FACULDADE DE TECNOLOGIA DE JOÃO PESSOA

PÓS-GRADUAÇÃO EM SEGURANÇA DA INFORMAÇÃO

ANÁLISE DE FERRAMENTAS FORENSES DE RECUPERAÇÃO DE DADOS

JOSILENE DOS SANTOS NASCIMENTO

João Pessoa-PB

2010

JOSILENE DOS SANTOS NASCIMENTO

ANÁLISE DE FERRAMENTAS FORENSES DE RECUPERAÇÃO DE DADOS

Monografia apresentada ao Curso de Pós Graduação em Segurança da Informação como requisito parcial para obtenção do título de Especialista em Segurança da Informação.

Orientadora: Klarissa de Souza Jerônimo

João Pessoa-PB

2010

Nascimento, Josilene dos Santos.

Análise de Ferramentas Forenses de Recuperação de dados / Josilene dos Santos Nascimento. -- João Pessoa, 2010.

76f.

Monografia (Curso de Segurança da Informação) – Faculdade de Tecnologia de João Pessoa – FATEC

1. Ciência da Informação. 2. Segurança.

I. Título.

TERMO DE APROVAÇÃO

JOSILENE DOS SANTOS NASCIMENTO

ANÁLISE DE FERRAMENTAS FORENSES DE RECUPERAÇÃO DE DADOS

Monografia aprovada como requisito parcial para obtenção do título de Especialista em Segurança da Informação da Faculdade de Tecnologia de João Pessoa, pela seguinte banca examinadora:

Orientadora: Klarissa de Souza Jerônimo

_____

_____

_____

João Pessoa, 13 de Março de 2010.

Este trabalho é dedicado aos meus queridos pais, Inácia Rosa e Severino Virgínio.

Se tiverdes fé como um grão de mostarda, direis a esta montanha: “transporta-te daqui para lá”, e ela irá e nada vos será impossível. (Evangelho de S Mateus 17:20).

AGRADECIMENTO

A Deus, por ter me dado a vida e a faculdade necessária para não apenas vivê-la, mas também procurar entender a grandeza que ela contém.

Aos meus pais e irmãos, pelo carinho, compreensão e cuidados a mim dirigidos e que me ajudaram a superar mais esta etapa de minha vida.

A minha amiga e orientadora Klarissa, que tanto se dedicou a este trabalho, motivando-me a seguir em frente e ajudando a saltar as pedras que surgiam pelo caminho.

Ao Professor Pedro Segundo, pela dedicação e paciência ao sugerir idéias e ajudar nas correções deste trabalho.

Aos meus professores e colegas, que me orientaram e acompanham nesta jornada de estudos e dias exaustivos.

Aos meus amigos, em especial aos “amigos-X”, que, ao me fazerem companhia e me darem ânimo e incentivo, inclusive dando dicas de formatação e conteúdo, faziam sempre com que as horas em meio às tarefas deste trabalho não parecessem assim tão longas e cansativas.

LISTA DE FIGURAS

LISTA DE TABELAS

LISTA DE GRÁFICOS

LISTA DE ABREVIATURAS E SIGLAS

AVI – Audio Video Interleave (formato de video)

BMP – Windows Bitmap (tipo de formato de imagem)

CD – Compact Disc (tipo de mídia de dados)

DVD – Digital Video Disc (tipo de mídia de dados)

EB – Exabytes (unidade de medida de armazenamento digital)

E-MAIL – Eletronic mail (correio eletrônico)

FAT – File Allocation Table (tabela de alocação de dados no FAT)

GIF – Graphics Interchange Format (tipo de formato de imagem)

HD – Hard Disk (tipo de mídia de dados)

HTM – Hypertext Markup Language (formato de documento para web)

JPEG/JPG – Joint Photographic Experts Group (tipo de formato de imagem)

MB – Megabytes (unidade de medida de armazenamento digital)

MFT – Master File Table (tabela de alocação de arquivos no NTFS)

MP3 – Moving Picture Experts Group 1 (MPEG) Audio Layer 3 (formato de compactação de áudio)

MPG – Moving Picture Group (formato de compactação de video)

MS-DOS – MicroSoft Disk Operating System (sistema de arquivos)

NTFS – New Technology File System (sistema de arquivos)

PDF – Portable Document Format (formato de arquivo)

PGP – Pretty Good Privacy (formato de arquivo para programa de criptografia e decriptografia de dados)

PNG – Portable Network Graphics (tipo de formato de imagem)

RA – Real Audio (formato de audio)

RIFF – Resource Interchange File Format (meta formato de armazenamento de dados)

RPM – Red Hat Package Manager (formato de gerenciador de pacotes)

TB – Terabytes (unidade de medida de armazenamento digital)

TSK – The Sleuth Kit (ferramenta forense)

WMV – Windows Media Video (formato de video)

SUMÁRIO

CAPÍTULO 3

RESUMO

A recuperação de dados, sensíveis ou comuns, é muito necessária atualmente devido ao enorme uso de equipamentos de armazenamento de dados. Este procedimento é adotado tanto por usuários comuns, que têm os seus arquivos perdidos por acidente, como por investigadores forenses em busca de evidências de crimes, sejam eles digitais ou não. Este trabalho analisa ferramentas de recuperação de dados forenses com o objetivo de estabelecer como e em que cenários elas podem ser usadas. O tipo de análise forense escolhido é o *Post Mortem*, com imagens de disco sendo usadas para os testes. Quatro ferramentas são analisadas: Foremost, Scalpel, Autopsy/The Sleuth Kit e FTK Imager, sendo uma delas proprietária e as demais em código livre. Sete tipos de sistemas de arquivos são usados para a criação das imagens de disco: Ext2, Ext3, Ext4, ReiserFS, FAT16, FAT32 e NTFS. Os dados são submetidos a quatro cenários diferentes. No primeiro, os arquivos são simplesmente apagados de partições de 100MB. No segundo, as partições de 100MB, que tiveram seus dados apagados, são completamente preenchidas por outros dados. No terceiro, os arquivos são simplesmente apagados de partições de 200MB. E, no quarto, um arquivo que corresponde a 35% do disco é copiado para as partições de 200MB, que tiveram seus dados apagados. As ferramentas são usadas para tentar a recuperação destes dados, com cada uma delas mostrando comportamentos diferentes dependendo do cenário e do sistema analisado.

ABSTRACT

Data recovery, either with common or sensitive data, is highly necessary nowadays, due to the massive use of data storage devices. This procedure is performed by regular users, who lose data accidentally, as well as by forensic investigators looking for evidences in crimes, digital or not. This paper is an analysis of forensic data recovery tools and it aims to establish how and in which situations such tools may be used. Post Mortem was the type of forensic analysis chosen in this work, and disk images were used for testing. Four tools were analyzed: Foremost, Scalpel, Autopsy/The Sleuth Kit and FTK Imager, one of them proprietary and the others, open source. Seven types of file systems are used for creating disk images: Ext2, Ext3, Ext4, ReiserFS, FAT16, FAT32 and NTFS. Data is submitted to four different scenarios: on the first one, data is simply deleted from 100MB partitions; on the second one, the 100MB partitions which had their data deleted were completely filled with other data; on the third one, data is simply deleted from 200MB partitions; and, on the fourth one, one file, equivalent to 35% of the disk, is copied to the 200MB partitions which had their data deleted. The tools are applied to attempt data recovery, each one behaving differently depending on the scenario and file system analyzed.

INTRODUÇÃO

Atualmente, a grande maioria dos dados e documentos utilizados para o desenvolvimento de quaisquer trabalhos é guardada dentro de dispositivos de armazenamento digital. A facilidade da internet para troca de arquivos e informações e o tamanho cada vez maior dos discos rígidos (HDs) disponíveis no mercado, aliados à grande massa de equipamentos para geração de dados digitais, como gravadores de voz, câmeras digitais, celulares, dentre outros, fazem com que os computadores sejam uma das melhores fontes de informações acerca das atividades que uma pessoa, seja ela física ou jurídica, desempenha dentro e fora de seu campo profissional.

Numa cena de crime, um bom lugar para procurar vestígios de atividades ilícitas e para traçar um perfil da pessoa investigada são os equipamentos digitais de que ela faz uso. No entanto, não apenas os dados que estão aparentes são importantes. Muitos dados que podem vir a ser vitais para um caso investigado podem ter sido apagados propositadamente na intenção de evitar problemas futuros ou por não serem mais de interesse do proprietário. Acrescente-se a isso o fato de que muitos dos atacantes de computadores, fazendo uso de técnicas anti-forenses, apagam os arquivos utilizados no intuito de esconder o registro de suas atividades.

Segundo Vacca (2005), é sabido que os arquivos apagados de um sistema não são necessariamente sobrescritos e ainda podem ser recuperados usando ferramentas e procedimentos adequados.

Este trabalho tem como objetivo principal fazer uma análise das ferramentas forenses disponíveis em software livre que são mais comumente usadas para a recuperação destes arquivos dentro de um sistema de arquivos. Adicionalmente, pretende-se incluir ferramentas proprietárias utilizadas em perícias forenses de informática.

Serão utilizados métodos de pesquisa bibliográfico e experimental, onde o pesquisador testará as ferramentas dentro de um ambiente controlado.

CAPÍTULO 1

FUNDAMENTAÇÃO METODOLÓGICA

Neste capítulo, serão tratadas as justificativas e os objetivos que motivaram este trabalho e a metodologia que o guiou.

1.1 JUSTIFICATIVA

A investigação forense digital, segundo Vacca (2005), é um conjunto de métodos utilizados para coleta, preservação, análise e documentação de dados computacionais. Estes procedimentos ajudarão na composição do cenário e linha de tempo de um crime envolvendo dispositivos eletrônicos. Com estes dados devidamente analisados em mãos, tanto os culpados poderão ser apontados e levados à justiça, como medidas poderão ser tomadas para que o mesmo evento não volte a ocorrer.

A segurança em qualquer sistema atual é de extrema importância devido aos inúmeros ataques que pessoas não autorizadas impõem a todo e qualquer serviço disponibilizado na Internet e a computadores e dispositivos que guardem dados críticos. Os atacantes buscam quaisquer tipos de informações que lhes dê acesso a uma forma de obter lucros ou que lhes facilite esta prática, roubando senhas de usuários de bancos, de correios eletrônicos, perfis de comunidades virtuais, dentre outros. As intenções podem ser variadas, como espionagem empresarial, adulteração de dados em sistemas de informação, calúnia, difamação, pirataria, fraudes bancárias.

Como estes tipos de fraudes eletrônicas aumentam a cada dia, faz-se necessário garantir a segurança dos dados que são trafegados na rede e dos que são armazenados em

mídias diversas. Devem ser garantidas tanto a integridade como a confidencialidade desses dados e dos conteúdos acessados.

Quando essa segurança é quebrada ou mesmo quando há dispositivos de armazenamento eletrônico de dados sendo supostamente usados para atividades ilícitas, faz-se necessário que haja uma investigação onde se apure como estes crimes aconteceram e quem os executou.

O que se sabe, também, é que os crimes digitais não são apenas os que são cometidos em redes e computadores alheios por pessoas não autorizadas, os seus donos ou usuários também podem usá-los para este fim.

Muitos crimes, sejam eles ocorridos em meios eletrônicos ou não, podem deixar rastros em dispositivos digitais. O princípio de troca de *Locard* diz que quando dois objetos quaisquer entram em contato, ambos levam algo do outro. Segundo Chisum (2000), pode-se dizer que um criminoso ao praticar o seu ato ilícito sempre deixa para trás algo de si e/ou da sua atividade. Isto é o que podemos chamar de evidência ou vestígio. Um investigador, ao analisar uma cena de crime, neste caso um dispositivo eletrônico, utilizando as técnicas da forense digital buscará estas evidências e vestígios que possam lhe ajudar a compor o passo a passo do acontecimento ou atividade ilícita.

Para se ter certeza de que as evidências corresponderão à realidade e que estas poderão ser atestadas como tal, os dados devem ser colhidos seguindo métodos determinados que os mantenham íntegros e que não possam ser refutados posteriormente.

Para que o dado seja aceito como evidência, então, alguns princípios devem ser seguidos. Segundo Ieong (2006), são eles:

- **Reconhecimento:** o investigador deve exaurir todos os recursos em mãos para descobrir, coletar, extrair, preservar e analisar os dados de forma que este se torne uma evidência válida.

- **Confiança:** para que o dado coletado não seja repudiado, deve-se garantir que ele seja copiado tal qual o é na realidade, ou seja, deve-se manter tanto sua integridade no conteúdo como a data e hora do último acesso, seus donos e permissões etc.

- **Relevância:** deve-se garantir que apenas os dados relevantes, ou seja, que possam ser realmente aproveitados para a resolução do caso em questão sejam coletados para que os custos e o tempo da investigação sejam gastos da melhor forma possível.

Para que uma ferramenta seja usada, ela deve garantir que todos os princípios anteriores sejam cumpridos, alterando o mínimo possível o estado da máquina sob investigação e o dado em si. Quaisquer erros podem ocasionar a perda da evidência.

Entretanto, não são apenas os aspectos jurídicos que dificultam a busca dessas evidências. São muitos os tipos de dispositivos de armazenamento eletrônico disponíveis no mercado e muitos são os sistemas e aplicações que fazem a interface entre homem e máquina. A particularidade na gravação e disposição de dados dentro dos sistemas de arquivos será abordada no Capítulo 2.

Este trabalho será focado na obtenção de dados que tenham sido removidos de um sistema de arquivos, através de ferramentas e procedimentos adequados. Serão avaliados aqui tanto as ferramentas como esses procedimentos.

1.2 OBJETIVOS

O objetivo principal deste trabalho é evidenciar a qualidade e o comportamento de ferramentas de recuperação de dados frente a diferentes tipos de sistemas de arquivos.

As ferramentas escolhidas são em sua maioria licenciadas como software livre, sendo que apenas uma delas é de código proprietário. Esta última foi incluída com o objetivo de testar as primeiras, fazendo uso da comparação qualitativa.

Dentro do escopo do trabalho está verificar que ferramentas podem ser utilizadas dentro de uma investigação forense, quais as que mesmo não tendo qualidade como prova, ainda podem ser usadas por usuários comuns ou administradores de rede e como seus resultados podem ser utilizados dentro de uma investigação formal ou informal.

Será mostrado como usar cada ferramenta dependendo da situação e do tipo de investigação, verificando-se quais os passos que deverão ser seguidos para obter o dado apagado do sistema.

1.3 METODOLOGIA

A pesquisa buscou levantar, dentre *n* ferramentas, quais as melhores, quais as que melhor se adéquam dentro de um determinado ambiente, dependendo do tipo de investigação e dos tipos de dados que quer se buscar, usando um modelo de estudo qualitativo.

Inicialmente, foi realizada uma pesquisa bibliográfica com o objetivo de levantar a situação atual em que se encontra o objeto de estudo. Foram utilizados livros, dissertações, artigos e dados coletados na Internet.

Além disso, foi realizada uma pesquisa experimental com os objetos de estudo sendo expostos a situações controladas em ambiente de laboratório cujos resultados foram avaliados pelo pesquisador de forma qualitativa e quantitativa. Tem-se como conceito de pesquisa experimental, segundo Gil (2002, p. 47), os seguintes procedimentos: “determinar um objeto de estudo, selecionar as variáveis que seriam capazes de influenciá-lo, definir as formas de controle e de observação dos efeitos que a variável produz no objeto”.

Depois de coletados os dados necessários, quais ferramentas e/ou procedimentos foram utilizados, como e para quê, foi realizada uma análise comparativa de cada uma delas e entre as de mesma natureza, observando quais benefícios elas trazem para o objetivo final que é a obtenção dos dados e recuperação de arquivos.

CAPÍTULO 2

FUNDAMENTAÇÃO TEÓRICA

Neste capítulo, é mostrada a fundamentação teórica necessária para o melhor entendimento do estudo de caso proposto. São estabelecidos conceitos relacionados com a computação forense e alguns de seus métodos de análise. Em seguida, é dada uma visão geral sobre os sistemas de arquivos que constituíram os ambientes para os exames deste trabalho. E, por fim, são detalhadas as ferramentas que são o objeto do estudo em si.

