Márcio Fernandes Justino

Identificação de Fragmentos de Arquivos em Processo de File Carving sobre Sistemas de Arquivos NTFS

São Paulo – SP

Maio / 2013

Márcio Fernandes Justino

Identificação de Fragmentos de Arquivos em Processo de File Carving sobre Sistemas de Arquivos NTFS

Pesquisa apresentada como requisito para conclusão do curso de Pós Graduação em Computação Forense da Universidade Presbiteriana Mackenzie de São Paulo.

Orientadora:
Ana Cristina Azevedo

UNIVERSIDADE PRESBITERIANA MACKENZIE
INSTITUTO DE COMPUTAÇÃO
PÓS GRADUAÇÃO EM COMPUTAÇÃO FORENSE

São Paulo – SP

Maio / 2013

Monografia sob o título Identificação de Fragmentos de Arquivos em Processo de File Carving sobre Sistemas de Arquivos NTFS, desenvolvida por Márcio Fernandes Justino, e aprovada em 13 de abril de 2013, São Paulo capital, pela banca constituída por:

Ana Cristina Azevedo
Orientadora

Ivete Irene dos Santos
Orientadora

Dedico a meus pais, cujo exemplo de
honestidade e trabalho tem marcado minha vida,
à minha esposa que tem me apoiado nesta caminhada e
à minha filha que, sempre sorridente, pude ver
crescer desde o início desta pesquisa.

Resumo

A fragmentação de arquivos tem sido o calcanhar de Aquiles dos processos investigativos e perícias forenses envolvendo a recuperação de arquivos e, principalmente, de arquivos não mais alocados, removidos ou corrompidos. A abordagem do tema tem caminhado para um avanço nos processos de localização dos fragmentos de arquivos e seus diferentes tipos de técnicas, possibilitando que as ferramentas evoluam cada vez mais no processo de identificação e localização de arquivos fragmentados, auxiliando assim na investigação mais robusta e eficiente de evidências digitais.

Palavras-chave: Fragmentação, investigação, arquivos, técnicas, evidências.

Abstract

The file fragmentation has been the Achilles heel of investigative processes and skills involving forensic file recovery and mainly unallocated, removed or corrupted files. The theme has moved towards a breakthrough in the localization processes fragments of files and their different types of techniques, enabling the tools evolve increasingly in process identification and location of fragmented files, thus assisting in the investigation of more robust and efficient digital evidence.

Keywords: Fragmentation, investigation, files, techniques, evidence.

Sumário

Lista de Figuras

Lista de Tabelas

1 Introdução

Juntamente com o avanço da tecnologia computacional e da internet, veio o aumento do número de pessoas conectadas trocando informações, seja em nível pessoal ou organizacional. Segundo o Centro de Estudos sobre as Tecnologias da Informação e da Comunicação (cetic.br), o número de usuários domésticos e no trabalho tem aumentado juntamente com o tempo em que os mesmos permanecem conectados à internet. A figura 1.1 mostra a evolução desses números até o presente momento.

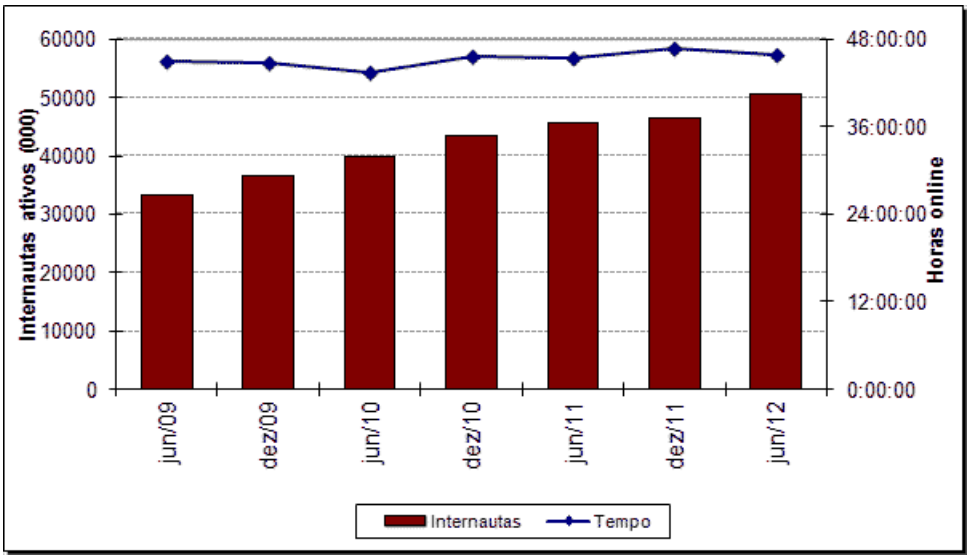

Figura 1.1: Internautas ativos em residências e no trabalho e horas navegadas - 2012 (IBOPE//NETRATINGS, 2012)

Nesse meio, existem usuários que promovem o cibercrime ou atividades ilegais na rede. As informações computacionais são armazenadas em discos rígidos *(HD's)* usando apropriados sistemas de arquivos que são suportados pelo sistema operacional instalado no computador. Existem diversos sistemas de arquivos para armazenamento de arquivos no mercado, e um dos mais comuns atualmente é o NTFS (MAHANT; B.B.MESHRAM, 2012).

Outras pesquisas destacam os sistemas de arquivos no âmbito da investigação forense

> o sistema de arquivos NTFS, assim como outros sistemas de arquivos, não

> foi criado com a computação forense em mente mas sim com a quantidade de informações no computador que pode ser usada em uma investigação (SVENSSON, 2005, p. 32).

Muitos criminosos se utilizam do artifício de excluir ou remover rastros de seus atos criminosos apagando os arquivos criados, manipulados ou alterados, acreditando que com isso seu crime seria perfeito, sem rastros da comprovação de seus atos. A técnica de file carving possibilita a análise de tais arquivos, provendo um avanço investigativo com a possibilidade de extrair provas de arquivos não mais alocados, porém que ainda estejam presentes fisicamente nos discos.

Segundo Caloyannides (2004, p. 26), a operação de remoção de um arquivo não faz absolutamente nada. Esta meramente altera um simples caractere na tabela de alocação do arquivo em questão indicando ao computador que o espaço desse arquivo foi tomado permitindo que os dados do arquivo possam ser sobrescritos no futuro, se necessário. Seguindo raciocínio semelhante, a operação de formatação não remove os dados sensíveis dos arquivos. A formatação faz com que os ponteiros da tabela de alocação que indicam onde os arquivos estão sejam liberados, perdendo assim sua localização pelo sistema de arquivos, mas mantendo os dados intactos em seu sistema. Caloyannides (2004, p. 33) demonstra que os arquivos são alocados no disco em unidades mínimas chamadas clusters. Se o sistema de arquivos não necessita de todo o cluster para armazenar as informações do arquivo, este irá marcar o final do arquivo na porção final de seus dados dentro do cluster, deixando uma porção do cluster com dados considerados como sendo lixo, não sobrescritos pelo sistema. Essa situação gera o que é chamado de *slack space*[1].

Por fim, o processo de formatação e remoção de um arquivo, juntamente com as áreas de slack space do disco, provoca o surgimento de áreas não alocadas, o que caracteriza o processo de file carving.

1.1 Justificativa

Segundo Memon (2011, p. S2), um dos primeiros desafios em file carving pode ser encontrado na tentativa de se recuperar arquivos fragmentados. O processo de file carving é de suma importância para a investigação forense computacional e envolve a identificação de arquivos perdidos, corrompidos ou removidos do equipamento investigado. A dispersão desses arquivos não mais indexados pela tabela de alocação de arquivos do sistema de arquivos NTFS torna o processo de identificação dos arquivos um desafio para a investigação e identificação de ilícitos.

[1] Espaço não alocado em um cluster que pode conter informações de outros arquivos que não foram sobrescritas.

> Durante uma investigação forense digital, muitas peças diferentes de dados são preservadas para investigação, das quais imagens de discos rígidos (HD) são as mais comuns. Essas imagens contêm os dados alocados para arquivos, bem como os dados não alocados. Os dados não alocados ainda podem conter informações relevantes para uma investigação, sob a forma de (partes de) intencionalmente excluídos ou arquivos temporários removidos automaticamente. Infelizmente, esses dados nem sempre são facilmente acessíveis: uma sequência de caracteres da pesquisa sobre os dados brutos pode recuperar (partes de) documentos de texto interessantes, mas ele não vai ajudar para obter a informação presente em, por exemplo, imagens ou arquivos compactados. Além disso, as sequências de caracteres exatas para procurar não podem ser conhecidas antecipadamente. Para obter esta informação, os arquivos apagados precisam ser recuperados (KLOET, 2007, p. 1).

