación de archivos en sistema ExtX, no se puede limitar a la reconstrucción de la tabla de inodos o ubicación de bloques, sino que requiere en muchas ocasiones recurrir a técnicas como el carving (“escarbar”) dentro de datos ubicados en los bloques, reconociendo que existen los direccionamientos directos en indirectos que deben ser reconstruidos para conocer la ubicación exacta del archivo a recuperar.

Recomendaciones finales

Como hemos podido observar en este breve documento, tratando de evitar al máximo terminología técnica (cuando es posible), el conocer la manera como se borran los archivos, nos puede ilustrar la manera de recuperarlos. Sería pretencioso decir que con estas líneas estamos en capacidad de recuperar información que se comprometa en los sistemas de archivo revisados, pero lo que podemos afirmar es que, se establece un marco general de comprensión de la dinámica propia de los sistemas de archivo generalmente utilizados.

Los sistemas de archivo son la parte central del almacenamiento de los datos y por tanto deberían ser estudiados como parte integral de la formación de los ingenieros en las áreas de tecnologías de información y por los administradores de los sistemas. Cuando se conocen las características propias de los sistemas de archivos se puede tener mayor capacidad de análisis y diagnóstico de la recuperación de los datos, de lo contrario estamos supeditados a lo que exista en los archivos de respaldo, si existen.

Cuando se presente una corrupción de datos en su sistema de archivo revise algunos aspectos propios de la dinámica de los sistemas de archivos:

1. Integridad de la tabla de referencia general de archivos (FAT, MFT, Inodos)
2. Estado de los apuntadores a los datos físicos en el disco
3. Los archivos de metadatos disponibles y activos en el sistema de archivos.
4. Parámetros del sistema operacional en cuanto al uso de memoria física y utilización de bloques no asignados en el tiempo.
5. Versión del sistema de archivos y las limitaciones propias en el almacenamiento de grandes volúmenes de archivos.

Los sistemas de archivo son los depositarios de la esencia misma de las organizaciones del siglo XXI: la información, sin ella o su adecuada custodia, éstas afrontarán grandes retos para mantener una operación coherente y confiable. Por tanto, conocer y analizar los sistemas de archivo si bien es una tarea eminentemente técnica, es preciso valorarla en el contexto organizacional como una práctica estratégica del área de tecnologías de información para proteger y custodiar los registros que dan cuenta de las estrategias y acciones de las organizaciones.

Referencias

- Carrier, B. (2005) File system forensic analysis. Addison Wesley.
Carrier, B. (2004) Digital Forensic Tool Testing Images. <http://dfft.sourceforge.net>
Venema, W. y Farmer, D. (2005) Forensic Discovery. Addison Wesley
Nelson et al (2004) Guide to computer forensics and investigations. Thomson. Course Technology.

Datos del Autor:

Jeimy J. Cano M., Ph.D

Ingeniero de Sistemas y Computación, Universidad de los Andes. Magíster en Ingeniería de Sistemas y Computación, Universidad de los Andes. Doctor of Philosophy (Ph.D) in Business Administration, Newport University. Profesional certificado en Computer Forensic Analysis (CFA) del World Institute for Security Enhancement, USA. Profesional certificado como Certified Fraud Examiner (CFE) por la Association of Certified Fraud Examiners, USA. Conferencista, profesor universitario e investigador independiente en Seguridad Informática a nivel nacional e internacional. Coordinador general de la lista de seguridad SEGURINFO. Contacto: jjcano@yahoo.com