2.1 COMPUTAÇÃO FORENSE

Computação forense é o termo usado para designar a ciência de investigação criminal aplicada em sistemas digitais (MELO, 2009). Esta é a ciência usada pelos peritos quando são encontrados vestígios que envolvam equipamentos que possam armazenar algum tipo de dado eletrônico numa cena de crime, mas suas práticas também podem ser utilizadas por administradores de sistemas que suspeitem que a sua rede esteja sendo usada por pessoas ou para finalidades não autorizadas, mas não desejem iniciar um processo judicial propriamente dito.

Antes de começar a coleta dos dados, deve-se primeiro planejar como a investigação será realizada para que evidências sensíveis e voláteis não sejam perdidas.

Um aspecto a ser considerado dentro de uma investigação é se ela será executada em um sistema desligado ou em uso. Alguns dados críticos somente podem ser obtidos com a máquina ainda ligada e em rede, esta é a chamada análise viva (*Live Analysis*). Quando estes dados não forem necessários ou já terem sido coletados, o computador poder ser desligado e

seu sistema analisado sem o perigo de este ser alterado. Esta análise é conhecida como *Post Mortem* (MELO, 2009).

A análise *Post Mortem* deve ser escolhida quando o sistema a ser analisado possa ser desligado e o dispositivo de armazenamento recolhido. Para isso, os dados são transferidos bit a bit (ou bloco a bloco) para outro local e esta cópia será analisada enquanto o original é mantido a salvo de modificações em um local adequado. Com o clone da mídia em mãos, o investigador seguirá o planejamento inicial para encontrar dados que servirão de prova para uma determinada hipótese.

No entanto, muitas vezes este tipo de análise não pode ser efetuado por motivos que fogem ao controle do perito. Segundo Melo (2009), o crescimento dos HDs, com alguns chegando a 1TB (um terabyte) de tamanho, faz com que esta cópia bit a bit torne-se cada vez mais complicada e dispendiosa. Outro fator que impede o desligamento de uma máquina para a cópia de sua mídia de armazenamento é de questão econômica. Eventualmente, o computador a ser examinado é um servidor de cujo funcionamento depende toda uma empresa. Cada minuto com um serviço fora do ar pode acarretar perdas de grandes valores em multas e/ou clientes. Mas, mesmo quando os dois motivos anteriores não se enquadrem no momento, alguns dados periciais simplesmente se perdem ao se desligar o sistema, como, por exemplo, clientes *logados* na máquina, dados da memória volátil, arquivos abertos no momento, processos ativos etc.

Sendo assim, outra técnica chamada Análise Viva (*Live analysis*) é usada. Esta abordagem está sendo cada vez mais seguida quando os dados voláteis do sistema são de suma importância para o caso investigado ou quando o sistema, por algum motivo, não possa ser desligado (MELO, 2009).

Alguns problemas neste tipo de investigação podem ser apontados. Um deles é que, ao se extrair um dado, o estado da máquina e outros dados são modificados. Outro que se pode observar é que os diversos dados dentro de uma máquina, dependendo de como eles são armazenados, têm níveis de volatilidade diferente. Sendo assim, deve-se respeitar a ordem de volatilidade dos dados no momento de extraí-los, sempre observando a relevância dos mesmos dentro do caso.

Ao analisar um sistema ainda ligado, o perito deve ter cuidado para afetá-lo o mínimo possível. Ou seja, qualquer aplicativo utilizado como toda e qualquer dependência necessária

para o seu funcionamento devem estar instalados na máquina do investigador ou em um CD especialmente preparado para isso. Este cuidado em especial deve ser tomado não apenas para não afetar o sistema em si, mas também para assegurar que os aplicativos estão livres de possíveis alterações que um atacante possa ter efetuado nos mesmos. Além disso, todos os dados colhidos devem ser gravados em outra mídia juntamente com seus *hashes*[1].

Outro tipo de investigação é a forense de rede, que é usada quando os dados periciais estão em arquivos e aplicações usadas para fazer computadores se ligarem em rede ou nos dados provenientes desta ligação. Neste tipo de análise, dispositivos de rede são destrinchados e ferramentas apropriadas são usadas para captar tráfego e estado de rede.

Neste trabalho será focada a análise *Post Mortem*, dado que serão testadas ferramentas de recuperação de arquivos apagados de sistemas de arquivos. No nosso caso, os sistemas são desligados e imagens bit a bit são retiradas.

A seguir, serão apresentados os sistemas de arquivos que foram usados para o desenvolvimento dos exames deste trabalho.

2.2 SISTEMAS DE ARQUIVOS

Diante da proposta deste trabalho de demonstrar a desenvoltura de ferramentas de recuperação de dados, os primeiros fatores influenciadores que se apresentam são os diferentes tipos de sistemas de arquivos. Faz-se necessário, portanto, descrever um pouco estes sistemas.

Usando uma definição simples, pode-se dizer que Sistemas de Arquivos são estruturas lógicas que permitem que um sistema operacional consiga escrever, acessar e organizar dados dentro de um disco rígido. Ou seja, é a forma utilizada pelo sistema para encontrar um dado

[1] Tipo de assinatura de tamanho fixo, gerada a partir de uma entrada de tamanho aleatório. Esta entrada pode ser desde mídias inteiras até mensagens de email.

ou arquivo necessário no disco depois de gravado. Basicamente, todos os sistemas fazem uso de uma tabela onde são associados os metadados de um arquivo e o espaço ocupado por ele no disco.

Para gravação de dados, um HD é dividido em setores. Estes setores são agrupados em blocos ou *clusters* (a forma como é chamado depende do sistema) cujo tamanho é determinado pelo sistema. A Figura 1 exemplifica a localização dos clusters no disco.

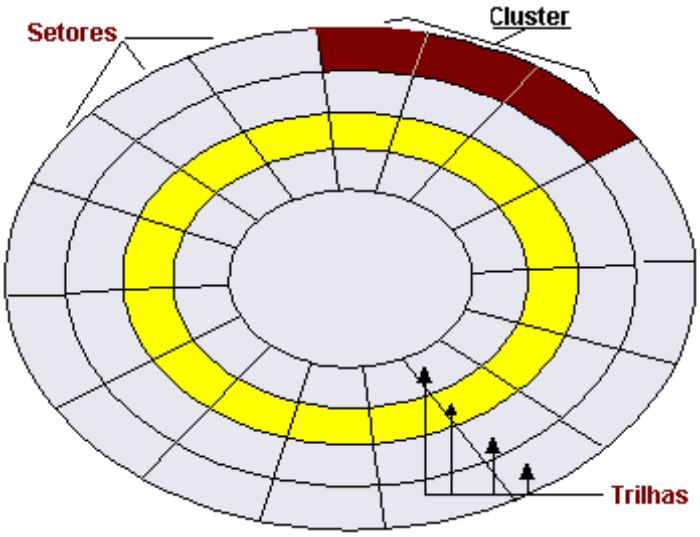

Figura 1 - Divisão do disco em clusters, setores e trilhas (MOURA, 2009)

Cada sistema operacional procura utilizar de um sistema de arquivos próprio para assim otimizar os seus recursos. Nos sistemas Windows, por exemplo, os mais usados ultimamente são o FAT32 e o NTFS. Nos sistemas Linux, os mais usados são o Ext3, ReiserFS e, o mais recente, Ext4, mas também encontramos o JFS e o XFS, entre outros.

Este trabalho focará nos sistemas que foram testados, os quais são: Ext2, Ext3, Ext4, ReiserFS, FAT16, FAT32 e NTFS.

2.2.1 Sistemas de arquivos EXT

O sistema **EXT** (*Extended File System)* é o sistema de arquivos padrão nos sistemas Linux. Atualmente, o mais utilizado é o Ext3, sendo que o Ext4 já é suportado nos sistemas mais novos. O primeiro EXT suportava partições de até 2GB, o que já era muito pouco para a época em que foi lançado, além de ser muito suscetível a fragmentação.

Deste modo, pouco depois surgiu o **Ext2** que trazia um suporte a partições de até 32TB além de diversos outros recursos. Segundo Morimoto (2007), o maior problema com este sistema é o fato de ele não ser bom em recuperação de erros. O fsck[2] neste sistema é extremamente demorado.

O **Ext3** nasceu, então, introduzindo o *journaling*, que é um tipo de *log* mantido durante o processamento dos dados de todas as alterações de arquivos (algo similar é usado no NTFS). Deste modo, mesmo se o sistema não for desligado corretamente, o mesmo pode se recuperar sem a necessidade de checar cada arquivo, observando-se as entradas do *log*. Segundo Morimoto (2007), como o Ext2, o Ext3 utiliza endereçamento com 32 bits, permitindo assim partições de até 32TB.

A escrita de arquivos nos discos é feita primeiramente buscando o primeiro bloco (correspondente ao *cluster* nos sistemas FAT/NTFS) que esteja disponível começando no início do sistema de arquivos. Para diminuir a fragmentação e o movimento da cabeça de leitura do HD, o sistema costuma reservar conjuntos de blocos para diretórios e para arquivos que estejam em crescimento.

O **Ext4**, com endereçamentos de 48bits, permite partições de até 1EB (um exabyte[3]) e torna inexistente o limite para subdiretórios de seu predecessor, que o mantinha em 32000 subdiretórios. Este sistema é, em si, uma evolução e melhoramento do Ext3, assim como este último foi uma evolução do Ext2.

---

[2] Aplicativo de checagem de erros em sistemas Linux.

[3] 1 EB = 1024 PB, 1 PB = 1024 TB, 1 TB = 1024 GB.

Ele também diminui ainda mais a fragmentação dos arquivos com relação ao Ext3, alterando o algoritmo de alocação de arquivos que dentro do sistema Ext4 é feita utilizando-se múltiplos esquemas ao mesmo tempo, segundo Jones (2009):

- **Pré-alocação de arquivos:** um conjunto de blocos adjacentes é reservado para arquivos em edição estimando-se o tamanho do arquivo.
- **Atraso na alocação de blocos:** os dados somente são escritos nos blocos físicos apenas quando for realmente necessário. Quanto mais a escrita for atrasada, mais blocos haverá para serem alocados, aumentando as chances de todos os fragmentos do arquivo ficarem sequenciais.
- **Múltipla alocação de blocos:** no Ext4, é possível a alocação de vários blocos ao mesmo tempo; no Ext3, apenas um é feito por vez. Isto reduz não apenas o processamento como a fragmentação, dado que quanto mais blocos são alocados por vez, menos o algoritmo de alocação é chamado e menor a chance dos fragmentos dos arquivos serem alocados para blocos não contíguos.

Mesmo com todos esses cuidados, a fragmentação em sistemas utilizados por muito tempo é inevitável. Sendo assim, o Ext4 usa o chamado desfragmentador *on-line*, que realoca os arquivos para blocos sequenciais sem a necessidade de uma ferramenta adicional.

Outro fator interessante no que diz respeito a este sistema é o seu fsck. O Ext4 marca os blocos que não estão alocados, então, quando o sistema for verificado, ele lerá apenas os blocos que estão sendo realmente usados, ignorando os demais.

2.2.2 Sistema de arquivos ReiserFS

O **ReiserFS** também possui suporte a *journaling*, mas o seu sistema de gravação no *log* difere do Ext3. Este último grava todas as alterações em arquivos, incluindo dados e metadados, enquanto o ReiserFS guarda em *log* apenas informações sobre os metadados. Isto

torna a recuperação do sistema mais rápida, mas impossibilita a recuperação de arquivos que estiverem sendo editados no instante do desligamento acidental ou abrupto.

No que se refere à gravação e alocação de espaço para dados, segundo Courtney (2001), o sistema ReiserFS utiliza o espaço exato de que os arquivos precisam, alocando-os não em blocos como nos demais sistemas supracitados, mas numa estrutura de árvore. Isso significa um bom aproveitamento de espaço quando o sistema for usado para guardar grandes quantidades de arquivos pequenos.

2.2.3 Sistemas de arquivos FAT

O **FAT16** é um sistema que foi derivado do FAT (*File Allocation Table*) e divide o HD em *clusters*. Segundo Morimoto (2007), cada qual possui um endereço único de 16 bits (de onde proveio o seu nome). Esta limitação no endereçamento (2^16 endereços possíveis) somada ao fato de cada *cluster* comportar 32KB de espaço faz com que cada partição usando este sistema possa ter apenas 2GB. O tamanho do *cluster* acarreta um grande desperdício de espaço, porque muitos *clusters* não são preenchidos completamente. Dispositivos menores ainda usam este sistema, como cartões de memória, *pendrives* etc., já que partições menores podem utilizar tamanhos menores de *clusters*, diminuindo o espaço perdido.

O aumento nos tamanhos dos HDs fez com que sistemas FAT16 se tornassem rapidamente obsoletos para o uso nestas mídias de armazenamento. Foi criado, então, o **FAT32** que usa um endereçamento de 32 bits. Este aumento por si só faz com que este sistema permita partições muito maiores, podendo chegar a até 2TB. Além disso, segundo Morimoto (2007), partições menores de até 8GB podem usar *clusters* menores, conseguindo, assim, uma boa economia de espaço. A maior limitação deste sistema consiste nele não conseguir gravar arquivos maiores que 4GB, o que impossibilita a gravação de arquivos como imagens de DVDs, por exemplo.

O sistema de endereçamento nestes sistemas é feito utilizando um índice (FAT) e um diretório raiz com a localização de cada arquivo de acordo com os seus respectivos *clusters*.

O processo de alocação de *clusters* dentro destes sistemas, segundo Carrier (2005), funciona da seguinte forma: o próximo conjunto de setores não ocupados é procurado para alocação. Ou seja, se o último arquivo foi alocado para o *cluster* 85 e houver um vazio de número 97, será este o próximo a ser usado em um novo processo de escrita, mesmo que haja *clusters* vazios antes do 85.

Cada arquivo é gravado no primeiro *cluster* encontrado livre. Este tipo de gravação faz com que a fragmentação seja grande na medida em que arquivos são apagados e outros são copiados nos *clusters* abandonados, o que acarreta um acesso de arquivos lento, diminuindo a velocidade do sistema como um todo. Em sistemas assim, é interessante que seja usado periodicamente programas de desfragmentação de discos. Estes programas copiam os arquivos para *clusters* sequenciais, diminuindo a movimentação da cabeça de leitura do HD.

2.2.4 Sistema de arquivos NTFS

As limitações do sistema FAT, somadas ao seu baixo desempenho e pouca segurança em ambientes de servidores, fizeram com que a Microsoft pensasse em outro sistema mais robusto, o **NTFS** (*New Technology File System*). Este sistema já nasceu suportando um endereçamento de *cluster* de 64bits com um tamanho de 512 bytes para cada um deles independente do tamanho da partição. Com este tamanho, no entanto, o que seria ganho em questão espaço se perderia em desempenho já que o processador teria muito mais trabalho de encontrar os dados em meio a um número tão grande de *clusters*. Sendo assim, ficou estabelecido que para partições maiores que 2G, o tamanho dos *clusters* seria fixado em 4KB por padrão (este valor pode ser modificado pelo usuário) (MORIMOTO, 2007).

Os atributos de arquivo armazenados na MFT (*Master File Table*), como é chamada a tabela de alocação no NTFS, são mais complexos que no FAT. Neste último, apenas são

gravados atributos de arquivos, diretório etc. Segundo Morimoto (2007), na MFT, são gravados também seu nome MS-DOS, dados para uma possível futura auditoria e informações de permissões. Muitas vezes, toda esta informação se torna grande demais para ser armazenada na MFT. Quando isto ocorre, ela é gravada em *clusters* livres do HD e a MFT apenas aponta onde estão.

O sistema NTFS grava os arquivos no disco de forma inteligente, procurando armazenar os dados dos arquivos de forma que eles fiquem em áreas sequenciais do HD. Para alocação de novos arquivos, segundo Carrier (2005), o sistema NTFS utiliza o esquema de tentar gravar os dados na menor área em que ele consiga se encaixar totalmente. Esta é uma maneira de otimizar a ocupação da partição e reduzir a fragmentação. Por exemplo, temos um arquivo que ocuparia cinco clusters para ser armazenado. Se o primeiro conjunto de clusters sequenciais encontrado tiver um tamanho 10, outro mais à frente tiver 7 e nenhum outro tiver tamanho entre 5 e 7, este último será o escolhido.