1.2 File Carving

A técnica de file carving é frequentemente utilizada durante investigações digitais e apresenta uma grande importância no processo investigativo visando a recuperação de vestígios e provas digitais. Conforme Memon (2011, p. S2), essa é uma técnica em que arquivos de dados são extraídos de um dispositivo digital sem o auxílio de tabelas de arquivo ou outros meta-dados do disco.

Um estudo de Garfinkel (2007, p.S2) descreve a importância do processo de file carving "na pratica forense, o processo de file carving pode recuperar arquivos que foram removidos e que tiveram sua entrada de diretório realocada para outros arquivos, desde que seus setores de dados ainda não tenham sido sobrescritos".

Deste modo, a intenção da utilização do processo de file carving está na recuperação de informações já perdidas pelo processo de indexação comumente conhecido dos sistemas de arquivos, assunto que será tratado mais detalhadamente nos próximos capítulos.

1.3 Fragmentação de Arquivos

Conforme Menom (2011, p. S2), um dos primeiros desafios do processo de investigação utilizando file carving é justamente a tentativa de recuperar os fragmentos de arquivos não alocados. A fragmentação de arquivos é um desafio para o processo de file carving e é de suma importância para a recuperação de arquivos perdidos em processos de investigação digital. Tendo em vista o presente desafio, tem-se a necessidade de se determinar qual a melhor técnica para identificação de fragmentos de arquivos no processo de file carving.

1.4 Hipótese(s)

Pretende-se com a análise dos metadados de arquivos (cabeçalho e rodapé), determinar um padrão de identificação de pontos de fragmentação de arquivos, visando a identificação, com maior consistência e confiabilidade, de suas partes, reduzindo consequentes falsos positivos apresentados, com certa frequência, durante o processo de file carving, prejudicando a identificação de possíveis evidências comprobatórias. A figura 1.2 demonstra as hipóteses de pesquisa:

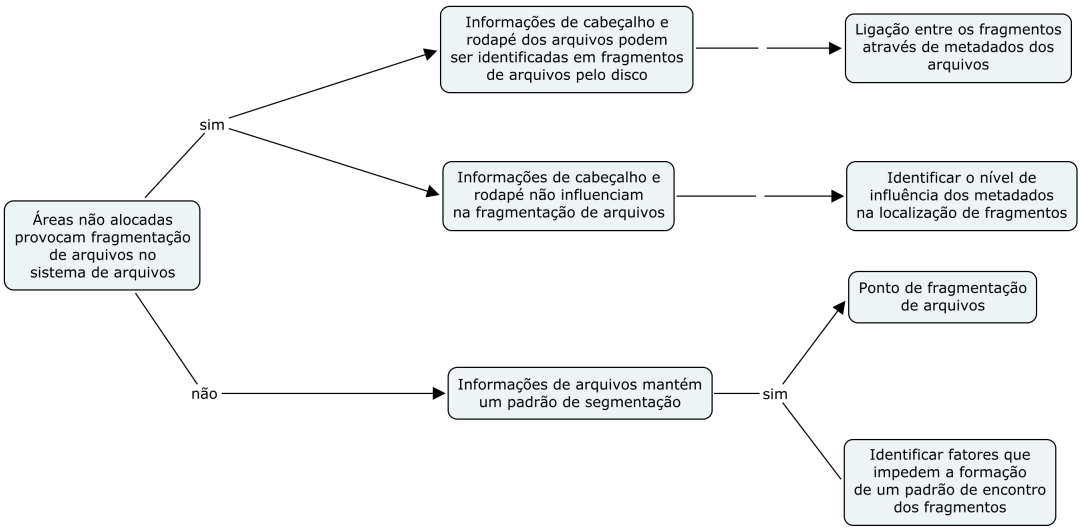

Figura 1.2: Hipótese de pesquisa

1.5 Objetivos

1.5.1 Objetivo Geral

Visto a necessidade de mitigar o impacto da fragmentação no processo de file carving em uma investigação digital, busca-se determinar uma melhor metodologia de localização dos fragmentos de arquivos, esclarecer uma maneira de localização dos fragmentos de um arquivo dentro da área não alocada em sistemas de arquivos NTFS.

1.5.2 Objetivo Específico

Para chegar ao objetivo principal e determinar uma melhor metodologia de localização de fragmentos de arquivos é necessário entender primeiramente e de forma mais detalhada alguns itens específicos:

- Verificar como são identificados os arquivos no sistema de arquivo NTFS.
- Verificar como um arquivo fragmentado é armazenado em um sistema NTFS.
- Levantar uma padronização entre os fragmentos de arquivos para melhor localização.
- Identificar formas de localização de fragmentos dos arquivos não alocados.
- Verificar o uso de entropia para localização de fragmentos de arquivos.

Verificar assim a forma como os arquivos são registrados nos sistemas de arquivos NTFS, o processo de diferenciação de tipos de arquivos para determinar o início e o fim de um arquivo (área de cabeçalho, área de dados, de metadados e ponto de fim de arquivo), podendo então encontrar certos padrões que possam permitir a identificação de partes de um arquivo fragmentado no sistema de arquivos.

O tema abrange já fez parte de diversas pesquisas e desafios internacionais mas, ainda é um desafio presente e não totalmente abordado. Assim, busco contribuir com melhoria significativa na identificação dos fragmentos com base no que já foi descoberto e pesquisado e também introduzir uma iniciativa de pesquisa sobre o uso de entropia para identificação de fragmentos de arquivos analisando seu conteúdo digital.

1.6 Metodologia

Pesquisar e descrever o funcionamento do sistema de arquivos NTFS e identificar em sua estrutura de armazenamento de arquivos o ponto de fragmentação dos arquivos, podendo dessa forma atingir o objetivo e determinar uma melhor forma de recuperação dos fragmentos de arquivos em áreas não alocadas, através das bibliografias levantadas, juntamente com pesquisas já realizadas sobre assuntos correlatos à fragmentação de arquivos e o processo de file carving.

O método de pesquisa utilizado é baseado nas definições de documentação do sistema de arquivos NTFS, junto ao seu criador (Microsoft), pesquisas correlatas referentes à fragmentação de arquivos e sua forma de identificação, pesquisar a ligação entre os fragmentos de dados para entender o ponto de fragmentação e identificar se há um padrão nessa relação de fragmentos de arquivos.

2 *Sistema de Arquivos*

2.1 Discos Rígidos

Os discos rígidos são usados para armazenar informações *não voláteis* como programas, documentos, planilhas, anexos de e-mail, arquivos de sistema e outros criados pelo usuário.

Um disco rígido é composto por uma ou mais partições lógicas, representadas por um volume que pode ser formatado utilizando um sistema de arquivo, seja esse um sistema FAT, NTFS (*Computadores que utilizam Windows*), ext3, ext4 (*Computadores que utilizam Linux*), entre outros sistemas existentes atualmente (SVENSSON, 2005).

Figura 2.1: Disco rígido moderno

2.2 O Sistema NTFS

Projetado pela Microsoft, o NTFS é o sistema de arquivos padrão de vários sistemas operacionais Microsoft desde a versão do Microsoft Windows NT (1993) até as versões mais recentes como o Windows 8 (2012), com algumas evoluções ao longo do tempo. Sendo este o predecessor do antigo sistema FAT, é o sistema mais presente em investigações de sistemas devido a superioridade de utilização de sistemas operacionais Windows em todo o mundo.

O sistema de arquivos NTFS foi projetado para melhor confiabilidade, segurança e suporte de grandes dispositivos de armazenamento de dados. Dispõe do uso de estruturas genéricas que servem de envoltório para estruturas de dados com conteúdo específico. Desta forma, o NTFS se torna um projeto escalável pois a estrutura interna de dados pode mudar inúmeras vezes enquanto que a sua casca (a estrutura genérica) permanece constante. Um bom exemplo desse modelo é que todos os bytes são alocados em arquivos no sistema (CARRIER, 2005).

2.3 Estrutura de Dados

Segundo um dos pesquisadores da estrutura de dados do sistema de arquivos NTFS, Carrier (2005, p. 199), a única estrutura consistente no NTFS está presente nos primeiros setores do disco, contendo os setores de boot e código.

O coração do sistema de arquivos NTFS está na Master File Table (MFT), pois esta contém as informações de todos os diretórios. Todo arquivo e diretório existente possui uma entrada na MFT, sendo que esta é uma estrutura bem simples de 1 KB de tamanho. A Entrada na MFT possui:

- Cabeçalho;
- Atributos; e
- Espaço não utilizado.

A figura 2.2 mostra como é distribuída a estrutura de um disco.

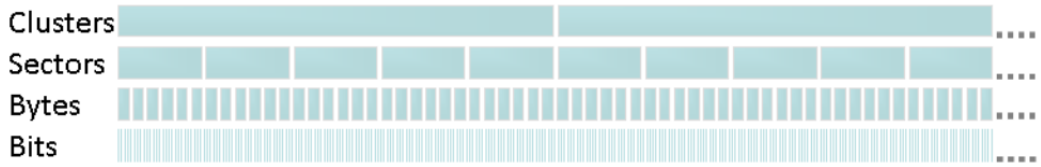

Figura 2.2: Estrutura física de um disco rígido (LTD., 2010).