Para comparar o desempenho entre um sistema FAT e um NTFS, deve-se levar em consideração qual o uso que será dado para ele. O NTFS, por ser mais complexo, funciona melhor quando aplicado em sistemas que utilizam grandes volumes de arquivos e possuem bons processadores. Já o FAT é mais utilizado em câmeras fotográficas, celulares etc., onde a manipulação de dados é mais simples e o tamanho das partições, como já dito anteriormente, é menor.

2.3 FERRAMENTAS FORENSES

Dados são apagados continuamente em um sistema de arquivos. Mas o que se sabe é que nenhum arquivo é realmente apagado de um disco a não ser que seja sobrescrito por outros dados. Apagar o conteúdo ocupado por um arquivo no disco gastaria tempo e um trabalho considerável do Sistema Operacional e do disco rígido propriamente dito. Em vez disso, o Sistema Operacional adota um método mais prático e otimizado, que consiste em marcar como “livres para uso” tanto os blocos ocupados pelo arquivo que se deseja apagar

como a entrada correspondente ao arquivo em seu diretório. Ferramentas diversas são desenvolvidas com a finalidade de recuperar dados.

As ferramentas forenses podem recuperar dados diretamente de um dispositivo físico ou lógico, como um disco rígido ou uma partição desse disco, respectivamente. Elas também podem recuperar dados a partir de uma imagem.

Imagem de um sistema de arquivos, de uma partição de dados ou de um disco (CD/DVD/HD) é um termo utilizado para designar uma fotografia ou cópia exata de um momento deste sistema, partição ou disco. Ela não refletirá nenhuma alteração posterior ao instante de sua criação.

A criação de imagens no contexto forense é bastante útil para preservar o dispositivo original de alterações ou para casos em que o sistema a ser examinado precisa continuar em funcionamento no seu ambiente de produção. Cria-se a imagem do sistema para realização dos exames e libera-se a mídia original para uso.

Serão focadas nesta seção apenas as ferramentas que foram alvo dos testes deste trabalho, quais sejam: Foremost, Scalpel, The Sleuth Kit, Autopsy e FTK Imager.

2.3.1 Foremost

A ferramenta Foremost, segundo o seu site oficial, *http://foremost.sourceforge.net/*, é uma aplicação desenvolvida originalmente pela *US Air Force Office of Special Investigations*, sendo usada para recuperação de arquivos baseando-se em seus cabeçalhos, legendas e estruturas de dados internos. Esta é uma ferramenta de código aberto que pode ser instalada sem custos adicionais em sistemas Linux e é usada para recuperação de dados em arquivos de imagens de discos ou diretamente nas mídias de armazenamento. Não é informado nos manuais do aplicativo que sistemas de arquivos são suportados por ele, no entanto, através dos testes realizados para o propósito deste trabalho, podemos afirmar que ele suporta os seguintes sistemas: Ext2, Ext3, Ext4, ReiserFS, FAT16, FAT32 e NTFS.

A instalação do Foremost foi feita com o código fonte da ferramenta obtido de seu site oficial. Este procedimento foi seguido pela razão de que no repositório de pacotes do Ubuntu ainda não estava disponível a versão mais recente desta ferramenta. A versão utilizada foi a 1.5.6.

A utilização dessa ferramenta é feita através da linha de comando do Linux, seguindo a sintaxe a seguir.

Sintaxe da linha de comando do Foremost:

```
foremost [-h][-V][-d][-vqwQT][-b<blocksize>][-o<dir>] [-t<type>][-s<num>][-i<file>]
```

A descrição para as opções acima podem ser obtidas no manual da ferramenta. Segue um exemplo da utilização da ferramenta:

```
foremost –t jpeg -i imagem_01.dd -o foremost_01
```

O comando acima, executado no mesmo diretório do arquivo “imagem_01.dd”, extrairá desse arquivo de imagem todos os arquivos do tipo *jpeg* para a pasta de nome “foremost_01” (esta pasta é criada no momento da extração dos dados, neste caso, na mesma pasta onde o comando é dado).

Seguem os tipos de arquivos que podem ser recuperados pela ferramenta Foremost, segundo o manual que é instalado no sistema junto com a própria ferramenta: jpg, gif, png, bmp, avi, exe, mpg, wav, riff, wmv, mov, pdf, ole, doc, zip, rar, htm e cpp.

Esta ferramenta gera um arquivo em formato de texto com um resumo da operação e uma pasta para cada tipo de arquivo encontrado.

2.3.2 Scalpel

A ferramenta Scalpel, segundo o seu site oficial, *http://www.digitalforensicssolutions.com/Scalpel/*, é uma ferramenta baseada no *foremost-0.69* com a finalidade de melhorar o seu desempenho e o uso de memória durante a análise. Esta é uma aplicação de código aberto (*opensource*) recomendada para ser instalada em sistemas Linux, mas a instalação em outros sistemas é possível bastando para isto compilar os fontes da ferramenta no sistema desejado. A última versão do Scalpel foi lançada em agosto de 2006. Assim como ocorre com o Foremost, não é informado nos manuais do aplicativo que sistemas de arquivos são suportados por ele, mas foi constatado através dos testes neste trabalho que ele suporta os seguintes sistemas: Ext2, Ext3, Ext4, ReiserFS, FAT16, FAT32 e NTFS.

A versão do site oficial do Scalpel e a dos repositórios do sistema utilizado para os testes era a mesma, assim, esta ferramenta foi instalada usando o procedimento mais simples, que é através de aplicativos do próprio sistema. A versão utilizada foi a 1.60.

A utilização dessa ferramenta também é feita através da linha de comando, seguindo a sintaxe a seguir.

Sintaxe da linha de comando do Scalpel:

```
scalpel [-b] [-c <file>] [-d] [-h] [-i <file>] [-m <blocksize>] [-n] [-o <dir>] [-O] [-p] [-r] [-s <num>] [-t] [-u] [-V] [-v] [FILES]...
```

A descrição de cada opção do comando acima pode ser obtida no manual da ferramenta que é instalado junto com a mesma.

Exemplo de uso do Scalpel:

```
scalpel imagem_01.dd -c /etc/scalpel/scalpel.conf -o scalpel_01
```

Os tipos de arquivos que podem ser recuperados pela ferramenta devem ser escolhidos no arquivo de configuração da ferramenta. As opções que estão disponíveis na última versão lançada são: art, gif, jpg, png, bmp, avi, mov, mpg, fws, doc, pst, ost, dbx, idx, mbx, wpc, htm, pdf, mail, PGP, txt, RPM, wav, ra, dat, zip, java, max e pins.

Esta ferramenta gera saídas semelhantes à Foremost.

2.3.3 The Sleuth Kit e Autopsy Browser

The Sleuth Kit (TSK) é um apanhado de ferramentas forenses. Ele é comumente usado em conjunto com o *Autopsy Browser*, que lhe proporciona uma interface gráfica para facilitar o manejo e a visualização do resultado da análise. Ambas são ferramentas de código aberto que, segundo o seu site oficial, http://www.sleuthkit.org/, podem ser instaladas em diversos sistemas Windows ou Unix (Linux, OS X, Cygwin, FreeBSD, OpenBSD e Solaris) e suporta imagens com os sistemas FATx, NTFS, Ext2 e Ext3.

A instalação do TSK e Autopsy foi feita através de seus fontes, pois, assim como ocorre com o Foremost, a versão dos repositórios do sistema é mais antiga que a disponível no site das ferramentas. As versões instaladas foram 3.1.0 e 2.22 para o TSK e para o Autopsy, respectivamente. A ferramenta TSK deve ser instalada antes da Autopsy.

Segue a sintaxe da linha de comando para ativação da ferramenta Autopsy em modo *web*:

```
autopsy [-c] [-C] [-d evid_locker] [-i device filesystem mnt] [-p port] [remoteaddr]
```

A descrição das opções pode ser obtida no manual da ferramenta. A Figura 2 mostra o comando mais simples para ativar o Autopsy, cuja saída informa o endereço a ser utilizado no browser ou navegador para acesso à ferramenta *web*.

```
josilene@josilene-desktop:/opt/autopsy-2.22$ sudo ./autopsy

================================================================================

                    Autopsy Forensic Browser
                http://www.sleuthkit.org/autopsy/
                           ver 2.22

================================================================================
Evidence Locker: /home/josilene/TCC/autopsy
Start Time: Sun Feb 14 15:37:34 2010
Remote Host: localhost
Local Port: 9999

Open an HTML browser on the remote host and paste this URL in it:

    http://localhost:9999/autopsy

Keep this process running and use <ctrl-c> to exit
```

Figura 2 - Ativação da ferramenta Autopsy por linha de comando

Ao digitar no navegador o endereço informado na Figura 2, a tela inicial do Autopsy é apresentada, conforme ilustrado na Figura 3.

A imagem a ser analisada deve ser incluída através do *browser* após a configuração do caso e do *host* (máquina a ser analisada) iniciando ao clicar no botão *"New Case"*, como mostra a Figura 3. A tela para configurar o *host* será apresentada logo após a do caso. Todo o processo é intuitivo.

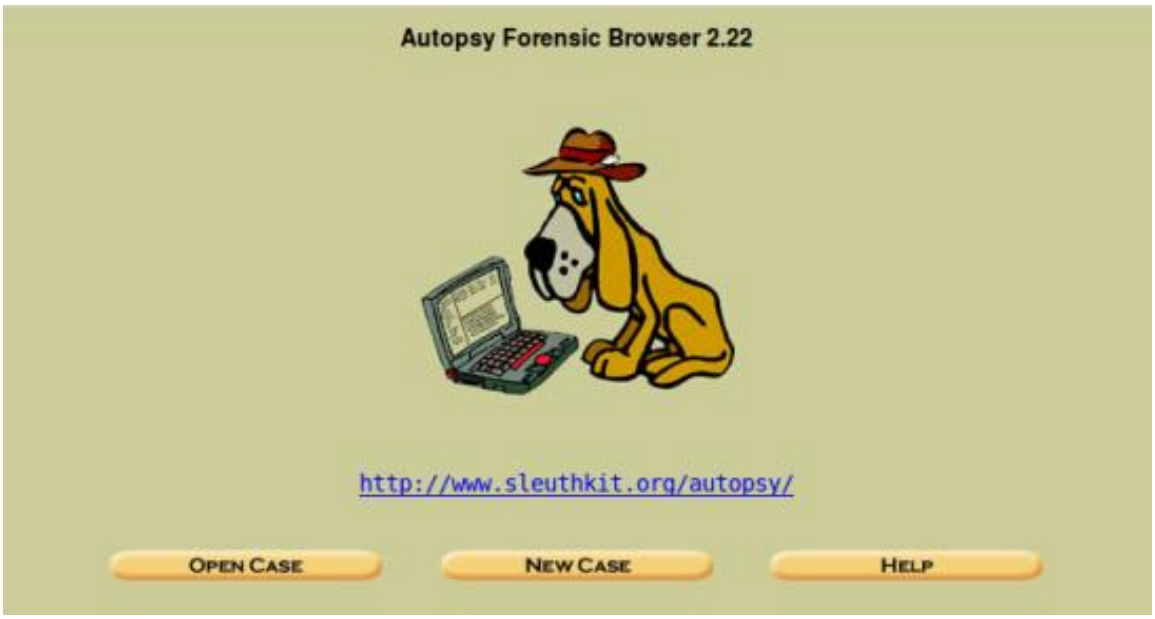

Figura 3 - Tela inicial da ferramenta Autopsy

Ao demandar a análise da imagem escolhida, depara-se com diversas técnicas de busca de evidência da ferramenta, que são todas divididas em abas. Cada uma delas é descrita no site de ajuda da ferramenta, *http://www.sleuthkit.org/autopsy/desc.php*.

Os arquivos recuperados podem ser vistos diretamente no *browser*, gravados todos de uma vez para o diretório de análise da imagem de disco do Autopsy ou apenas os escolhidos são gravados um a um para qualquer diretório determinado pelo investigador.

Outra ferramenta *web* usada como alternativa para o TSK é a PTK (http://ptk.dflabs.com/). Esta ferramenta foi testada, no entanto, não foi incluída nos resultados finais deste trabalho por ser apenas uma interface gráfica como o Autopsy e prover, naturalmente, os mesmos resultados na recuperação de dados que esta última.

2.3.4 FTK Imager

O FTK Imager é um software de criação de imagens e recuperação de dados distribuído gratuitamente pela empresa AccessData através de seu site oficial, http://www.accessdata.com/downloads.html.

A versão usada para os testes neste trabalho (versão 2.5.1) é compatível com sistemas Windows e dispensou a instalação em disco.

Segundo a sua documentação, este software suporta os seguintes sistemas de arquivos: FAT12, FAT16, FAT32, NTFS, Ext2, Ext3, HFS, HFS+ e Reiser.

A Figura 4 dá uma visão geral da ferramenta sendo utilizada para recuperar um arquivo de uma imagem de sistema de arquivos Ext2.

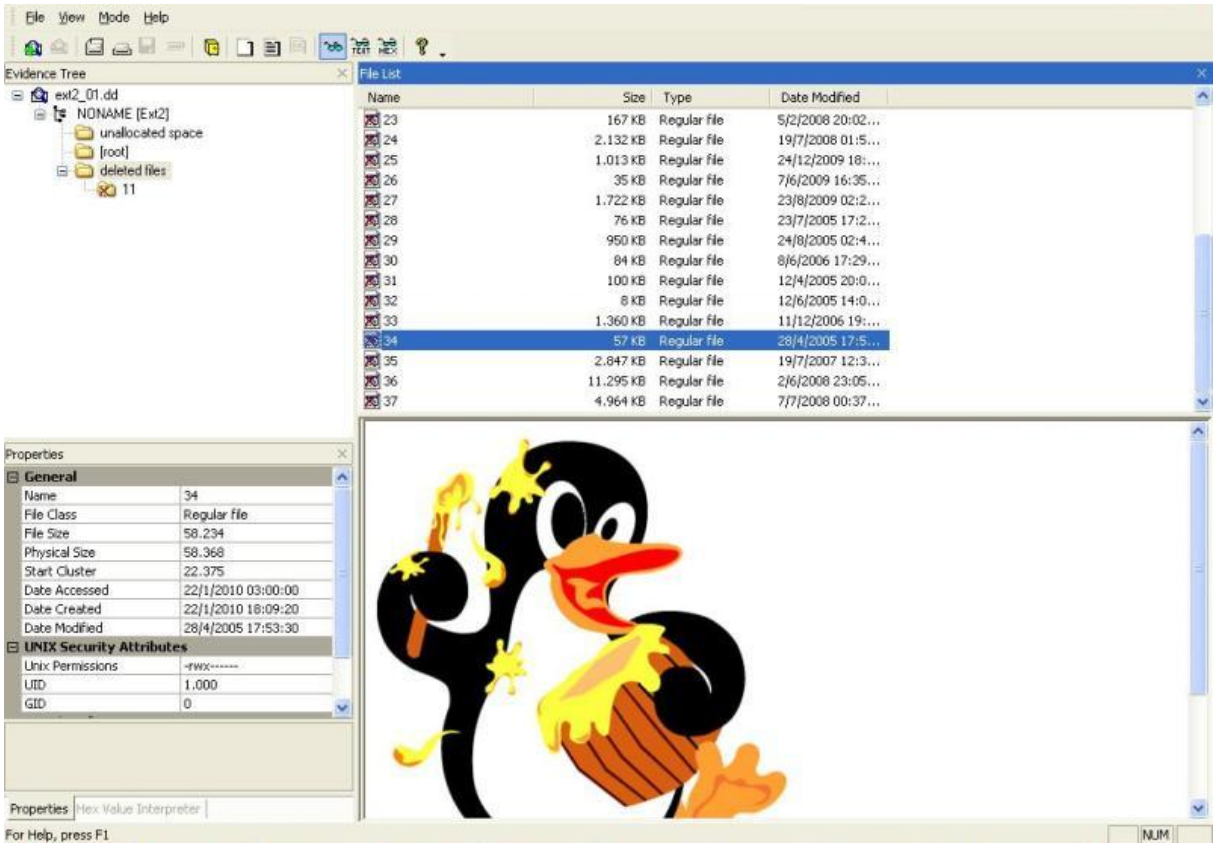

Figura 4 – Visualização de um arquivo recuperando usando o FTK Imager

2.3.5 Evitando a recuperação de dados

Como visto, recuperar dados em um HD, apesar de ser um processo muitas vezes complicado e dispendioso, é perfeitamente possível. Caso os dados armazenados sejam confidenciais, ao descartar um HD é recomendável que se execute alguns procedimentos para que seja eliminada qualquer possibilidade de recuperação de dados.