A figura 2.3 demonstra um registro de entrada na MFT.

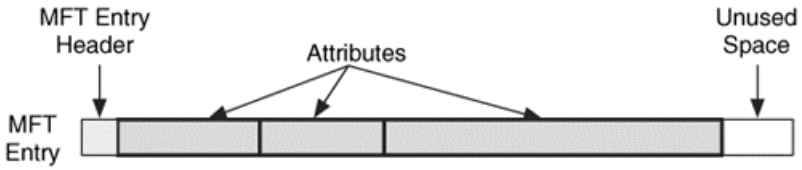

Figura 2.3: Entrada na MFT - Cabeçalho e espaço reservado aos diferentes tipos de atributos. Para o exemplo, a entrada possui 3 atributos.

2.3.1 Cluster

De acordo com Svensson (2005, p. 25), um disco rígido é dividido em setores sendo que o sistema de arquivos se utiliza de setores para formar um cluster. O tamanho do cluster depende da formatação do disco. A tabela 2.1 demonstra tamanhos padrões definidos de cluster para diferentes tamanhos e formas de formatação do disco NTFS:

Tamanho do Disco	**Tamanho do Cluster Padrão**
até 512 MB	512 bytes
513 MB - 1024 MB	1 KB
1025 MB - 2048 MB	2 KB
acima de 2048 MB	4 KB

Tabela 2.1: Tamanho do Cluster Padrão Para Formatos NTFS (quanto maior o tamanho do disco maior o tamanho do cluster).

Conforme descrito por Svensson(2005, p. 17), o disco rígido é dividido em *setores* que tem seu tamanho determinado pelo hardware. Por sua vez, os setores, em conjuntos múltiplos, formam um *cluster*. Os arquivos armazenados no disco são divididos pelos clusters, sendo que o cluster somente pode armazenar um único arquivo. Se um arquivo é maior que o tamanho de um cluster, o sistema utiliza novos clusters para suportar todo o tamanho do arquivo. Se um cluster ocupado ainda dispor de espaço disponível para armazenamento ele não é mais utilizado, deixando um espaço entre o final do arquivo e o final do cluster, o que recebe o nome de *slack space*.

O disco também possui espaços chamados de espaço não alocado (unallocated space) que se trata de regiões não alocadas do disco podendo estas estarem vazias ou conter informações de arquivos já apagados anteriormente e que podem ser de grande valor para a análise forense.

Nos sistemas de arquivos NTFS, todos os clusters possuem um identificador chamado de **Logical Cluster Number** (LCN). O LCN é um número sequencial que representa todos os clusters do disco, do começo do disco até o seu final, iniciando-se em 0 (zero), referindo ao setor de boot do sistema de arquivos. O sistema de arquivo converte o identificador LCN em um endereço físico de disco (posição dos bytes onde o cluster está localizado no disco), multiplicando o identificador LCN com o tamanho do cluster.

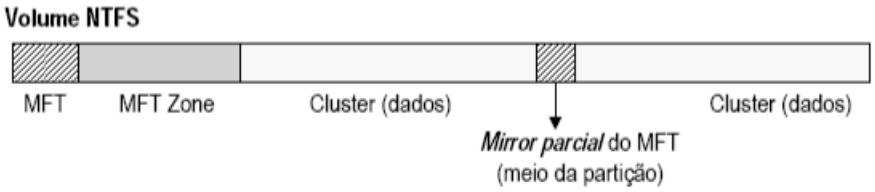

Figura 2.4: Estrutura do NTFS

Os clusters de um mesmo arquivo também recebem um identificador chamado de **Virtual Cluster Number** (VNC). O VNC representa a ordem física dos clusters no arquivo, não sendo necessário, portanto, que a ordem física no disco seja contínua (SVENSSON, 2005).

2.3.2 Master File Table

Segundo Svensson (2005, p. 26), todos os arquivos no sistema NTFS apresentam ao menos uma entrada no registro da *Master File Table* (MFT). Os primeiros 16 registros da MFT são reservados pelo sistema de arquivos para manter metadados[1] específicos do NTFS. A tabela 2.2 detalha os 16 primeiros registros reservados ao NTFS na MFT:

O primeiro registro da tabela MFT é um registro referente à própria MFT em si. Esse registro é seguido por outro arquivo que representa a cópia parcial da MFT (um espelho da própria master file table, **$MFTMIRR**). Esse registro de espelho da MFT funciona como um backup para importantes metadados de arquivos, usado para recuperação da MFT em caso de problemas.

O registro **$BITMAP** que contém todo o mapeamento de clusters que estão em uso (alocados por arquivos). Todo cluster é identificado por um bit no registro **$BITMAP**, sendo que quando o cluster está sendo usado o bit tem seu valor igual a hum. Esse bit também pode ser

[1] Dados que definem ou descrevem outra parte dos dados (WYMAN et al., 2012).

Nome MTF	**Registro**	**Descrição**
$MFT	0	NTFS's Master File Table. Contém um arquivo base para cada arquivo ou diretório do disco.
$MFTMIRR	1	Uma cópia parcial da MFT que serve como backup para casos de falhas de um setor.
$LOGFILE	2	Log de transação de arquivos.
$VOLUME	3	Contém o número serial do volume e data de criação.
$ATTRDEF	4	Definição de atributos.
.	5	Diretório raiz do disco.
$BITMAP	6	Contém um mapeamento binário de todos os clusters do volume (usados e não usados).
$BOOT	7	Registro de boot do drive.
$BADCLUS	8	Lista de setores com problemas no drive.
$SECURE	9	Contém uma única descrição de segurança para todos arquivos do volume.
$UPCASE	10	Mapeia caracteres em texto minúsculo para texto maiúsculo.
$EXTEND	11	Extensões opcionais como as quotas, pontos de reanálise de dados e identificador de objetos.
	12-15	Reservado para uso futuro.

Tabela 2.2: Registros reservados do NTFS na Master File Table.

encontrado em registros de diretórios com um propósito de manter um registro de quais clusters do buffer de alocação de índices estão em uso e quais não estão.

Já o registro **$BADCLUS** é usado para manter uma lista dos clusters que são marcados como defeituosos pelo sistema operacional. Quando o sistema detecta um cluster com defeito ele registra esse cluster no **$BADCLUS** e esse cluster passa a não ser mais utilizado pelo sistema.

A figura 2.5 demonstra a estrutura e formatação da *Master File Table*.

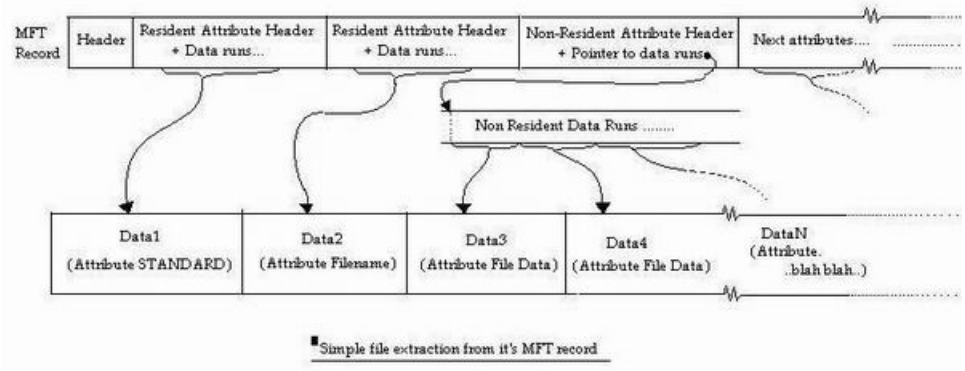

Figura 2.5: Registro MFT - Representação da estrutura da MFT (T., 2013).

2.4 Alocação de Arquivos

2.4.1 Registros da MFT

Os registros da MFT incluem inúmeros atributos e valores, sendo limitado em 1 KB fisicamente. Os atributos de registros são divididos em dois componentes lógicos - cabeçalho e área de dados - sendo que no cabeçalho são armazenados os tipos dos atributos, a localização e o tamanho do valor do atributo (SVENSSON, 2005).

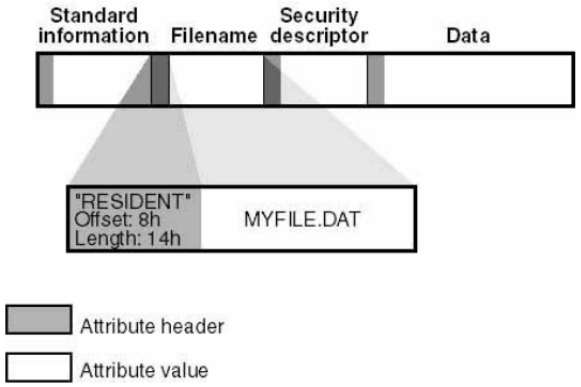

Figura 2.6: Registro MFT - Representação de um atributo e o tamanho de dados.