Apenas sobrescrever os arquivos gravados anteriormente não é o suficiente. O mais recomendável atualmente, com todas as técnicas avançadas de recuperação, é utilizar um aplicativo chamado "shred" e executá-lo via linha de comando como mostrado a seguir:

```
shred -n 5 -vz /dev/hda
```

Desta forma, o programa escreverá, em todo disco (*/dev/hda*), dados aleatórios repetindo o mesmo procedimento (-*vz*) cinco vezes seguidas (-*n 5*). Esta é a forma lógica mais recomendável para eliminar a possibilidade de recuperação de dados, segundo Morimoto (2007). Utiliza-se também o aplicativo wipe, substituindo o conteúdo da mídia por zeros, através de comandos apropriados. Outra forma seria destruir o disco fisicamente.

CAPÍTULO 3

EXAMES

Neste capítulo, serão detalhados os passos seguidos para desenvolver os testes sobre as ferramentas. A escolha das ferramentas e dos tipos de sistemas de arquivos foi feita através de pesquisa sobre quais eram os mais utilizados atualmente.

Os testes foram realizados sobre uma plataforma Linux Ubuntu 9.04, Kernel 2.6.28-17, para os testes com as ferramentas Foremost, Scalpel e TSK/Autopsy; e, sobre Windows XP, SP3, para os testes com a ferramenta FTK Imager. Ambos os sistemas usam uma plataforma de 32 bits.

3.1 MONTANDO OS SISTEMAS DE ARQUIVOS

Os sistemas de arquivos escolhidos para os testes, como já mencionado no Capítulo 2, foram: Ext2, Ext3, Ext4, ReiserFS, FAT16, FAT32 e NTFS.

A instalação dos sistemas de arquivos para os testes foi feita criando-os em arquivos dentro do sistema em uso e montando-os como partições. O passo a passo para a instalação e montagem de cada um deles é descrito na Tabela 1, seguindo um modelo de Jones (2007). Foram escolhidas partições de tamanhos diferentes (100MB e 200MB) para facilitar a manipulação de arquivos dentro delas e a posterior análise dos dados.

As montagens dos sistemas de 100MB e 200MB foram feitas em momentos diferentes, pois os dispositivos de *loopback* (*/dev/loopX*") são limitados em número e cada um não pode ser ocupado totalmente e ao mesmo tempo por dois ou mais sistemas de arquivos.

	Imagens – 100M	Imagens – 200M
Ext2	dd if=/dev/zero of=ext2.img bs=1k count=100000 losetup /dev/loop0 ext2.img mkfs.ext2 -c /dev/loop0 100000 mount -t ext2 /dev/loop0 /mnt/disk-1/	dd if=/dev/zero of=ext2.img bs=1k count=200000 losetup /dev/loop0 ext2.img mkfs.ext2 -c /dev/loop0 200000 mount -t ext2 /dev/loop0 /mnt/disk-1/
Ext3	dd if=/dev/zero of=ext3.img bs=1k count=100000 losetup /dev/loop1 ext3.img mkfs.ext3 -c /dev/loop1 100000 mount -t ext3 /dev/loop1 /mnt/disk-2/	dd if=/dev/zero of=ext3.img bs=1k count=200000 losetup /dev/loop1 ext3.img mkfs.ext3 -c /dev/loop1 200000 mount -t ext3 /dev/loop1 /mnt/disk-2/
Ext4	dd if=/dev/zero of=ext4.img bs=1k count=100000 losetup /dev/loop2 ext4.img mkfs.ext4 -c /dev/loop2 100000 mount -t ext4 /dev/loop2 /mnt/disk-3/	dd if=/dev/zero of=ext4.img bs=1k count=200000 losetup /dev/loop2 ext4.img mkfs.ext4 -c /dev/loop2 200000 mount -t ext4 /dev/loop2 /mnt/disk-3/
ReiserFS	dd if=/dev/zero of=reiserfs.img bs=1k count=100000 losetup /dev/loop3 reiserfs.img mkfs.reiserfs -f /dev/loop3 mount -t reiserfs /dev/loop3 /mnt/disk-4/	dd if=/dev/zero of=reiserfs.img bs=1k count=200000 losetup /dev/loop3 reiserfs.img mkfs.reiserfs -f /dev/loop3 mount -t reiserfs /dev/loop3 /mnt/disk-4/
FAT16	dd if=/dev/zero of=fat16.img bs=1k count=100000 losetup /dev/loop4 fat16.img mkfs.vfat -F 16 /dev/loop4 mount -t vfat /dev/loop4 /mnt/disk-5/	dd if=/dev/zero of=fat16.img bs=1k count=200000 losetup /dev/loop4 fat16.img mkfs.vfat -F 16 /dev/loop4 mount -t vfat /dev/loop4 /mnt/disk-5/
FAT32	dd if=/dev/zero of=fat32.img bs=1k count=100000 losetup /dev/loop5 fat32.img mkfs.vfat -F 32 /dev/loop5 mount -t vfat /dev/loop5 /mnt/disk-6/	dd if=/dev/zero of=fat32.img bs=1k count=200000 losetup /dev/loop5 fat32.img mkfs.vfat -F 32 /dev/loop5 mount -t vfat /dev/loop5 /mnt/disk-6/
NTFS	dd if=/dev/zero of=ntfs.img bs=1k count=100000 losetup /dev/loop6 ntfs.img mkfs.ntfs /dev/loop6 200000 mount -t ntfs /dev/loop6 /mnt/disk-7/	dd if=/dev/zero of=ntfs.img bs=1k count=200000 losetup /dev/loop6 ntfs.img mkfs.ntfs /dev/loop6 400000 mount -t ntfs /dev/loop6 /mnt/disk-7/

Tabela 1 - Passo a passo da criação e montagem das partições utilizadas nos testes

Estas partições, montadas como mostrado na Tabela 1, foram tomadas como base para o preparo das imagens que foram submetidas às ferramentas de análise forense. As imagens são arquivos correspondentes aos sistemas que serão examinados e tiveram seu conceito explicado na seção 2.3.

Os comandos da Tabela 1 realizam para cada sistema de arquivos e tamanho da imagem:

1. Wipe da partição, escrevendo zero em todo o seu conteúdo;
2. Criação do sistema de arquivos na partição, utilizando todo o seu tamanho;
3. Montagem da partição no sistema de arquivos, tornando-a disponível para uso no sistema operacional, onde será possível copiar e apagar arquivos.

A preparação das imagens utilizadas neste trabalho, através de escritas, exclusões e reescritas de dados das mesmas, é organizada em cenários, apresentados na sessão 3.2.

3.2 CENÁRIOS

A abordagem escolhida para os testes foi a de comparar o desempenho de cada ferramenta frente a diferentes sistemas de arquivos e cenários, que serão detalhados mais adiante.

Assim, vinte e cinco (25) arquivos de extensões diversas foram copiados para as partições montadas que, por questão de uniformidade, são os mesmos arquivos usados para todos os testes, os quais são elencados na Tabela 2 e formam ao todo um tamanho de 32,7 MB.

Como o objetivo é a recuperação de arquivos apagados do disco, foram preparados quatro cenários diferentes para a realização dos testes, cada um simulando uma situação em que os arquivos a serem recuperados poderiam ser expostos dentro de um sistema de arquivos.

TIPO	TAMANHO	TIPO	TAMANHO
Figuras JPG	83KB, 35KB	**Áudio WMA**	2311KB
Figuras GIF	1722KB, 40KB, 5KB, 8KB	**Adobe PDF**	212KB
Figuras BMP	2132KB,1013KB	**Apresentações (Power Point)**	80KB, 100KB
Figura PNG	57KB	**Planilhas (Excel)**	18KB, 167KB
Vídeo AVI	11295KB	**Documento (Word)**	76KB, 84KB
Vídeo WMV	4964KB	**Arquivos ZIP**	170KB, 950KB
Vídeo FLV	3660KB	**Arquivos EXE**	161KB, 2847KB
Áudio MP3	1360KB		

Tabela 2 – Arquivos copiados para as imagens, os quais as ferramentas tentarão recuperar

3.2.1 Cenário 1

Este cenário exemplifica o caso em que um arquivo tenha sido apagado do disco e a sua recuperação tenha sido iniciada sem que o sistema tenha sofrido quaisquer alterações. O objetivo é testar o desempenho das ferramentas e o comportamento dos sistemas na tentativa de recuperar os arquivos recentemente apagados.

Foram usadas as partições formatadas de 100MB e em cada uma delas feito o seguinte processo:

- Cópia dos 25 arquivos;
- Desmontagem/montagem;
- Limpeza da partição (remoção dos arquivos);
- Desmontagem/montagem.

O processo de desmontagem/montagem é repetido para apagar quaisquer resquícios que possam ter sobrado no buffer dos sistemas. Isto poderia comprometer os resultados dos testes caso a ferramenta recuperasse a arquivo não por ter podido retirá-lo de dentro da partição, mas por ele ainda estar na memória do sistema.

Depois disso, o seguinte script foi executado e assim obtidas as imagens, "<sistema>_01.dd", prontas para os testes deste primeiro cenário.:

```
#!/bin/bash

##Imagens com os arquivos simplesmente deletados

dd if=/dev/loop0 of=/home/josilene/TCC/img-testes1/ext2_01.dd
dd if=/dev/loop1 of=/home/josilene/TCC/img-testes1/ext3_01.dd
dd if=/dev/loop2 of=/home/josilene/TCC/img-testes1/ext4_01.dd
dd if=/dev/loop3 of=/home/josilene/TCC/img-testes1/reiserfs_01.dd
dd if=/dev/loop4 of=/home/josilene/TCC/img-testes1/fat16_01.dd
dd if=/dev/loop5 of=/home/josilene/TCC/img-testes1/fat32_01.dd
dd if=/dev/loop6 of=/home/josilene/TCC/img-testes1/ntfs_01.dd
```

3.2.2 Cenário 2

Neste cenário, as partições de onde os arquivos foram apagados são agora preenchidas com outros arquivos diversos, sem relação com os primeiros, de forma tal que a partição pareça 100% ocupada. Os arquivos que foram usados na sobrescrição também foram apagados da partição antes da criação da imagem para os testes.

Neste caso, os arquivos escritos e apagados no Cenário 1 têm uma chance quase total de terem sido sobrescritos. Busca-se, então, verificar se as ferramentas conseguem ainda recuperar algum arquivo ou fragmentos quando estes são submetidos a uma situação como esta.

Para este segundo cenário, foram usadas as partições de 100MB antes usadas para compor o Cenário 1. Em cada uma delas foi feito o seguinte processo:

- Cópia de arquivos diversos de forma a preencher todo o espaço da partição;
- Desmontagem/montagem;
- Limpeza da partição;
- Desmontagem/montagem.

Depois disso, o seguinte script foi executado de forma a obter as imagens, "<sistema>_02.dd", que seriam utilizadas para os testes do Cenário 2:

```
#!/bin/bash

#Imagens com os arquivos deletados e sobrescritos com as partições
deixadas com 100% de espaço usado. Os arquivos usados para sobrescricao
também foram deletados

dd if=/dev/loop0 of=/home/josilene/TCC/img-testes1/ext2_02.dd
dd if=/dev/loop1 of=/home/josilene/TCC/img-testes1/ext3_02.dd
dd if=/dev/loop2 of=/home/josilene/TCC/img-testes1/ext4_02.dd
dd if=/dev/loop3 of=/home/josilene/TCC/img-testes1/reiserfs_02.dd
dd if=/dev/loop4 of=/home/josilene/TCC/img-testes1/fat16_02.dd
dd if=/dev/loop5 of=/home/josilene/TCC/img-testes1/fat32_02.dd
dd if=/dev/loop6 of=/home/josilene/TCC/img-testes1/ntfs_02.dd
```

3.2.3 Cenário 3

Para formação do terceiro cenário, o processo do primeiro cenário foi repetido, com exceção de que o tamanho das partições foi maior, 200MB. Esta diferenciação foi necessária para que sobrasse bastante espaço na partição, servindo, assim, aos propósitos dos testes do quarto cenário. Espera-se que o comportamento deste terceiro cenário seja o mesmo do primeiro, com exceção de que neste há mais espaço para os arquivos se espalharem na partição, dependendo do algoritmo de escrita em disco do sistema utilizado.

Para obtenção das imagens para o terceiro cenário, "<sistema>_03.dd, que serão utilizadas para posterior análise das ferramentas de recuperação de dados, foi usado com o script a seguir:

```
#!/bin/bash

##Imagens com os arquivos deletados com o tamanho de aproximadamente
200MB

dd if=/dev/loop0 of=/home/josilene/TCC/img-testes1/ext2_03.dd
dd if=/dev/loop1 of=/home/josilene/TCC/img-testes1/ext3_03.dd
dd if=/dev/loop2 of=/home/josilene/TCC/img-testes1/ext4_03.dd
dd if=/dev/loop3 of=/home/josilene/TCC/img-testes1/reiserfs_03.dd
dd if=/dev/loop4 of=/home/josilene/TCC/img-testes1/fat16_03.dd
dd if=/dev/loop5 of=/home/josilene/TCC/img-testes1/fat32_03.dd
dd if=/dev/loop6 of=/home/josilene/TCC/img-testes1/ntfs_03.dd
```

3.2.4 Cenário 4

Neste cenário, é utilizado para a sobrescrição apenas um arquivo com um tamanho de 71 MB, que é um pouco mais do que o dobro do tamanho do total dos arquivos a serem recuperados. Como as partições a serem usadas são de aproximadamente 200MB, sobra espaço o suficiente para acomodar este arquivo sem que os demais sejam tocados. Ou seja, dependendo da formatação e da manipulação dos dados do sistema testado, os arquivos que deveriam ser recuperados podem ter sido sobrescritos ou não, respectivamente dificultando e facilitando o trabalho das ferramentas de recuperação de dados aqui testadas. Espera-se, portanto, que o desempenho das ferramentas para o quarto cenário seja melhor do que o demonstrado para o segundo cenário. O arquivo usado para a sobrescrição também foi apagado do disco antes da criação das imagens.

Para o Cenário 4, foram usadas as mesmas partições de 200MB utilizadas anteriormente para formar as imagens do Cenário 3. Em cada uma delas foi feito o seguinte processo:

- Cópia de um arquivo qualquer de 71MB;
- Desmontagem/montagem;

- Limpeza da partição;
- Desmontagem/montagem;

Depois disso, foram criadas, usando com o script abaixo, as imagens das partições para o Cenário 4, "<sistema>_04.dd, usadas para posterior análise com as ferramentas de recuperação:

```
#!/bin/bash

##Imagens com os arquivos deletados e sobrescritos de forma a sobrar
espaço com o mesmo tamanho deles. Os arquivos usados para sobrescricao
tambem foram deletados.

dd if=/dev/loop0 of=/home/josilene/TCC/img-testes1/ext2_04.dd
dd if=/dev/loop1 of=/home/josilene/TCC/img-testes1/ext3_04.dd
dd if=/dev/loop2 of=/home/josilene/TCC/img-testes1/ext4_04.dd
dd if=/dev/loop3 of=/home/josilene/TCC/img-testes1/reiserfs_04.dd
dd if=/dev/loop4 of=/home/josilene/TCC/img-testes1/fat16_04.dd
dd if=/dev/loop5 of=/home/josilene/TCC/img-testes1/fat32_04.dd
dd if=/dev/loop6 of=/home/josilene/TCC/img-testes1/ntfs_04.dd
```

Os arquivos das imagens resultantes para cada um dos cenários são listados na Tabela 3.