A tabela 2.3 define os inúmeros atributos do sistema de arquivos NTFS.

2.4.2 Registros de Arquivos

Todo registro de arquivo consiste de ao menos um par de atributo/valor. Se toda a informação residente no par atributo/valor de um registro de arquivo couber no registro da MFT, nenhum dado adicional é associado no volume, ficando a informação alocada somente no registro da MFT. Por outro lado, um arquivo com uma informação maior, que não cabe somente no registro da MFT, teria seus dados armazenados em clusters em outro lugar do volume de dados. Sendo assim, todo arquivo pode estar associado a múltiplos clusters de dados e a informação de localização dos mesmos é listada na área de dados do registro da MFT.

Um cluster de dados de um arquivo altamente fragmentado também podem ser muito extenso para ser armazenado nos atributos de um registro de arquivo, assim a lista de atributos do registro de arquivo é usado para direcionar a outro ponto de cabeçalho de dados na MFT que por sua vez direciona para os clusters de dados com as informações restantes referentes ao arquivo (SVENSSON, 2005).

A figura 2.7 demonstra essa fragmentação de informações do arquivo.

Tipo do Atributo	Descrição
$VOLUME_INFORMATION	Versão do volume de dados.
$VOLUME_NAME	Nome do volume de dados.
$FILE_NAME	Nome do arquivo ou do diretório. Um arquivo pode ter múltiplos atributos de $FILE_NAME.
$STANDARD_INFORMATION	Atributos de arquivo (somente leitura, data de criação e de modificação, etc).
$SECURITY_DESCRIPTOR	Presente para manter compatibilidade com outras versões do sistema de arquivos. Armazenava informações de segurança dos arquivos e diretórios.
$DATA	Dados do arquivo.
$INDEX_ROOT	Implementa a alocação do nome do arquivo.
$INDEX_ALLOCATION	Implementa a alocação do nome do arquivo.
$BITMAP	Índices para grandes diretórios (somente para diretórios).
$OBJECT_ID	Usado pelo serviço de rastreamento de link distribuído.
$REPARSE_POINT	Armazena um ponto de reanálise de dados.
$ATTRIBUTE_LIST	Lista o local de todos os registros de atributos da MFT que não couberam no registro da MFT.
$EA_INFORMATION, $EA	Atributos estendidos de compatibilidade.
$LOGGED_UTILITY_STREAM	A **EFS** (Encrypting File System) armazena informações que são usadas na manipulação de arquivos encriptados.

Tabela 2.3: Registros reservados do NTFS na Master File Table.

2.4.3 Registros de Diretório

Os registros de diretórios possuem um atributo de indexação raiz que possui o ponto de entrada de cada arquivo e subdiretório de sua estrutura.

A figura 2.8 mostra essa indexação.

As entradas de índice consistem do nome do arquivo e da cópia das informações padrões da entrada de diretório. Como o número de arquivos em diretórios pode crescer, o espaço pode não ser suficiente para armazenar todas as entradas de índice então, os índices adicionais são armazenados na área de alocação conhecida como *"Index Allocation Buffer"*. A área de $BITMAP também é usada para indicar quais VNC's presentes no "Index Allocation Buffer" estão sendo usados.

A figura 2.9 demonstra o funcionamento desse processo.

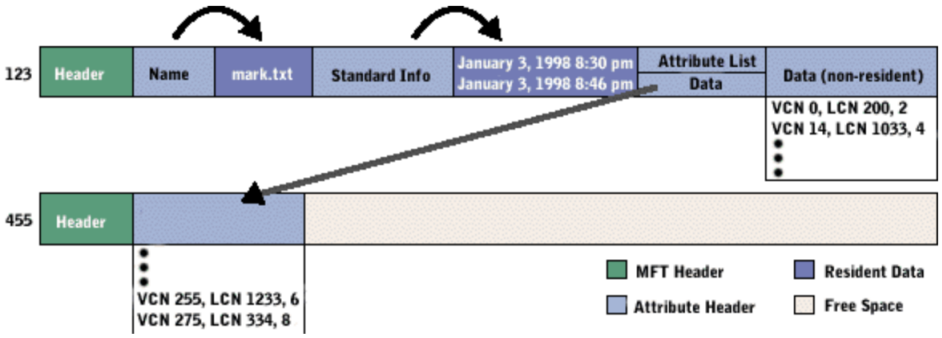

Figura 2.7: Registro de arquivo da MFT (SVENSSON, 2005).

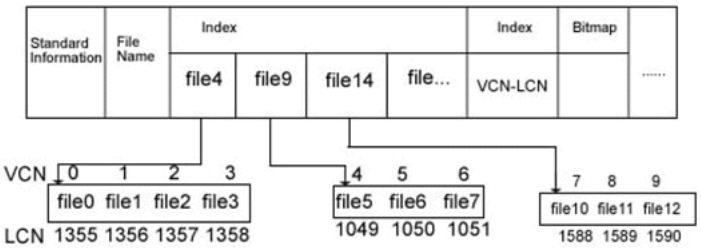

Figura 2.8: Registro de diretório .

2.4.4 Remoção de Arquivos

Quando um novo arquivo é criado em um sistema de arquivos NTFS, o arquivo **$BITMAP** do registro da MFT precisa ser atualizado pois os clusters anteriormente considerados *não alocados* agora estão ocupados por um arquivo. Assim, quando um arquivo é excluído o arquivo **$BITMAP** do registro da MFT também precisa ser atualizado. Os bits correspondentes ao cluster anteriormente alocado pelo arquivo removido são zerados (marcados como *não alocados*). O arquivo foi então marcado para remoção porém seu conteúdo continua no disco no mesmo

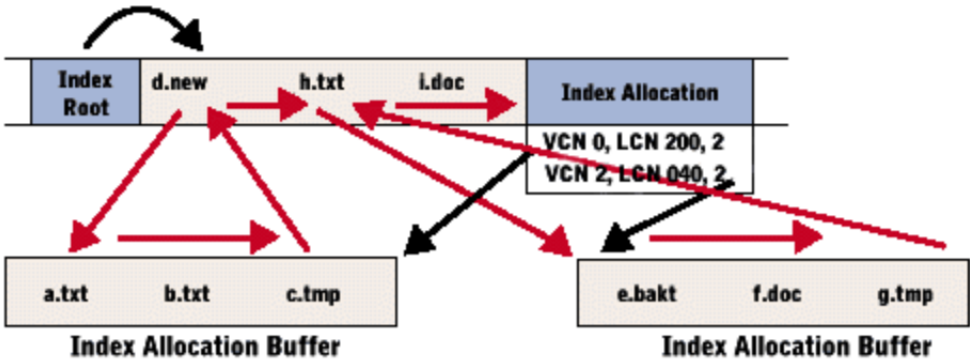

Figura 2.9: Registro de diretório e a “Index Allocation Buffer”.

lugar que estava anteriormente alocado até que o mesmo local seja utilizado para armazenar outro arquivo. A entrada do índice do arquivo é também marcada para exclusão porém, a entrada do índice é rapidamente sobrescrita, o que no caso do arquivo pode demorar dependendo do tamanho do volume de dados (SVENSSON, 2005).

2.5 Análise NTFS

Svensson (2005, p.33) diz que registro da MFT pertencentes à arquivos já apagados podem ser recuperados pois, como vimos anteriormente, os arquivos não são excluídos do disco e sim deixam de ser referenciados, permanecendo no mesmo lugar até que outro arquivo sobrescreva seu conteúdo. As chances de se recuperar um arquivo excluído, no entanto, diminuem com o tempo uma vez que o sistema NTFS sobrescreve arquivos não mais alocados antes de alocar espaço adicional para a MFT. Informações padrões de arquivo, assim como seu nome, estão contidas nesses registros da MFT. Isso pode ser útil para auxiliar a encontrar as partes do arquivo não mais alocado uma vez que as informações de apontamento para as partes do arquivo não mais alocadas e ainda residentes no disco, estão presentes no registro de indexação do arquivo encontrado. Sem esse registro do arquivo ainda seria possível se recuperar um arquivo deletado fazendo uma procura física no disco desde que este arquivo não esteja fragmentado. Arquivos fragmentados são muito difíceis de se recuperar totalmente através de pesquisas físicas no disco.

Tendo isso em vista, Svensson relata que o índice pertencente à um arquivo removido pode não mais ser localizado visto que o mesmo pode ser rapidamente sobrescrito, o que já foi visto anteriormente. Essa entrada de arquivo (índice) pode ser muito importante na reconstituição do crime. Quando uma entrada de índice é marcada para remoção, todas as entradas seguintes são movidas para cima sobrescrevendo, portanto, o índice marcado para remoção. Se essa entrada de índice é a última entrada em um registro de diretório ou no buffer de alocação de índices ela pode então não ser sobrescrita podendo assim ser recuperada.