	Cenário 1	**Cenário 2**	**Cenário 3**	**Cenário 4**
Ext2	ext2_01.dd	ext2_02.dd	ext2_03.dd	ext2_04.dd
Ext3	ext3_01.dd	ext3_02.dd	ext3_03.dd	ext3_04.dd
Ext4	ext4_01.dd	ext4_02.dd	ext4_03.dd	ext4_04.dd
ReiserFS	reiserfs_01.dd	reiserfs_02.dd	reiserfs_03.dd	reiserfs_04.dd
FAT16	fat16_01.dd	fat16_02.dd	fat16_03.dd	fat16_04.dd
FAT32	fat32_01.dd	fat32_02.dd	fat32_03.dd	fat32_04.dd
NTFS	ntfs_01.dd	ntfs_02.dd	ntfs_03.dd	ntfs_04.dd

Tabela 3 - Lista dos arquivos de imagens correspondentes aos cenários

3.3 DESCRIÇÃO DOS EXAMES

A realização dos exames para cada ferramenta é detalhada nesta seção. Será descrito como foi executado cada teste e qual o comportamento para cada ferramenta.

3.3.1 Exame 1 - Foremost

Os exames com o Foremost foram realizados através de uma linha de comando para cada imagem. Nenhuma configuração adicional foi feita na ferramenta depois de sua instalação, que também seguiu o padrão.

Para executar a ferramenta e inicializar o teste foi usado o comando abaixo, o qual foi repetido para cada sistema e cenário:

```
foremost -i <imagem> -o <pasta_de_destino>
```

Com este comando, todos os tipos de arquivos suportados pela ferramenta seriam buscados dentro da imagem especificada. Um exemplo de como foi realizado o teste é mostrado abaixo. O teste mostrado no exemplo foi realizado sobre uma imagem do Cenário 1 com o sistema Ext2. A saída da ferramenta foi gerada numa pasta de nome “ext2_01” no mesmo local de onde o comando foi executado.

```
foremost -i ext2_01.dd -o ext2_01
```

Desta forma, para cada cenário e sistema o teste era executado e, assim, era gerada uma pasta com os arquivos recuperados separados em pastas individuais de acordo com a sua extensão como mostrado na Figura 5.

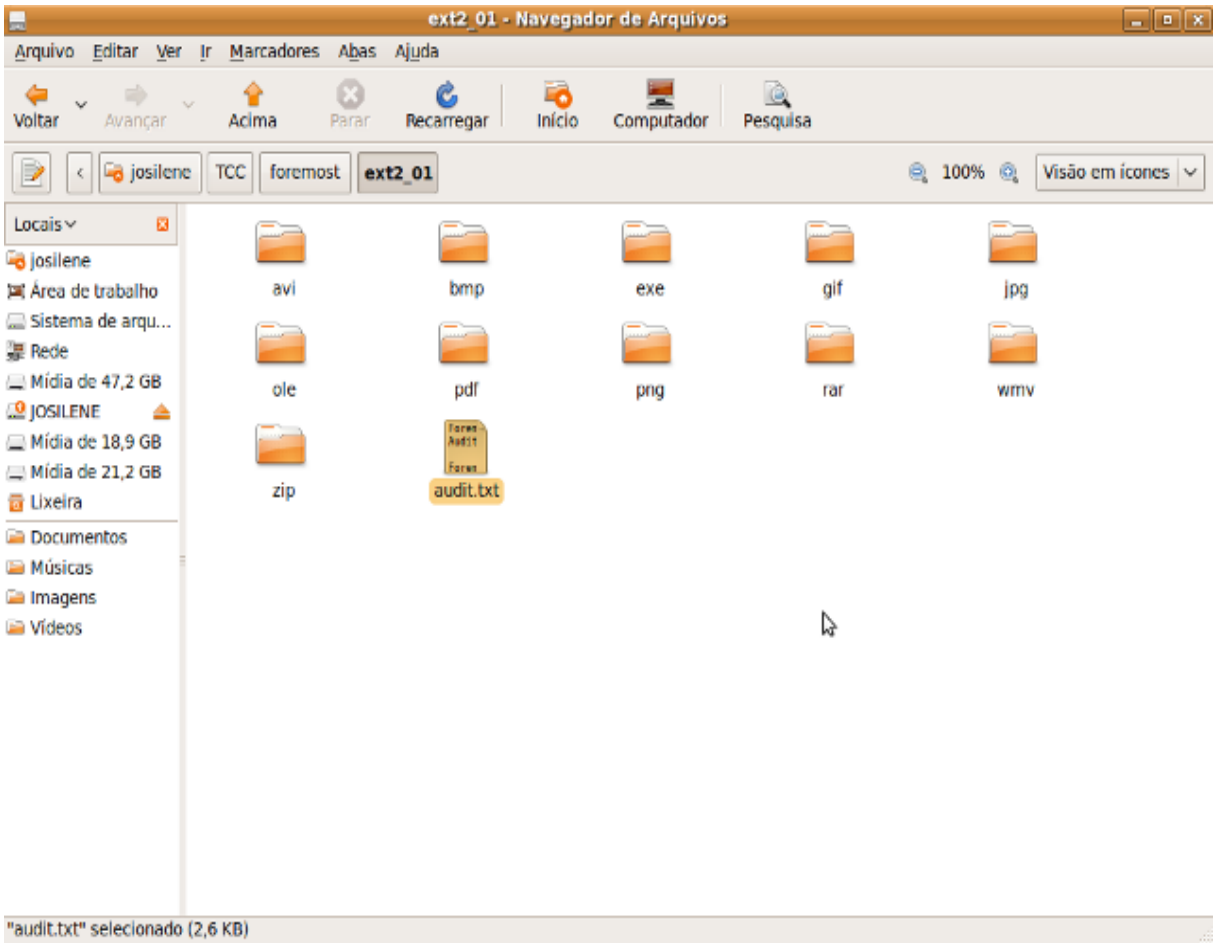

Figura 5 – Saída gerada pelo foremost

3.3.2 Exame 2 - Scalpel

Os testes do Scalpel e as saídas geradas pelos comandos são semelhantes aos do Foremost. No entanto, para que o teste funcionasse, foi necessário que as configurações padrão da ferramenta fossem modificadas. Diferentemente do Foremost, a escolha do tipo de

arquivo a ser recuperado não é feita por linha de comando, mas diretamente em seu arquivo de configuração, “scalpel.conf”. Um trecho dele pode ser visto a seguir:

```
#-----------------------------------------------------------------
# HTML
#-----------------------------------------------------------------
#
#     htm   n     50000   <html                </html>
#
#-----------------------------------------------------------------
# ADOBE PDF
#-----------------------------------------------------------------
#
      pdf   y     5000000     %PDF  %EOF\x0d    REVERSE
      pdf   y     5000000     %PDF  %EOF\x0a    REVERSE
```

Para os propósitos deste teste, foram desmarcadas, ou seja, foram retirados os caracteres “#” das linhas correspondes a todas as extensões que se pretendia recuperar. Depois de o arquivo ter sido devidamente salvo, os testes foram feitos seguindo o modelo abaixo para todos os cenários e sistemas:

```
scalpel <imagem> -c /etc/scalpel/scalpel.conf -o <pasta_de_destino>
```

Exemplo de teste para o Cenário 1 sobre uma imagem do sistema Ext2:

```
scalpel ext2_01.dd -c /etc/scalpel/scalpel.conf -o ext2_01
```

Como saída, este comando gera uma pasta de nome “ext2_02”, como especificado, e os arquivos são divididos em subpastas dependendo de suas extensões.

3.3.3 Exame 3 - TSK/ Autopsy

Os exames realizados com o TSK são feitos através de uma ferramenta gráfica via *web*, o Autopsy, conforme explicado na seção 2.3.3.

Para ativar o Autopsy, foi usada, dentro da pasta onde a ferramenta foi compilada, uma linha de comando como já mostrado no Capítulo 2 na Figura 2.

Para iniciar o teste, faz-se necessário adicionar as imagens a serem analisadas na ferramenta. No entanto, primeiro foi preciso configurar o caso e o *host* do teste. Para isso, seguiu-se o procedimento de clicar em *"New Case"* (vide Figura 3), preencher os campos como solicitado, confirmar; depois, na página que se abriu, clicar em *"New Host"*, novamente preenchendo o que foi pedido e, só então, escolher a imagem.

Para analisar a imagem incluída, basta selecioná-la e clicar no botão *"Analyse"* como mostrado na Figura 6. A tela inicial de análise de imagem do Autopsy é então aberta. Para obter uma listagem dos arquivos a serem recuperados, basta clicar no botão *"File Analysis"* e o resultado é como está ilustrado na Figura 7.

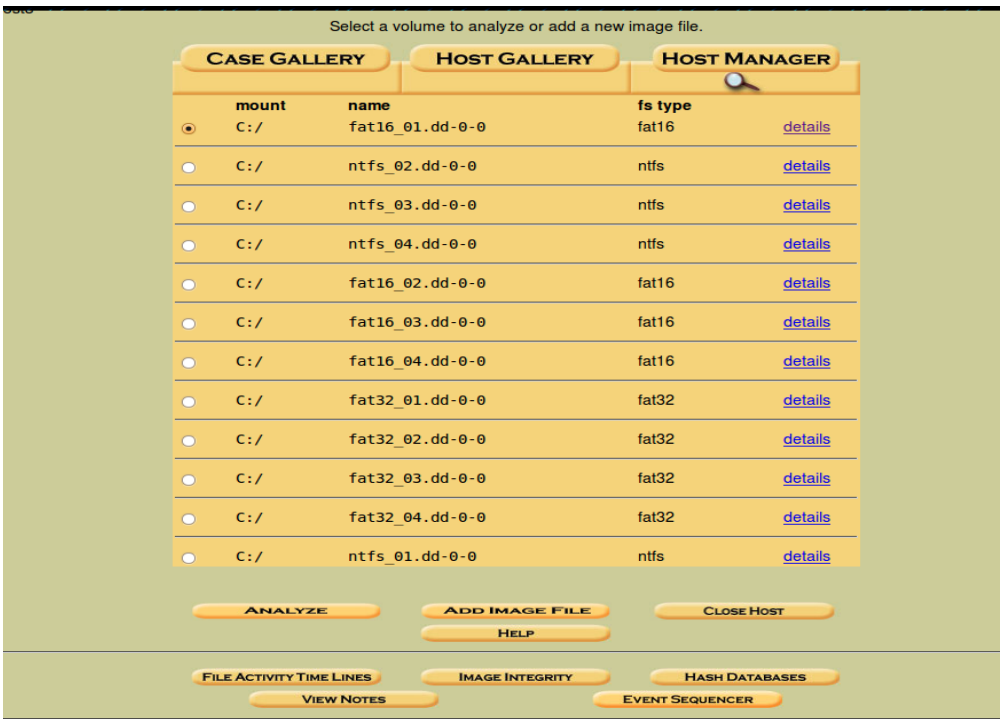

Figura 6 - Tela da ferramenta Autopsy onde são expostas as imagens incluídas

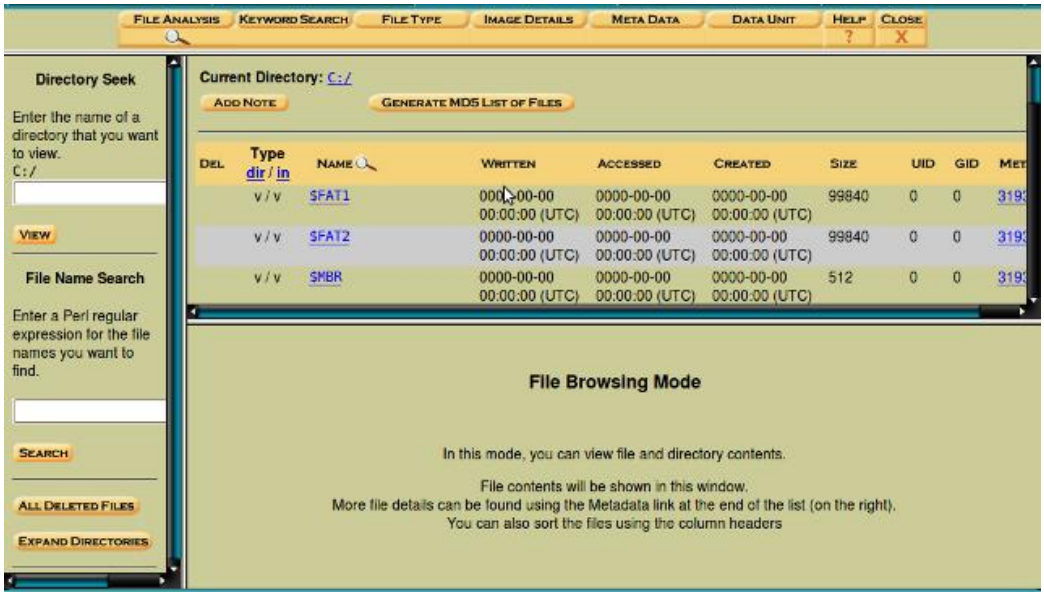

Figura 7 - Página da análise "File Analysis" do Autopsy

Neste tipo de análise pode-se selecionar um arquivo qualquer (clicando nele) e, clicando em *"Export"*, salvá-lo numa pasta qualquer à escolha do investigador. Dependendo dos dados recuperados e do tipo de arquivo, pode-se, inclusive, visualizar o arquivo sem a necessidade de salvá-lo em disco. Deve-se lembrar que tais procedimentos só serão possíveis se o arquivo tiver sido realmente recuperado. Esta ferramenta pode recuperar também apenas fragmentos e/ou os metadados dos arquivos, mas não permitir visualizar o seu conteúdo. Um exemplo de visualização de um arquivo recuperado pode ser visto na Figura 8.

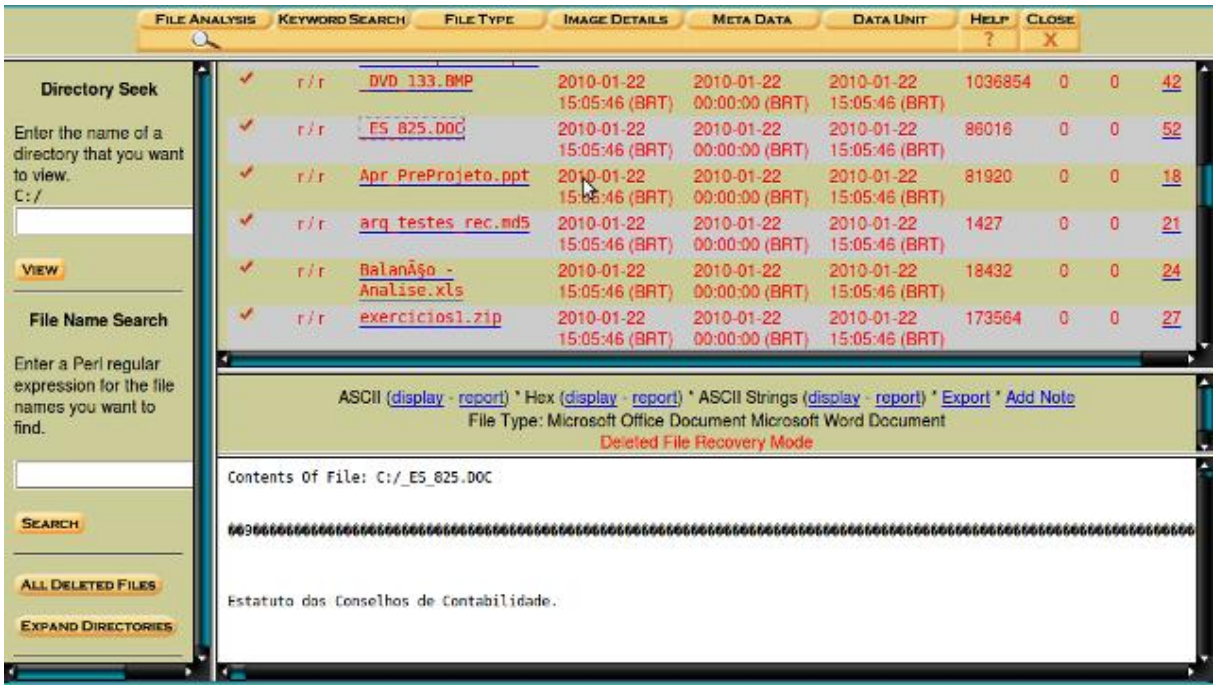

Figura 8 - Visualização de um arquivo recuperado usando a análise "File Analysis"

Também é possível de se obter todos os arquivos possivelmente recuperados ao se utilizar a análise *"File Type"*, cuja tela inicial é mostrada na Figura 9. Clica-se em *"Sort Files by Type"*, faz-se as escolhas necessárias na tela que se segue e, depois de confirmadas as opções, a tela que se vê é a ilustrada na Figura 10. Se a opção de gravar os arquivos em disco foi marcada, todos os arquivos recuperados podem ser conferidos dentro da pasta de análise do Autopsy no diretório explicitado na página mostrada na Figura 10. As imagens podem ser visualizadas pelo navegador mesmo sem terem sido salvas em disco ao seguir o link indicado nesta página de análise do Autopsy.

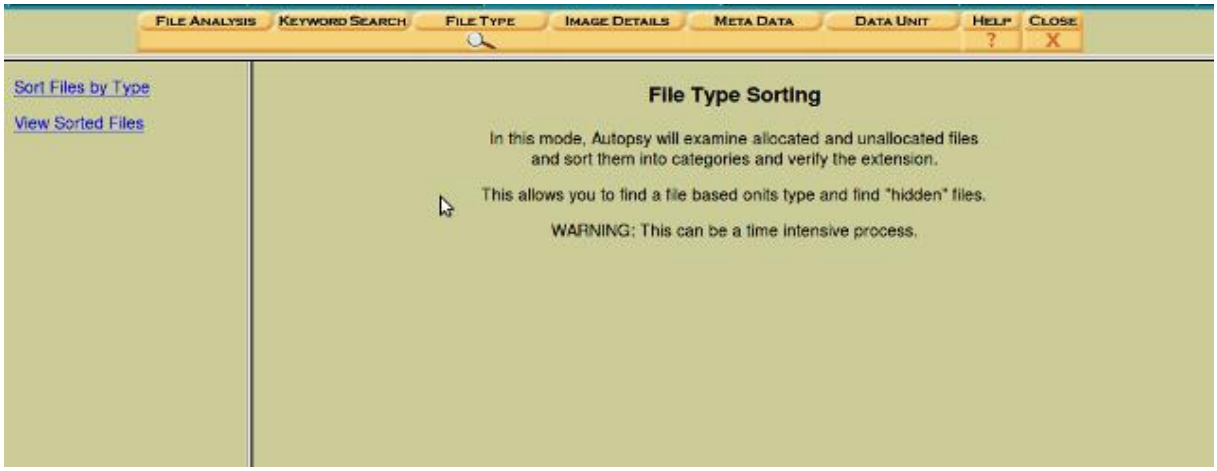

Figura 9 - Tela inicial da análise "File Type" do Autopsy

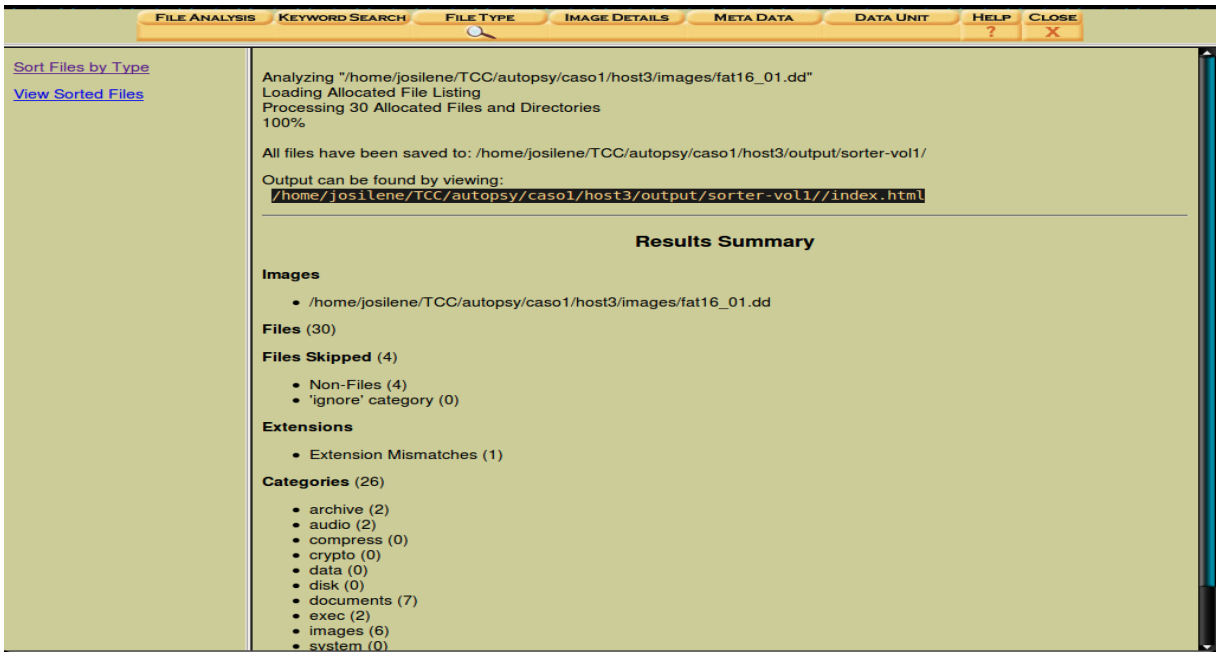

Figura 10 - Resultado obtido com o "File Type" da análise da imagem "fat16_01.dd"

3.3.4 Exame 4 – FTK Imager

Os exames com esta ferramenta seguiram o procedimento que será detalhado a seguir.

A ferramenta é ativada simplesmente com um clique duplo sobre o seu arquivo executável, *"FTK Imager.exe"*. Dentro de sua interface gráfica, para incluir uma nova imagem a ser analisada, basta ir ao botão *"Add Evidence Item"* e dentro da janela que se abre, escolher o item *"Image File"*, clicar em Avançar, escolher a imagem e clicar em concluir.

A imagem é então incluída dentro do programa, que tenta recuperar todos os dados da partição ou disco do qual a imagem foi feita.

3.4 RESULTADOS

Nesta seção serão expostos os resultados dos testes realizados, que serão divididos em cenários para simplificar a explicação e a dedução da conclusão. Cada cenário foi explanado no item 3.2.

Os resultados foram separados em quatro tipos:

- **Total:** quando o seu conteúdo pode ser visto por completo. O resultado foi encaixado nesta categoria mesmo quando as ferramentas não conseguiram recuperar os metadados dos arquivos. As ferramentas Scalpel e Foremost não conseguem recuperar os nomes dos arquivos para nenhum sistema de arquivos e as demais ferramentas também não conseguem recuperar estes metadados em alguns sistemas e cenários.
- **Parcial – Fragmentos de arquivos:** quando o arquivo é recuperado e visualizado em pedaços. Exemplo: parte de uma figura.

- **Parcial – Metadados:** quando apenas os nomes dos arquivos são recuperados e nenhum fragmento do arquivo pode ser visto. Os metadados são todos os atributos dos arquivos diferentes do conteúdo, tais como nome, extensão, tamanho, datas de criação e última alteração. Neste trabalho, foi considerada recuperação de metadados quando pelo menos o nome do arquivo foi recuperado. Nos casos em que as ferramentas puderam recuperar outros metadados dos arquivos, como tamanho e datas de modificação, último acesso e criação, isso será explicitado.

- **Nulo:** quando nada do que é recuperado for válido, ou seja, quando não é possível ser visualizado nenhum fragmento ou metadado dos arquivos.

Seguem observações gerais sobre o comportamento das ferramentas utilizadas:

- O Autopsy não reconheceu o sistema de arquivos ReiserFS;
- O FTK Imager não reconheceu os sistemas ReiserFS e Ext4;
- O Foremost e o Scalpel não recuperam metadados originais.
- São utilizados gráficos para ilustrar os resultados obtidos após a execução dos exames de cada cenário. Nesses gráficos, o número de arquivos recuperados pelas ferramentas é indicado no eixo X e as ferramentas, no eixo Y. As cores das barras representam o tipo de recuperação, conforme informado nesta seção.

3.4.1 Cenário 1

Os resultados para os testes do Cenário 1 estão ilustrados em gráficos divididos por sistema de arquivos.

O desempenho das ferramentas ao tentar recuperar os arquivos descritos na Tabela 2 que foram apagados de partições formatadas com os sistemas Ext2, Ext3, Ext4, ReiserFS, FAT16, FAT32, e NTFS é mostrado, respectivamente, nos Gráficos 1 a 7.

Para o sistema Ext2, mostrado no Gráfico 1, a ferramenta FTK Imager foi a que obteve o melhor desempenho, conseguindo recuperar todos os arquivos. No entanto, deve-se observar que os arquivos foram recuperados sem seus nomes originais ou extensões. Estes dados puderam ser apenas vistos separadamente em outra seção da ferramenta. O Autopsy conseguiu a recuperação apenas dos nomes dos arquivos, enquanto as demais ferramentas apenas recuperaram alguns arquivos ou fragmentos deles.

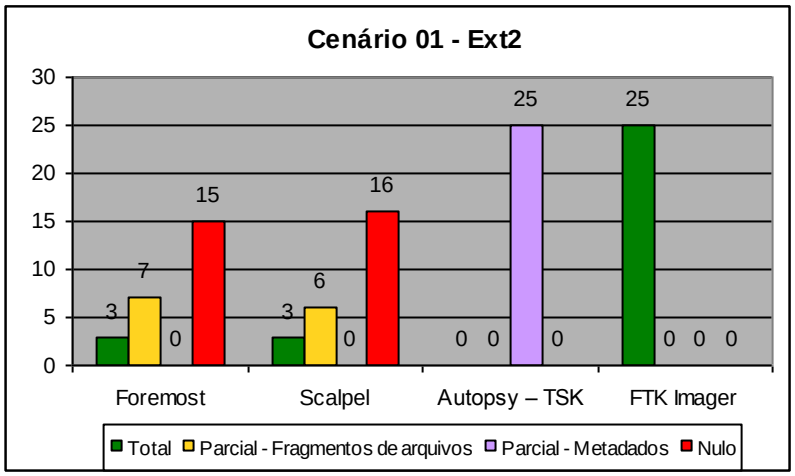

Gráfico 1 – Quantidade de dados recuperados para o Cenário 1, usando o sistema Ext2

Os resultados para o sistema de arquivos Ext3, mostrados no Gráfico 2, são semelhantes aos do Gráfico 1, exceto no tocante à ferramenta FTK Imager, que, desta vez, obteve o mesmo desempenho da Autopsy, ou seja, apenas alguns metadados foram recuperados. Do sistema Ext3 no Cenário 1, verifica-se, portanto, que apenas as ferramentas Foremost e Scalpel foram capazes de recuperar o conteúdo integral ou fragmentos dos arquivos, lembrando que as mesmas não são capazes de recuperar nomes de arquivos.

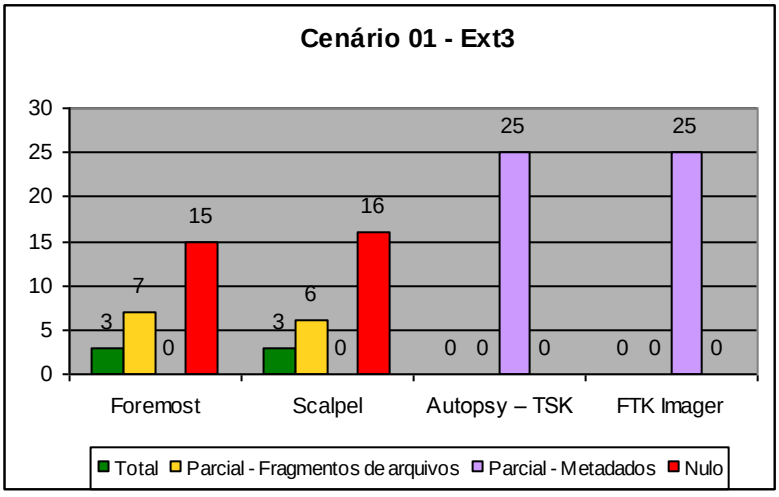

Gráfico 2 – Quantidade de dados recuperados para o Cenário 1, usando o sistema Ext3

As ferramentas Scalpel e Foremost mostram um desempenho melhor ao lidar com imagens do sistema Ext4 no Cenário 1. Entretanto, como não são capazes de recuperar metadados, a melhor abordagem para este caso e para os anteriores é utilizar mais de uma ferramenta. O FTK Imager não reconheceu o sistema Ext4. Estes resultados podem ser melhor visualizados no Gráfico 3.

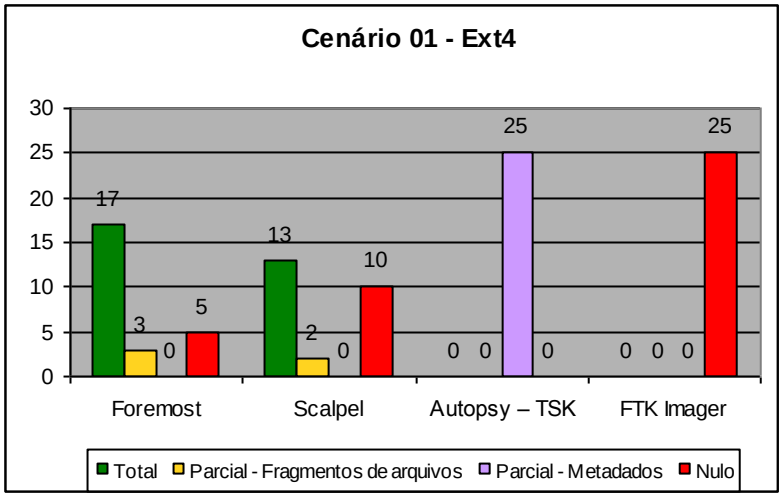

Gráfico 3 – Quantidade de dados recuperados para o Cenário 1, usando o sistema Ext4

As ferramentas FTK Imager e Autopsy não reconheceram o sistema de arquivos ReiserFS. Sendo assim, apenas as ferramentas Scalpel e Foremost tiveram um desempenho aproveitável para este sistema, como visto no Gráfico 4, que se assemelhou ao desempenho demonstrado para o sistema Ext4.

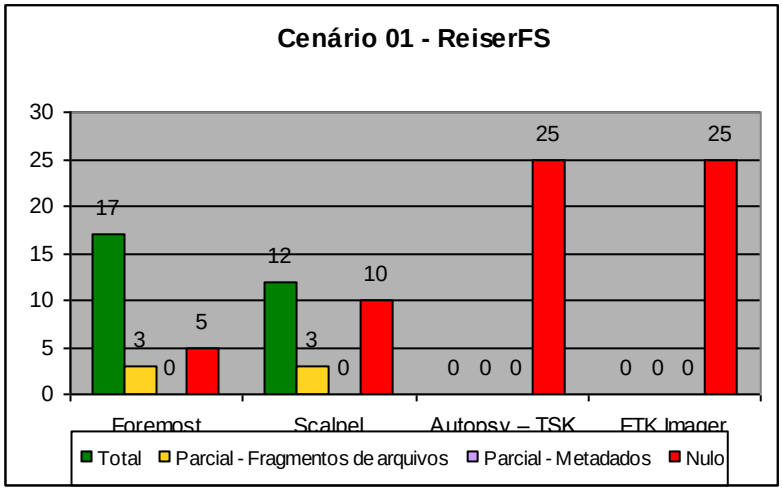

Gráfico 4 – Quantidade de dados recuperados para o Cenário 1 e o sistema ReiserFS

O desempenho das ferramentas sofreu uma melhora significativa, como demonstrado no Gráfico 5, ao analisar uma imagem do sistema FAT16. Para esta imagem, ambas as ferramentas Autopsy e FTK Imager tiveram 100% de aproveitamento.