3 *File Carving*

File Carving é o processo de recuperação de fragmentos de arquivos não mais indexados baseado no seu conteúdo e na ausência de metadados do sistema de arquivos (HAND et al., 2012).

O processo de file carving é útil na recuperação de arquivos e amplamente utilizado em investigações digitais (*em forense computacional*). Devido a isso, esse é um dos processos mais importantes e desafiadores da computação forense (GARFINKEL, 2007).

O termo "Carving", definido por Cohen (2007, p. 1), é comumente utilizado para indicar a recuperação de arquivos em imagens de dados desestruturados, imagens que não contem as informações úteis do sistema de arquivos, informações estas que são utilizadas para auxiliar na recuperação do arquivo. Muitos analistas forenses se utilizam de técnicas de escultura como um último recurso devido à dificuldade das técnicas atuais. Diversas técnicas atuais utilizam técnicas de inspeção manual para recuperar arquivos, utilizando-se da tentativa e erro. Atualmente essa técnica tem se tornado impraticável devido ao grande volume de dados dos dispositivos de armazenamento de informações.

3.1 Desafio

Um dos primeiros desafios do processo de file carving, segundo Memon (2008, p. S2), se encontra na tentativa de se recuperar arquivos fragmentados. Um ponto chave no processo de recuperação desses arquivos fragmentados é encontrar a relação de fragmentação do arquivo que pode beneficiar o processo de recuperação de dados. Técnicas tradicionais não conseguem recuperar arquivos quando o sistema de arquivos é ou está corrompido, quando a tabela de alocação não está presente ou possui endereçamentos errados ou incompletos. Assim, a técnica de File Carving apresenta sua capacidade de recuperar arquivos em espaços não alocados do disco (*área do disco não apontada pela tabela de partição - no caso do NTFS, a Master File Table*).

3.2 Assinatura de Arquivo

As ferramentas de file carving mais comuns analisam as informações de assinatura do arquivo, através do cabeçalho e do rodapé do arquivo, determinando assim o conteúdo do arquivo entre esses blocos. Infelizmente, ferramentas poderosas de file carving atuais ainda falham na recuperação de arquivos fragmentados pois somente verificam cabeçalhos conhecidos previamente e muitos arquivos podem apresentar o conteúdo que não condiz com sua assinatura ou arquivos dentro de outros arquivos, que seriam falsamente identificados por uma ferramenta que somente identifica arquivos através do cabeçalho.

A assinatura de alguns tipos de arquivos mais comuns pode ser vista na tabela 3.1.

Extensão	Assinatura	Descrição
EXE	4D 5A	Arquivo executável (Windows / DOS)
E01	4C 56 46 09 0D 0A FF 00	Arquivo lógico de evidência
JPEG	FF D8 FF E0	Arquivo de imagem compacto
JPG	FF D8 FF E0	Arquivo de imagem compacto
PDF	25 50 44 46	Arquivo (Acrobat Reader)
PNG	89 50 4E 47 0D 0A 1A 0A	Arquivo de imagem compacto com transparência

Tabela 3.1: Exemplos de assinatura de arquivos pelo cabeçalho (DATABASE, 2013).

Um exemplo de arquivo PDF e um PNG abertos pelo WinHex[1] mostrando exatamente a assinatura do arquivo PDF contido no cabeçalho (Offset = 00000000).

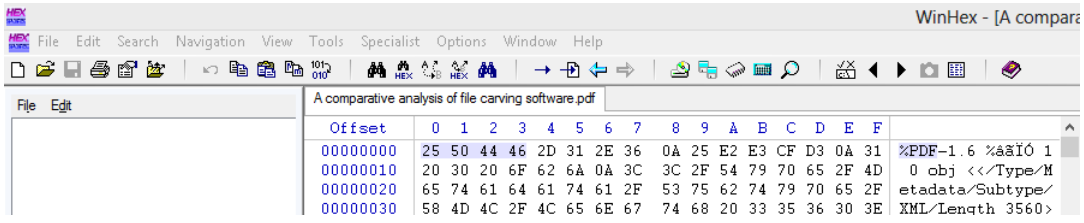

Figura 3.1: WinHex exibindo a assinatura de um arquivo PDF.

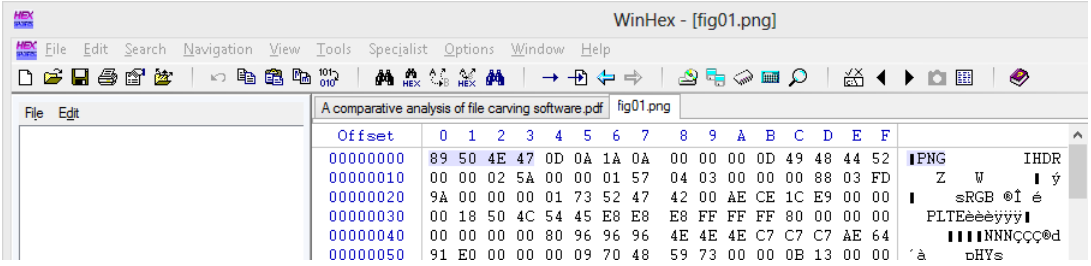

Figura 3.2: WinHex exibindo a assinatura de um arquivo PNG.

[1]Editor hexadecimal particularmente muito útil na recuperação de dados e verificação de arquivos em computação forense.

3.3 Fragmentação de Arquivos

Segundo Kloet (2007, p. 8), um arquivo fragmentado é um arquivo dividido em várias partes onde cada parte pode estar localizada em um lugar diferente em um mesmo conjunto de dados. Sistemas operacionais modernos, tais como o NTFS, tentam gravar os dados evitando a fragmentação porém, ainda existem algumas situações em que a fragmentação ocorre. Memon (2008, p.S3) define que um arquivo é dito fragmentado quando o mesmo está armazenado de forma descontinuada nos clusters, e que o maior desafio para o processo de data carving é justamente a recuperação de arquivos quando esses estão fragmentados em duas ou mais partes.

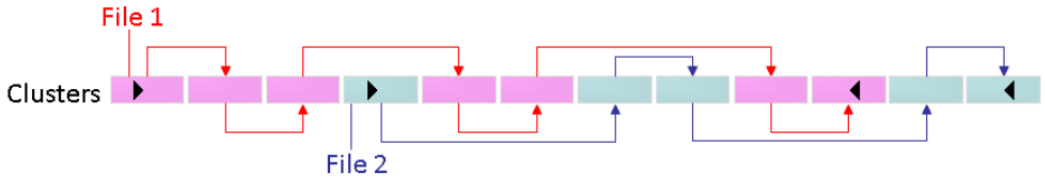

Figura 3.3: Demonstração de fragmentação de arquivos nos clusters do disco rígido (LTD., 2010).

Garfinkel (2007, p. S4) defendia a hipótese de que diferentes tipos de arquivos poderiam apresentar diferentes padrões de fragmentação, o que poderia determinar uma fragmentação diferenciada para um tipo de arquivo específico no disco. Arquivos comumente de sistemas, instalados juntamente com a parte do sistema operacional apresentariam um baixo fator de fragmentação enquanto arquivos comuns e altamente importantes na análise forense, por se tratarem de informações do dia a dia tais como documentos (.doc), mídias (.avi, .jpeg), arquivos de correio eletrônico armazenados (.pst), planilhas com cálculos e fórmulas (.xls), arquivos de log e arquivos texto (.txt), aparentam ter uma tendência a maior fragmentação se comparados à arquivos menos significativos para o processo forense. Garfinkel (2007, p. S2) então chegou a uma de suas principais conclusões, de que até 42% de arquivos de e-mail (.pst), 17% de arquivos microsoft word (.doc) e 16% de arquivos de imagem (.jpeg) são fragmentados, deixando bem claro que a recuperação de arquivos fragmentados é um problema crítico para a análise forense. Outro fator importante que tem influência na fragmentação, que pode ser percebido por Garfinkel (2007, p. S6), é que em dispositivos com maior capacidade de armazenamento, o fator de fragmentação é menor se comparado a dispositivos de menor capacidade de armazenamento.

Alguns fatores que motivam a fragmentação de arquivos são citados:

- Quando não há mais espaço de mídia suficiente na sequência física para a gravação de um arquivo, assim, para ser alocado no dispositivo ele tem que ser dividido em dois ou mais fragmentos.

- Na sequência de um arquivo já alocado, o espaço restante no cluster não é suficiente para a gravação do arquivo de forma sequencial, sendo necessária a divisão do arquivo em dois ou mais fragmentos.

3.3.1 Fragmentação Linear

Kloet (2007, p.S4) referencia a fragmentação linear a arquivos que estão fragmentados, seguindo a mesma sequência física do dispositivo.

A figura 3.4 demonstra a fragmentação linear na ordem original e física do dispositivo.

Figura 3.4: Exemplo de Fragmentação Linear

A fragmentação linear é um dos pilares para desenvolvimento de algoritmos de identificação de fragmentos devido ao problema da fragmentação ser muito complexo, necessitando de uma identificação passo a passo, assim esse é o ponto inicial de análise, a forma mais simples de iniciar a identificação do problema.

3.3.2 Fragmentação Não Linear

Conforme Kloet, a fragmentação não linear representa os arquivos fragmentados fora da ordem normal de sequência do disco. A figura 3.5 demonstra como os fragmentos do arquivo *F1* estão fora da sequência natural do disco.

Figura 3.5: Exemplo de Fragmentação Não Linear

3.3.3 Arquivo Parcial

A figura 3.5 demonstra não somente a fragmentação não linear mas também a sobreposição de parte do arquivo. Isso demonstra um arquivo não mais alocado que fora parcialmente sobrescrito. Um arquivo removido pode nunca mais ser totalmente recuperado porém, parte da

informação útil do arquivo ainda pode estar presente na área antes alocada para o mesmo. Kloet diz ainda que para o processo de carving não há diferença entre uma informação parcial e um arquivo fragmentado que ainda não tenha sido totalmente recuperado. Um primeiro detalhe sobre a afirmação de Kloet é que, um dos arquivos pode ser totalmente recuperado dependendo do tempo e da profundidade de busca da ferramenta de carving utilizada (em algum momento conseguiria encontrar as demais partes do arquivo), enquanto um arquivo parcial (sobrescrito) não possui mais suas partes perdidas (por terem sido sobrescritas) no dispositivo e que a ferramenta de carving em algum momento terá que definir se tratar de um arquivo parcial.

3.3.4 Recuperação de Arquivos Fragmentados

Para recuperar arquivos fragmentados, qualquer ferramenta de file carving precisa ser capaz de identificar o ponto de início do arquivo e os blocos necessários para reconstruir o arquivo. Segundo Memon (2008, p. S4), há basicamente três pontos a se seguir no processo de carving:

1. Identificar o ponto inicial do arquivo.
2. Identificar os blocos pertencentes ao arquivo.
3. Organizar os blocos de forma correta para que se possa reconstruir o arquivo.

Garfinkel (2007, p. S10) havia dito que as ferramentas de carving implementam uma variedade de otimizações para validação de objetos. Essas otimizações por sua vez, dependentes de cada propriedade do validador.

3.4 Aspectos de Carving

3.4.1 Carving - Cabeçalho e Rodapé

Atualmente a simples comparação byte por byte é uma operação muito rápida que computadores atuais podem processar com facilidade. Sendo assim, a verificação de cabeçalhos e rodapés estáticos de tipos de arquivos se torna um dos primeiros passos que um algoritmo de recuperação pode seguir. Com base nas informações contidas no cabeçalho e rodapé de um arquivo pode-se determinar o tipo do arquivo em questão e até mesmo, em alguns casos, informações mais específicas referentes ao arquivo (metadados) (GARFINKEL, 2007).

Como já visto no item 3.1, a análise de cabeçalho e rodapé é a técnica mais básica de carving. A técnica consiste em procurar no conjunto de dados por padrões que definem o

início do bloco de um arquivo, como por exemplo o cabeçalho de um arquivo executável (.exe) demonstrado pela figura 3.6 (KLOET, 2007).

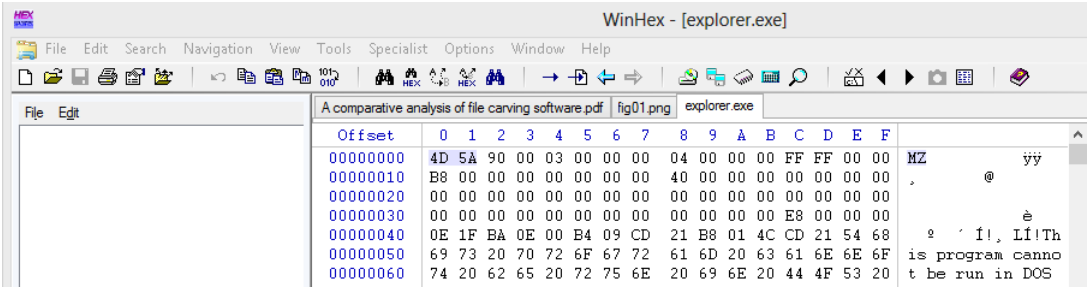

Figura 3.6: Estruda de cabeçalho de um arquivo do tipo EXE

O valor hexadecimal *"4D 5A"* representa os bytes do cabeçalho do arquivo EXE que o identificam como sendo um arquivo EXE. Cada arquivo EXE é marcado a partir desse cabeçalho. A marca final do arquivo ou o rodapé do arquivo delimita o final do arquivo. O conteúdo entre o cabeçalho do arquivo e o final do arquivo representa informação do arquivo como pode ser visto pela figura 3.7.

Figura 3.7: Área de dados de um arquivo do tipo PNG (KLOET, 2007)

Segundo Kloet (2007, p. S9), a análise de cabeçalho e rodapé é um método que apresenta alguns problemas relevantes:

- Muitos falsos positivos.
- Incapacidade de detectar arquivos fragmentados e arquivos parciais.
- Alguns arquivos não apresentam cabeçalho fixo, assim como alguns arquivos também apresentam rodapé variável, o que impossibilita sua identificação por esse método.

O uso desse tipo de análise (cabeçalho e rodapé) por si só não é suficiente para identificar com exatidão os fragmentos de arquivos encontrados no processo, uma vez que muito do conteúdo do arquivo é ignorado por esta técnica. Setores adicionados, removidos ou modificados no espaço de dados situado entre o cabeçalho e o rodapé de um arquivo nunca serão examinados, assim, o uso desse tipo de análise deve ser utilizado somente para rejeitar um objeto de dados, sendo necessária a utilização exaustiva de outros processos (algoritmos) no processamento dos fragmentos que são aprovados por esta análise (GARFINKEL, 2007, p. S7).

3.4.2 Carving - Cabeçalho / Máximo Bloco de Dados

Se utiliza da interpretação do tamanho máximo possível do bloco de dados após a identificação do cabeçalho do arquivo. Esta se torna possível e funcional uma vez que alguns tipos de arquivos não sofrem influência na interpretação dos seus dados se alguma outra informação estiver presente além do final dos dados válidos do arquivo, tais como arquivos de imagem (JPEG) e arquivos de áudio (MP3).

O mesmo problema se repete com método de cabeçalho com tamanho máximo possível de dados, técnica que visa recuperar os dados pelo comprimento máximo disposto para o tipo de arquivo para sua área de dados. Além desses os problemas da técnica de cabeçalho e rodapé, esta técnica ainda tem outros dois problemas ligados à área de dados. Tendo em vista que a área máxima não é exatamente a área máxima estipulada pela definição do tipo de arquivo e sim uma questão de referência, esses dados podem ser primeiramente coletados além do tamanho real do arquivo no dispositivo de origem como também poderá ser menor que o próprio tamanho dos dados do arquivo no dispositivo de origem (GARFINKEL, 2007, p. S10).

3.4.3 Carving - Estrutura de Arquivo

A metodologia de carving em estrutura de arquivo se baseia no uso do layout interno do arquivo (área de dados) para identificar se uma determinada informação pertence ou não a um determinado arquivo. As informações do layout do arquivo normalmente são derivadas de especificações dos formatos de arquivos, responsáveis pela criação daquele tipo de arquivo

específico. Esta técnica é utilizada, em alguns casos, para a identificação de arquivos corrompidos ou fragmentados, se a estrutura de dados é bem detalhada e extensa o suficiente para seu entendimento. Se esta técnica se depara com um arquivo fragmentado que não pode ser recuperado, ela se utiliza de outras maneiras para lidar com esse tipo de informação (KLOET, 2007).

- A informação pode ser descartada, levando a não compreensão de uma informação que na realidade pode fazer parte do arquivo, gerando assim um *falso negativo*.
- A informação pode ser considerada, mesmo não sendo parte da informação de um determinado arquivo, levando a interpretação de um *falso positivo*. O processo ainda pode entender que essa informação foi "considerada" e marcá-la como um *falso positivo conhecido*.

Kloet demonstra então que com uso dessa técnica por si só não deve ser utilizada para recuperar todos os fragmentos ou partes de um arquivo. Isso porque quando um arquivo é fragmentado ou parcialmente substituído em uma posição em que a estrutura de dados restante não está mais presente ou pouca parte dela está presente, a técnica não é capaz de determinar com precisão a informação do fragmento ou da parte de um arquivo. O tipo de arquivo de imagem PNG é um bom exemplo dessa ocorrência, a área de dados da estrutura do arquivo PNG possui uma pequena estrutura, gerando o problema da identificação de fragmentos ou partes desse tipo de arquivo.