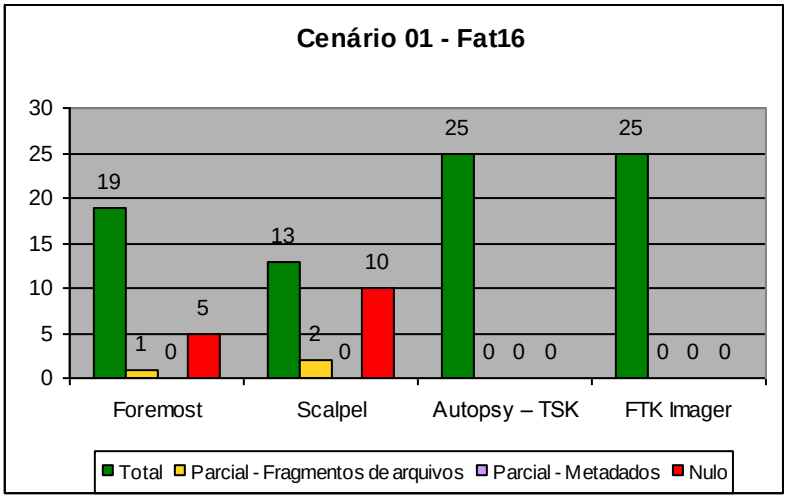

Gráfico 5 – Quantidade de dados recuperados para o Cenário 1, usando o sistema FAT16

O desempenho das ferramentas para o sistema FAT32 foi semelhante ao mostrado para o seu sistema predecessor, como pode ser visto no Gráfico 6.

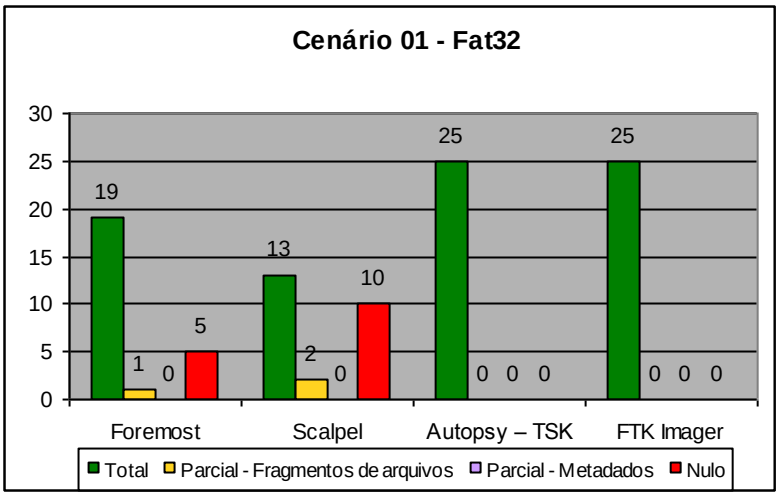

Gráfico 6 – Quantidade de dados recuperados para o Cenário 1, usando o sistema FAT32

Submetidas à imagem do sistema NTFS, nenhuma ferramenta obteve 100% de sucesso, como mostrado no Gráfico 7. Mesmo a Autopsy, ao recuperar quase todos os arquivos da imagem, não conseguiu o mesmo para os seus metadados. As ferramentas Scalpel e Foremost não demonstraram nenhuma mudança em seu comportamento com relação aos

demais sistemas testados. O FTK Imager conseguiu o seu pior desempenho com relação à recuperação de dados neste sistema.

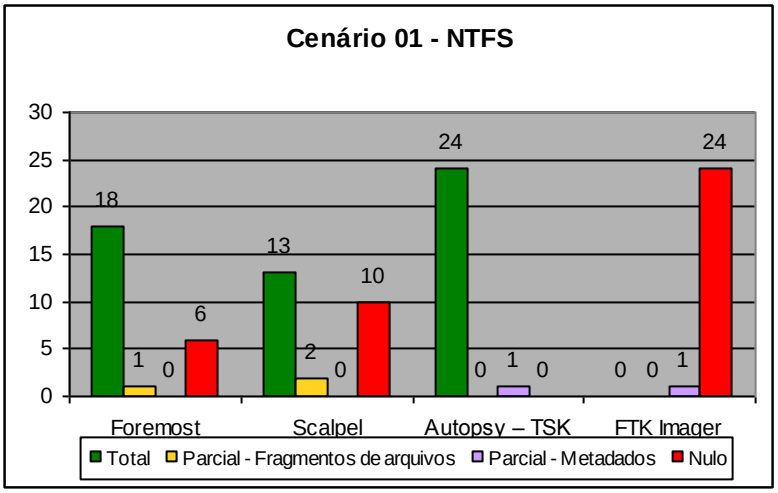

Gráfico 7 – Quantidade de dados recuperados para o Cenário 1, usando o sistema NTFS

As ferramentas Foremost e Scalpel tiveram desempenhos semelhantes, o que se justifica pelo fato da segunda ser baseada na primeira, como explicado na seção 2.3.2, entretanto o Foremost se mostrou ligeiramente superior ao Scalpel para o Cenário1. Como pode ser observado, as ferramentas tendem a recuperar melhor os arquivos quando usadas sobre sistemas Windows, apesar da ferramenta FTK Imager ter tido um desempenho muito bom ao recuperar todo o conteúdo dos arquivos do sistema Ext2.

3.4.2 Cenário 2

Neste cenário, foram copiados arquivos diversos para as partições já usadas antes para os testes do Cenário 1 de forma que ocupassem todo o espaço disponível. O que se espera é medir a capacidade de recuperação de arquivos das ferramentas quando os arquivos a serem recuperados tenham sido sobrescritos por outros dados.

Como pode ser visto nos gráficos desta seção, o desempenho das ferramentas foi quase sempre nulo. As ferramentas FTK Imager e Autopsy tiveram um desempenho semelhante

para todas as imagens, exceto para a Ext4, que não é um sistema reconhecido pelo FTK Imager.

As ferramentas Foremost e Scalpel, para todas as imagens de sistema deste cenário, não conseguiram recuperar nenhum arquivo.

As ferramentas Autopsy e FTK Imager obtiveram resultados semelhantes para as análises dos sistemas Ext2 (Gráfico 8), Ext3(Gráfico 9), FAT16 (Gráfico 10) e FAT32 (Gráfico 11), recuperando quase todos os nomes dos arquivos. O que se conclui é que ambas, mesmo sem ter acesso aos dados sobrescritos, ainda conseguem visualizar algumas informações de quase 100% dos arquivos anteriormente gravados nestas partições.

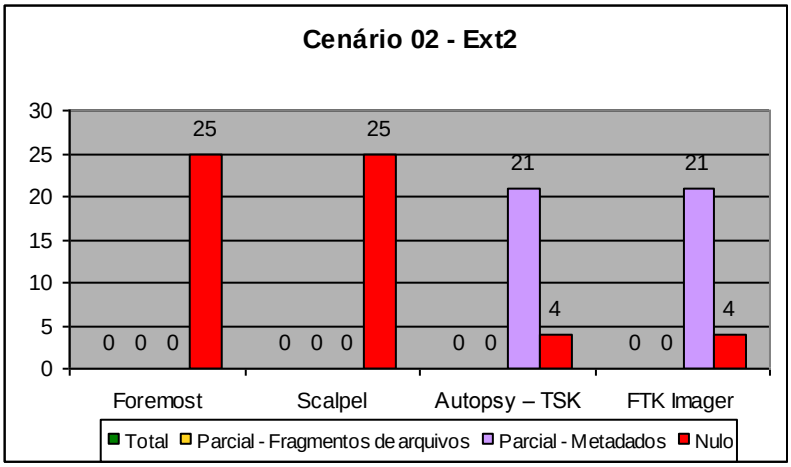

Gráfico 8 – Quantidade de dados recuperados para o Cenário 2, usando o sistema Ext2

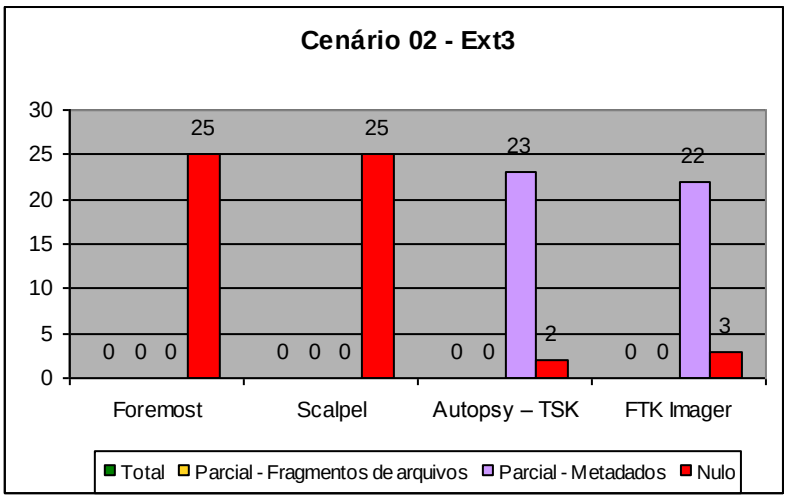

Gráfico 9 – Quantidade de dados recuperados para o Cenário 2, usando o sistema Ext3

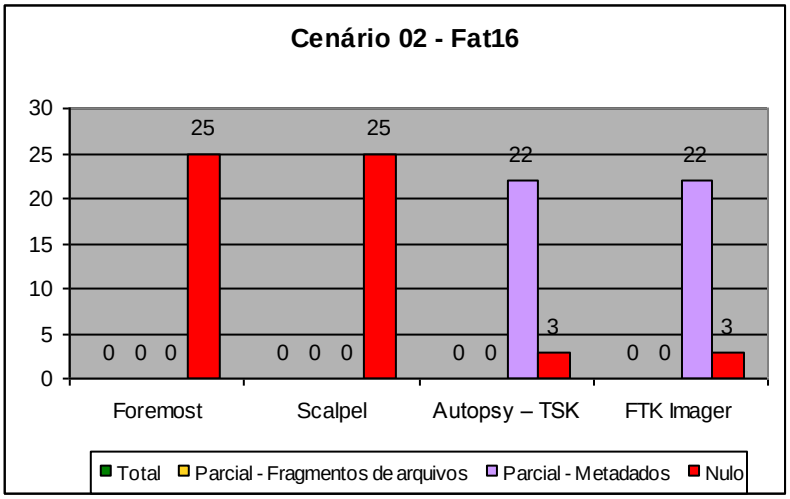

Gráfico 10 – Quantidade de dados recuperados para o Cenário 2, usando o sistema FAT16

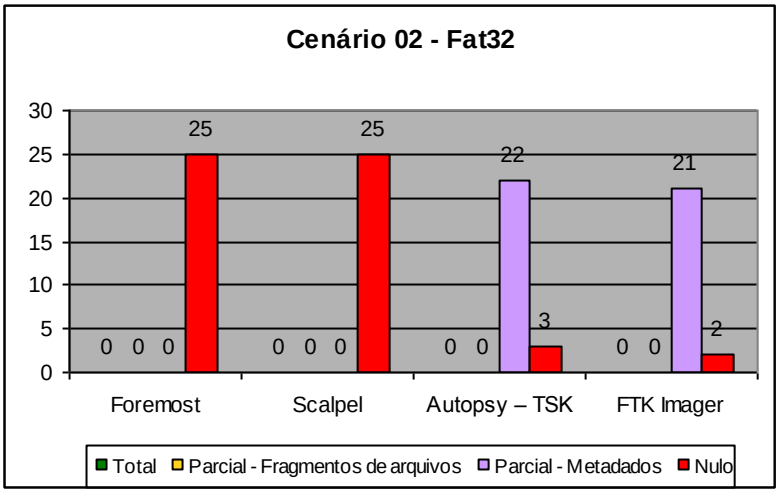

Gráfico 11 – Quantidade de dados recuperados para o Cenário 2, usando o sistema FAT32

O resultado dos testes do Cenário 2 para o sistema Ext4 pode ser visualizado no Gráfico 12. A ferramenta Autopsy obteve um desempenho muito semelhante ao demonstrado para os outros sistemas de mesma natureza, ou seja, Ext2 e Ext3, no mesmo cenário. O FTK Imager não reconhece este sistema, Ext4, nem o ReiserFS, mostrado no Gráfico 13, onde todas as ferramentas obtiveram resultados nulos.

O desempenho das ferramentas sobre um sistema NTFS é mostrado no Gráfico 14. Como pode ser observado, o Autopsy e o FTK Imager conseguem recuperar apenas metadados de um dos arquivos.

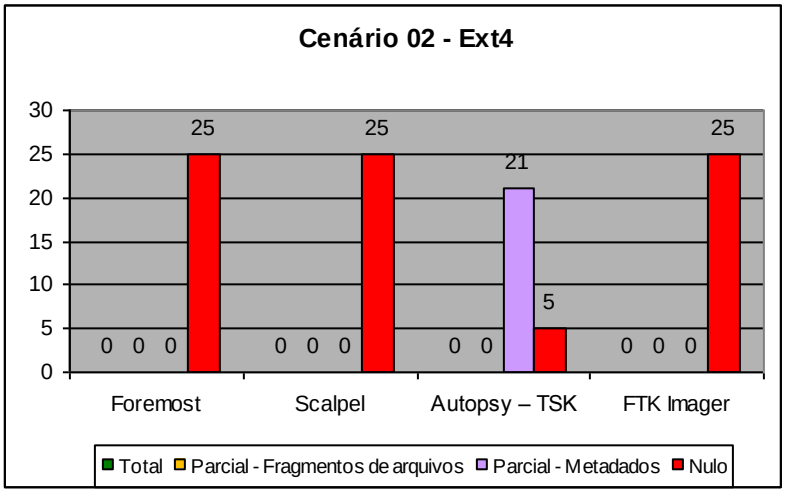

Gráfico 12 – Quantidade de dados recuperados para o Cenário 2, usando o sistema Ext4

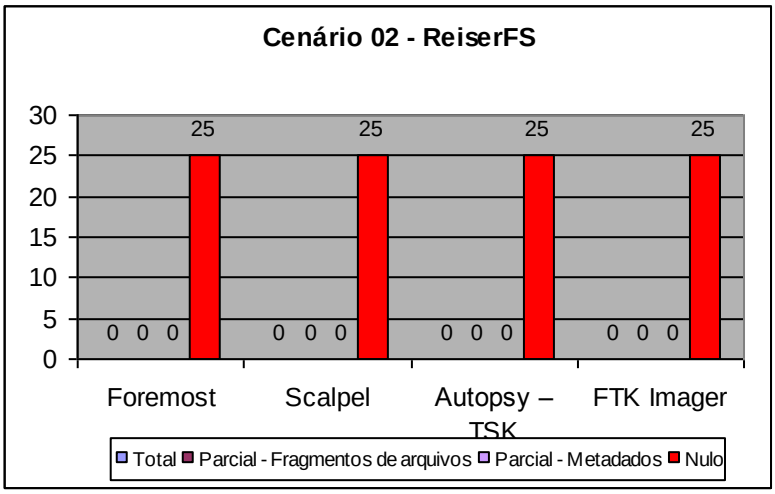

Gráfico 13 – Quantidade de dados recuperados para o Cenário 2 e o sistema ReiserFS

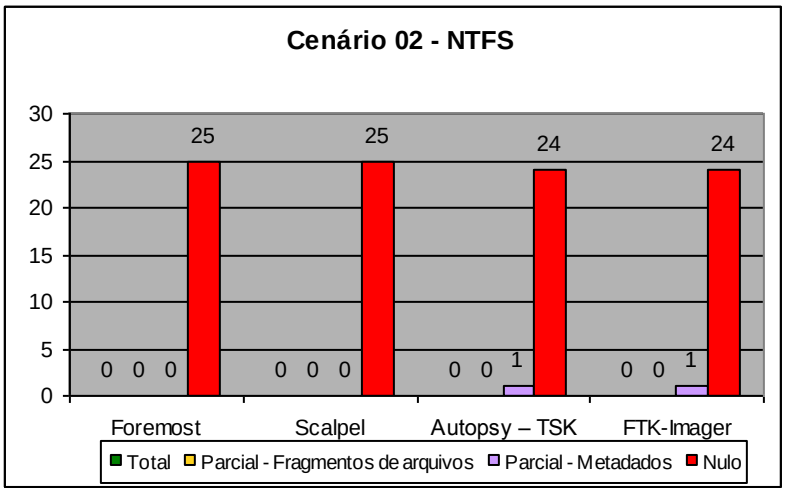

Gráfico 14 – Quantidade de dados recuperados para o Cenário 2, usando o sistema NTFS

Do Cenário 2, de onde os arquivos foram apagados e sobrescritos com dados que preencheram todo o tamanho da partição, esperava-se que o conteúdo de todos os arquivos fosse sobrescritos, como de fato foi comprovado pelos resultados das ferramentas.