Para capacitar a detecção de fragmentação, outra técnica surge buscando mitigar os problemas da técnica de carving em estrutura de arquivo. A técnica de carving baseada no conteúdo do *bloco de dados* se baseia no princípio utilizado pelos discos rígidos para armazenar os dados em setores, onde a fragmentação somente ocorre sobre os limites setoriais. Assim, essa abordagem se utiliza desse conhecimento para checar cada bloco de dado e verificar se o mesmo faz parte do arquivo. Como descreve Kloet, uma forma básica de determinar o tipo de caractere presente no bloco é identificar o tipo de informação contida no mesmo, como exemplo, um bloco que possui uma parte dos dados caracteres texto e outra parte de dados sendo zeros, levando a conclusão que o bloco de dados pertence à um tipo de texto unicode.

As imagens 3.8 e 3.9 demonstram a abordagem dessa técnica.

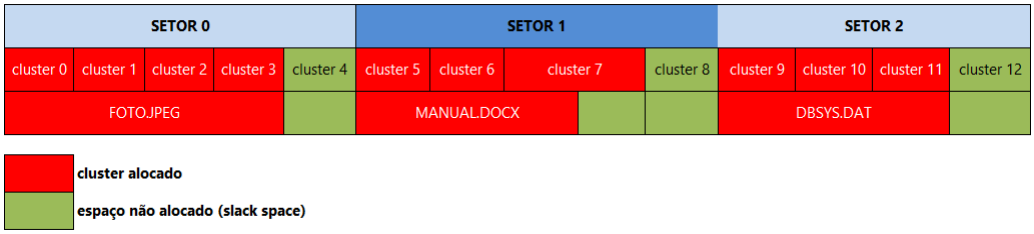

Figura 3.8: Arquivos não fragmentados distribuídos nos clusters.

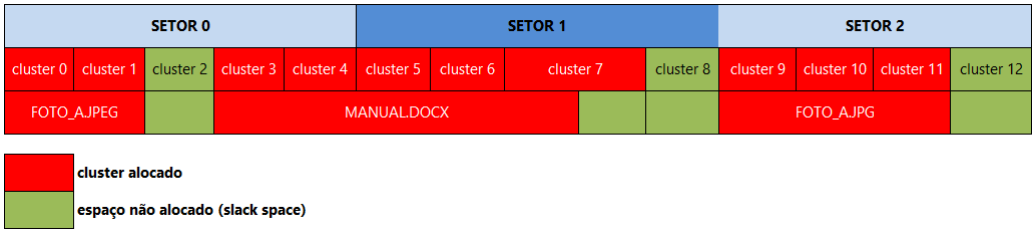

Figura 3.9: Arquivos fragmentados distribuídos nos clusters.

4 *File Carving Avançado*

Com os sistemas de arquivos mais modernos e a dificuldade de se utilizar técnicas tradicionais para recuperar arquivos, faz-se necessária a utilização de técnicas mais avançadas e automáticas para possibilitar a operação de carving em volumes grandes de dados, mesmo que os dados tenham informações no sistema de arquivos. Nesse contexto, diversas ferramentas de file carving foram criadas para automatizar o processo de recuperação (COHEN, 2007).

4.1 Fragmentação

4.1.1 Classificação de Fragmentos de Arquivos

Outra necessidade não menos importante no processo de forense de arquivos fragmentados é a classificação dos fragmentos. A pesquisa pelas partes de um determinado arquivo podem levar a buscas infindáveis se o arquivo for muito extenso, faz-se necessário qualificar os fragmentos do arquivo de forma a identificar que um fragmento possa ou não fazer parte do arquivo em questão.

O fragmento de um arquivo do tipo TXT, servindo de exemplo, não poderia fazer parte de área de metadados de um arquivo de imagem do tipo JPEG (comprimido) pois a estrutura dos dados de um tipo de arquivo se diferem muito (FITZGERALD et al., 2012, p. S44).

4.1.2 Aprendizagem de Máquina

Alguns métodos de inteligência artificial, aplicados ao aprendizado computacional (*Machine Learning*), tem sido explorados para tratar o problema da classificação dos fragmentos de arquivos. Alguns desses métodos, explorados recentemente, são os *Support Vector Machines* (SVM's) - ferramentas muito úteis no aprendizado supervisionado. Segundo Li et al. (2010), uma das maiores vantagens das SVM's é sua possibilidade de explorar padrões em espaços de elevadas dimensões, o que com métodos convencionais estatísticos se torna muito difícil ou

insuficiente para ser aplicado. Seu uso se baseia no tratamento vetorial do histograma de fragmentos de dados, uma vez que o histograma tem custo computacional baixo, tornando o uso das SVM's um método extremamente rápido e eficiente.

4.1.3 Ponto de Fragmentação

Conforme Menon (2008, p. S6), os arquivos fragmentados em mais de duas partes são extremamente raros porém, se um arquivo é fragmentado em mais de duas partes irão existir múltiplos pontos de fragmentação para o arquivo. A identificação do primeiro ponto de fragmentação não difere da localização dos demais fragmentos para arquivos com mais de duas partes de fragmentação.

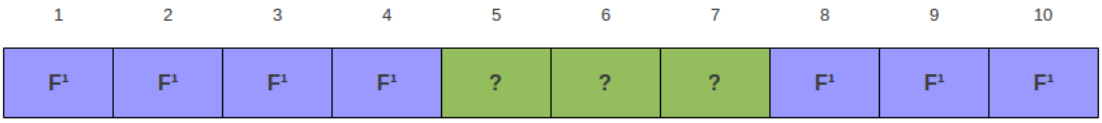

Figura 4.1: Fragmentos do arquivo *F* fictício.

A informação contida entre o final do fragmento e o início de outro fragmento pertencente ao arquivo, contém informações não pertencentes ao arquivo, o que pode ser visto na figura 4.1 entre o bloco **4** (final do primeiro fragmento) e o bloco **8** (início do próximo fragmento do arquivo *F*).

Sendo assim, Menon se utiliza de uma abordagem de métodos para identificar se um bloco contém informações pertencentes ao arquivo ou não, e assim, decidir se o fragmento realmente faz parte do arquivo analisado.

Um dos métodos abordado é o de uso de palavras-chave e assinatura de arquivo para determinar se a informação contida no bloco tem relação com o mesmo tipo do arquivo analisado, havendo divergência nesse ponto, determina-se então um ponto de fragmentação do arquivo. Se um arquivo analisado do tipo executável (.exe) contém em um de seus blocos o cabeçalho de um arquivo de imagem (.jpeg) pode-se então considerar um ponto de entrada de fragmentação nesse bloco. De forma similar, o uso de palavras-chave pode ser usado para identificar entradas inválidas e não suportadas para determinados tipos de arquivos, o que também se pode considerar como um ponto de entrada de fragmentação.

A análise de conteúdo do bloco também é um outro método eficiente na localização de fragmentos de arquivos sendo que uma mudança drástica nas características dos dados de um arquivo pode indicar uma mudança estrutural, em consequência determinar que este fragmento não faz parte do arquivo em análise.

Pode-se dizer que a essência do processo de file carving é a extração de todos os bytes que pertencem ao arquivo dada uma imagem (evidência) investigada. Esse processo de extração é possível pela função de mapeamento, que mapeia os bytes da imagem nos bytes do arquivo (COHEN, 2007).

Cohen (2007, p. 2) demonstra um exemplo de função de mapeamento através da figura 4.2:

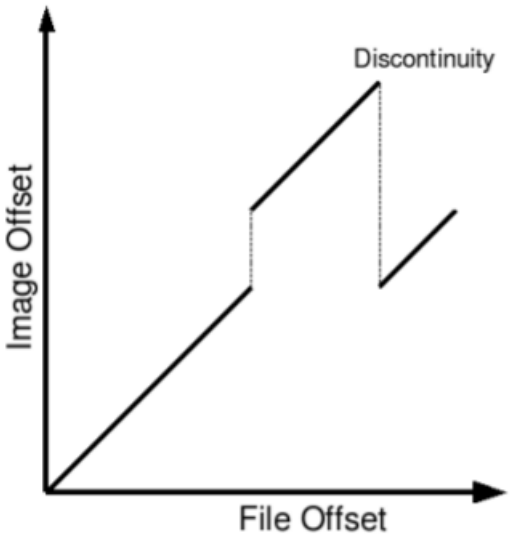

Figura 4.2: Função de mapeamento de processo de file carving.

Uma das propriedades interessantes da função de mapeamento é a capacidade de demonstrar as descontinuidades (possíveis fragmentações) ao longo dos bytes da imagem extraída. Na imagem 4.2 nota-se que há uma descontinuidade no conjunto de bytes relativos da imagem que está sendo extraída em um determinado momento. A interseção desse ponto de descontinuidade na função de mapeamento é conhecida como ponto de fragmentação. Um ponto de fragmentação presente em planos diferentes pode representar uma sequência de arquivos descontinuados, demonstrado pelos pontos **P1** e **P2** na figura 4.3.

Inicialmente essa teoria pode levar ao entendimento de que possam existir inúmeras descontinuidades entre os trechos de dados (pontos P1 e P2) mas na realidade, em sistemas operacionais modernos, os índices de fragmentação se mantém relativamente baixos. Como resultado, Cohen descreve que quanto menor for a distância entre os pontos P1 e P2 (região descontinua do arquivo) maior a probabilidade da fragmentação se enquadrar no nível 1.

4.1.4 Análise de Entropia

Com base na ideia de um dos métodos do ponto de fragmentação, o de análise de conteúdo, surge então a necessidade de aplicação de uma metodologia capaz de analisar os blocos de

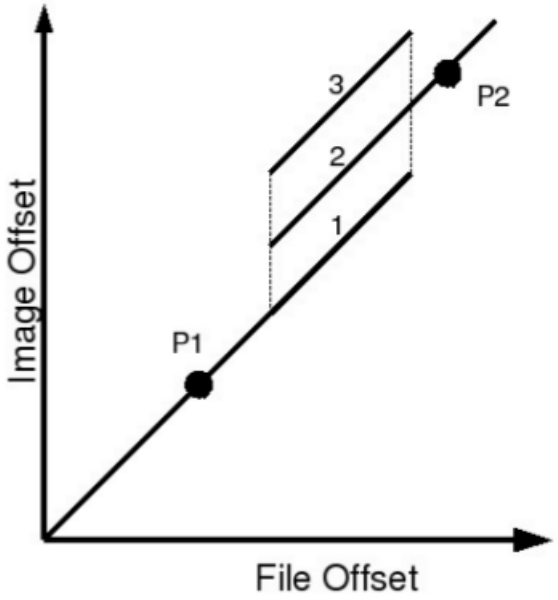

Figura 4.3: Descontinuidade múltipla, indicação de possível fragmentação de múltiplos arquivos.

arquivos em busca de divergências e características dos dados contidos nos mesmos.

A análise de conteúdo por entropia tem sido utilizada por pesquisadores e estudiosos da área forense com o objetivo de identificar conteúdo oculto em arquivos, atribuído a uma técnica anti-forense comumente usada, conhecida como *Esteganografia*. Quando aplicada na identificação de conteúdo oculto ou modificado, tem apresentado excelentes resultados por determinar alterações na estrutura digital e nas características dos dados.

Conceito originado da teoria da termodinâmica, a entropia está presente também no mundo da informação digital. Enquanto na termodinâmica se torna impossível determinar os estados possíveis da matéria, utiliza-se então uma aproximação, na entropia da informação digital, por se tratar de arquivos digitais, a probabilidade de cada estado é conhecida com precisão, já que se tem conhecimento exato do conteúdo do arquivo. Na análise de informações a preocupação recai sobre os bytes de dados, representado por até 256 valores possíveis. Podemos então pensar em entropia como o grau em que o bloco de dados encontra-se desordenado (KLOET, 2007, p. 13).

Em 2006, a abordagem estatística da entropia sobre conteúdo de arquivo foi a campeã no desafio de técnicas de file carving. Uma das técnicas usadas nessa abordagem foi a utilização do cálculo de entropia do bloco de dados.

Kloet cita um exemplo para exemplificar o uso da entropia. Tomando como exemplo um arquivo PNG com grande área de dados, vários blocos seriam vários blocos de dados que uma

técnica de file carving baseada em estrutura não seria capaz de identificar possíveis fragmentos, porém, esses blocos apresentam praticamente o mesmo tipo de dados compactados (comum em conteúdo de um arquivo de imagem), o que significa que a entropia desses blocos deve permanecer estável entre eles. Quando ocorre uma mudança muito súbita da entropia nesses blocos, surge então um forte indício de que isso possa se tratar de um ponto de fragmentação. Isso faz a técnica de entropia, em conjunto com técnicas de inspeção manual, principalmente em tipo de arquivos que apresentam uma pequena área de estrutura de dados e blocos de dados mais comprimidos, como o arquivo PNG citado. O principal problema da técnica de entropia está no cálculo de blocos que pertencem ao arquivo de blocos que não pertencem ao arquivo, causando falsos positivos se utilizada com nesse cenário.

Muito embora a entropia, juntamente com a abordagem estatística dos dados funcione para tipos de arquivos que apresentam um padrão significativamente definido, esta não funciona tão bem quando os padrões de dados não são tão óbvios (LI et al., 2010, p. 237).

5 *Considerações Finais*

Com base em todos os métodos apresentados, inclusive a abordagem da análise de entropia na recuperação e classificação de fragmentos de dados, a conclusão que se chega é que a recuperação de dados fragmentados é e continua sendo um desafio para as investigações forenses, pela dificuldade de se aplicar as técnicas, pela diferenciação entre técnicas e tipos de aplicação para tipos de arquivos diferenciados e pelo crescente volume de informações presentes nas imagens investigadas. Tendo em vista sua complexidade e a variedade de situações em que os fragmentos podem se apresentar, e a impossibilidade, até o presente momento, da aplicação de automação do processo em uma ferramenta objetiva de file carving, faz-se necessária a abordagem de várias técnicas demonstradas nesta pesquisa e em diversas outras, o que demanda um tempo demasiado, incluindo análises manuais sobre arquivos, sendo praticamente inviável em investigações forenses que usualmente não apresentam tempo suficiente para uma análise complexa dos dados envolvidos.

Sendo assim, a escolha da abordagem ideal para localização dos fragmentos depende de inúmeros fatores que podem estar presentes na imagem investigada, o tipo de sistema de arquivo, os tipos de arquivos mais comuns presentes na imagem investigada e os mais possíveis de se apresentar provas concretas, a situação do nível de fragmentação desses arquivos, as condições dessas informações presentes em slack space. Em grandes volumes de dados, analisar esses quesitos baseado nas informações de dados presentes é uma tarefa muito complexa e custosa. Algumas ferramentas forenses tem voltado seus olhos e estudos para a recuperação de dados, tais como *Foremost* e *Scalpel*, que fazem uso de algumas destas técnicas para aproximar cada vez mais as condições de recuperação dos arquivos, conseguindo assim um resultado mais eficaz e satisfatório, o que não seria garantido se utilizada uma ou outra abordagem.

A análise de entropia ainda está sendo estudada com sua abordagem direcionada ao comportamento de divergência dos dados. Muito embora essa técnica ainda apresente falhas nos processos de identificação de fragmentos, essa é uma das abordagens mais promissoras de melhoria de eficácia no processo de identificação e classificação de fragmentos de dados sendo necessário mais estudos para possibilitar a automatização do processo para evitar a necessidade

de análises manuais sobre imagens forenses.

Referências Bibliográficas

CARRIER, B. *File System Forensic Analysis*. [S.l.]: Addison Wesley Professional, 2005.

COHEN, M. Advanced carving techniques. In: *DFRWS*. [S.l.: s.n.], 2007. p. 32.

DATABASE, F. O. F. S. *File Signatures*. 2013. Disponível em: <http://filesignatures.net/index.php?page=all>.

FITZGERALD, S. et al. Using nlp techniques for file fragment classification. *Digital Investigation*, v. 9, p. S44–S49, 2012.

GARFINKEL, S. L. Carving contiguous and fragmented files with fast object validation. In: *Digital Investigation*. [S.l.: s.n.], 2007. v. 4S, p. S2–S12.

HAND, S. et al. Bin-carver: Automatic recovery of binary executable files. *Digital Investigation*, v. 9, p. S108–S117, 2012.

IBOPE//NETRATINGS, N. *Painel IBOPE/NetRatings*. cetic.br, 2012. Disponível em: <http://www.cetic.br/usuarios/ibope/w-tab02-01-cons.htm>.

KLOET, S. *Measuring and Improving the Quality of File Carving Methods*. Dissertação (Mestrado) — Eindhoven University of Technology, 10 2007.

LI, Q. et al. A novel support vector machine approach to high entropy data fragment classification. In: *Proceedings of the South African Information Security*. [S.l.: s.n.], 2010. p. 236–247.

LTD., D. R. S. How file recovery works. p. 10, 2010.

MAHANT, S. H.; B.B.MESHRAM. Ntfs deleted files recovery: Forensics view. *IRACST - International Journal of Computer Science and Information Technology and Security (IJCSITS)*, v. 2, n. 3, p. 491–497, 2012.

SVENSSON, A. *Computer Forensics Applied to Windows NTFS Computers*. Dissertação (Mestrado) — Stockholm's University / Royal Institute of Technology, 04 2005.

T., R. *Undelete a file in NTFS*. CodeProject, 2013. Disponível em: <http://www.codeproject.com/Articles/9293/Undelete-a-file-in-NTFS>.

WYMAN, B. et al. Metadados. *OUCH!*, 2012.