Observou-se que duas das ferramentas, Autopsy e FTK Imager ainda conseguem recuperar algumas informações da maioria dos arquivos nos sistemas de arquivos que essas ferramentas reconheciam, com exceção do NTFS onde a recuperação dos metadados foi quase nula. Essa capacidade das ferramentas de ainda recuperarem informações dos arquivos apagados e sobrescritos pode ser explicada pela gravação dos metadados dos novos arquivos não ocupar todo o espaço na tabela de partição antes preenchido pelos metadados dos arquivos apagados. Isso pode ter acontecido porque os arquivos apagados foram sobrescritos por uma quantidade menor de arquivos, com tamanho maior.

3.4.3 Cenário 3

Não houve mudanças nos resultados deste cenário ao compará-los aos do Cenário 1, quando as ferramentas atuaram sobre as partições nos sistemas Ext2 (Gráfico 15), FAT16 (Gráfico 16) e FAT32 (Gráfico 17).

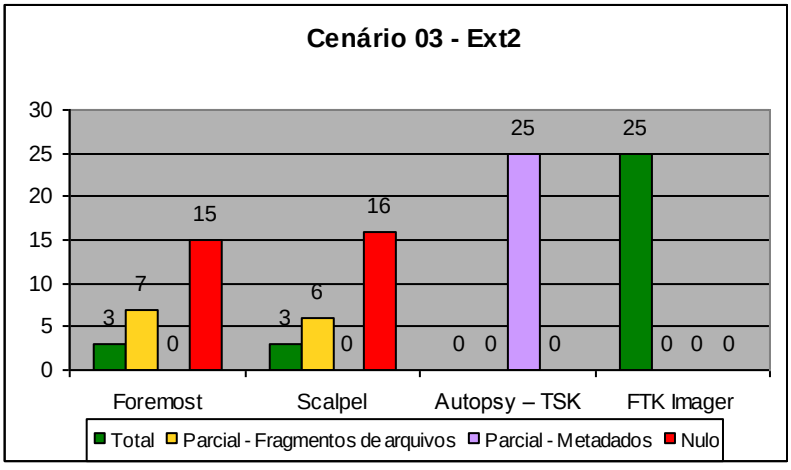

Gráfico 15 – Quantidade de dados recuperados para o Cenário 3, usando o sistema Ext2

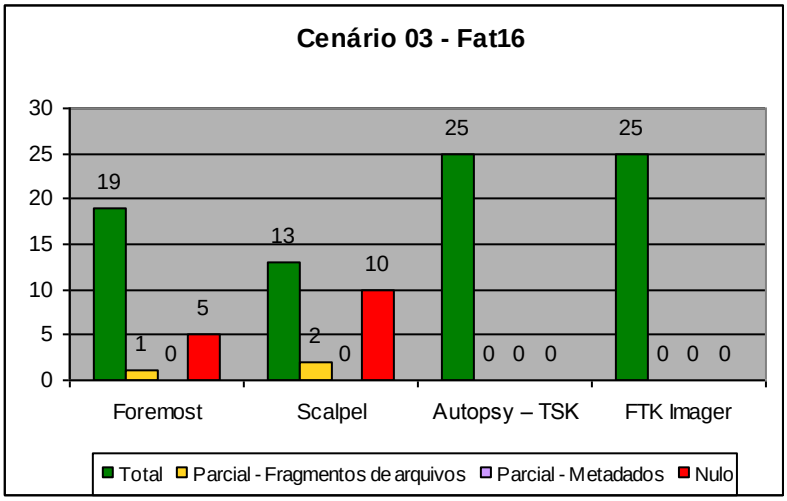

Gráfico 16 – Quantidade de dados recuperados para o Cenário 3, usando o sistema FAT16

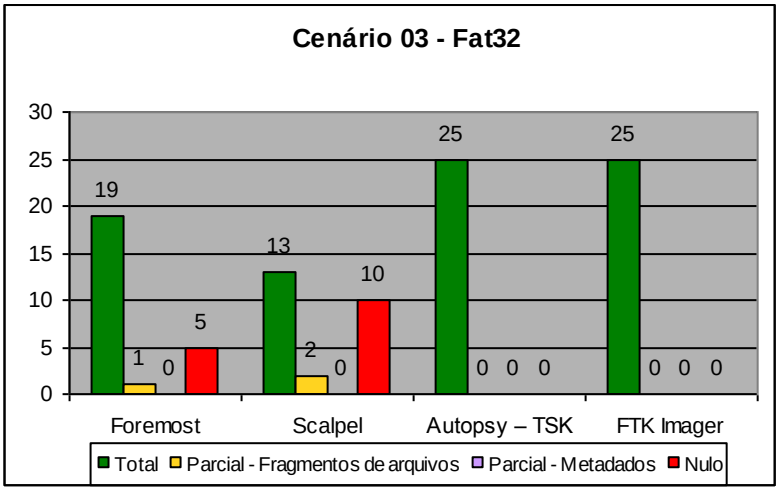

Gráfico 17 – Quantidade de dados recuperados para o Cenário 3, usando o sistema FAT32

No entanto, uma pequena diferença em relação ao Cenário 1 é notada nos resultados das ferramentas Foremost e Scalpel, que obtiveram um ligeiro declínio no desempenho para o sistema de arquivos Ext3 (Gráfico 18) e, por outro lado, obtiveram uma melhora nos resultados para os sistemas Ext4 (Gráfico 19), ReiserFS (Gráfico 20) e NTFS (Gráfico 21), recuperando mais arquivos no Cenário 3 do que no Cenário 1.

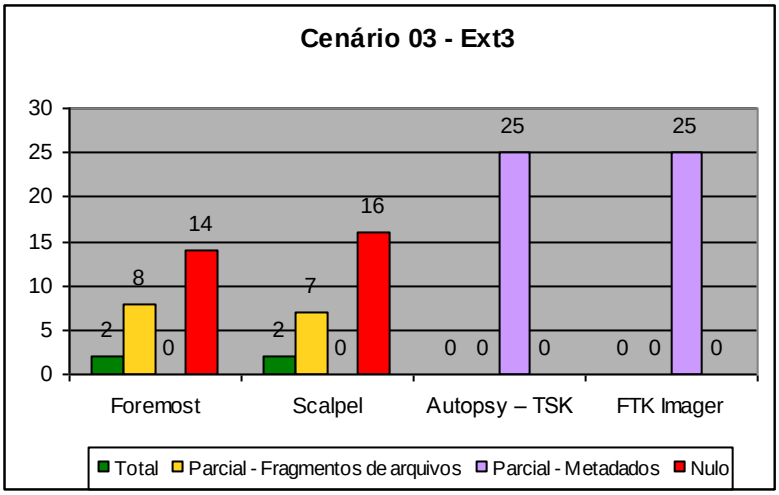

Gráfico 18 – Quantidade de dados recuperados para o Cenário 3, usando o sistema Ext3

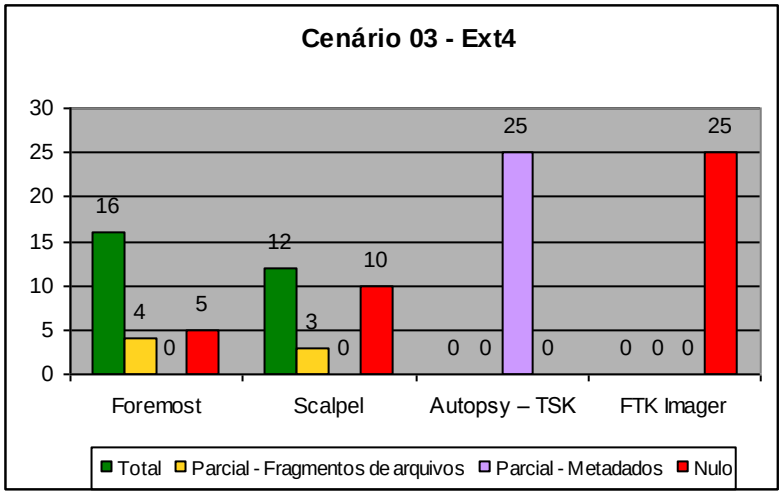

Gráfico 19 – Quantidade de dados recuperados para o Cenário 3, usando o sistema Ext4

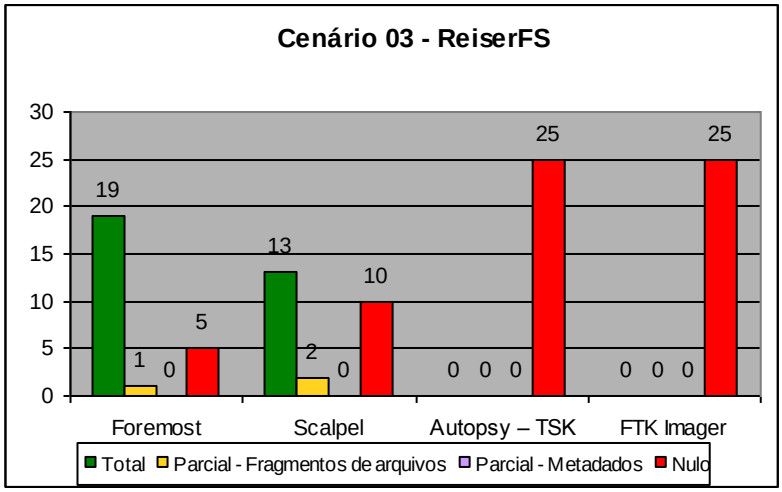

Gráfico 20 – Quantidade de dados recuperados para o Cenário 3 e o sistema ReiserFS

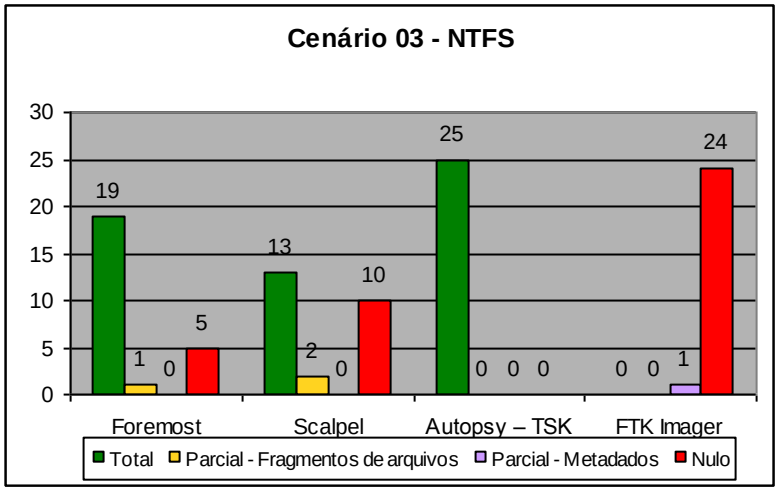

Gráfico 21 – Quantidade de dados recuperados para o Cenário 3, usando o sistema NTFS

Os resultados dos testes para este cenário são semelhantes aos obtidos para o Cenário 1, o que já era esperado, dado que o processo de criação das imagens foi praticamente o mesmo, diferindo apenas no tamanho das partições.

Como para os demais resultados mostrados até agora, o ideal para este cenário é usar mais de uma ferramenta para obter o maior número tanto de conteúdo de arquivos recuperados como de informações sobre os arquivos (metadados). Excetuem-se, naturalmente, os casos em que uma só ferramenta consegue recuperar todos os dados como ocorre com o Autopsy (sistemas FAT16, FAT32 e NTFS) e FTK Imager (Ext2, FAT16 e FAT32) para o Cenário 3.

3.4.4 Cenário 4

Neste cenário, foram usadas as mesmas imagens do Cenário 3, onde após os arquivos serem apagados, apenas um arquivo de 71MB foi copiado para sobrescrever a partição. Como já explicado na seção 3.2.3, a intenção foi deixar espaço suficiente para que, dependendo do modo de operar do sistema de arquivos usado, os arquivos a serem recuperados não sejam

todos sobrescritos. Aqui também pode ser observado o comportamento dos sistemas de arquivos.

O desempenho das ferramentas frente a uma imagem do sistema Ext2 submetida ao Cenário 4 é mostrado no Gráfico 22. Comparando os resultados com os do Cenário 2, a mudança foi mínima. As ferramentas Autopsy e FTK Imager apenas conseguiram recuperar mais alguns metadados (nomes) dos arquivos, o que mostra que estes foram muito provavelmente sobrescritos.

O mesmo comportamento foi verificado nos resultados para o sistema Ext3, ilustrado no Gráfico 23.

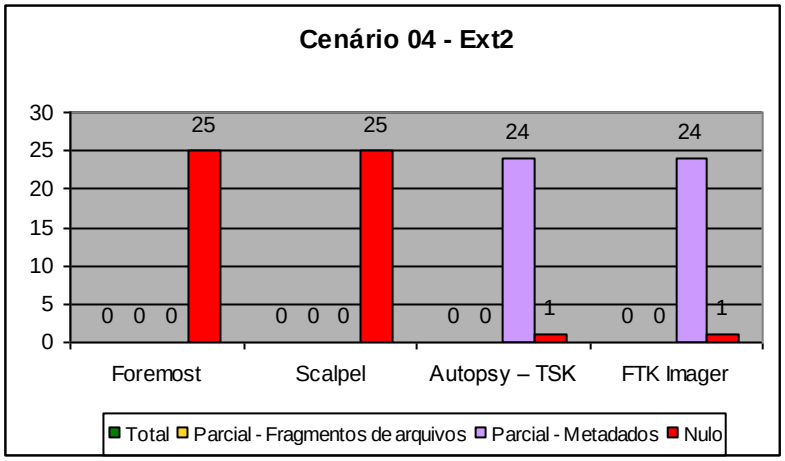

Gráfico 22 – Quantidade de dados recuperados para o Cenário 4, usando o sistema Ext2

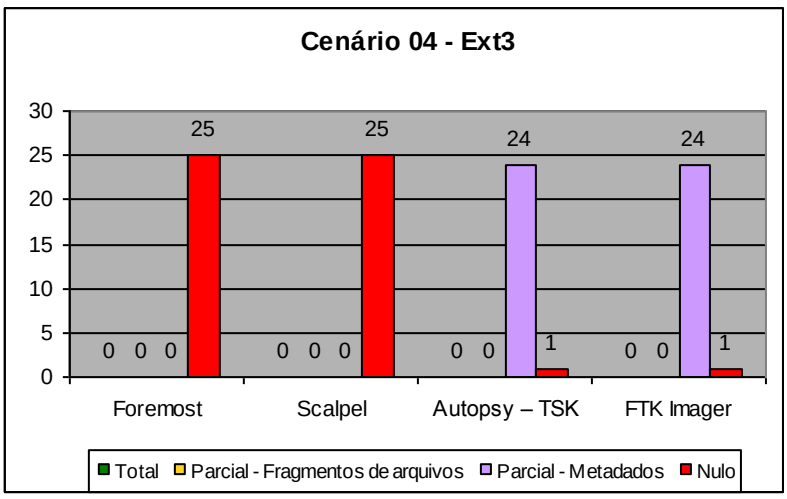

Gráfico 23 – Quantidade de dados recuperados para o Cenário 4, usando o sistema Ext3

Com relação ao sistema Ext4, mostrado no Gráfico 24, pode-se notar uma melhora significativa nos resultados das ferramentas Foremost e Scalpel em comparação com o

Cenário 2. Como aconselhado para os Cenários 1 e 3, onde uma ferramenta recupera os metadados e outra, os arquivos, a abordagem recomendada é usar mais de uma ferramenta e tentar fazer associações entre metadados (nome, extensão, tamanho, datas) e conteúdo.

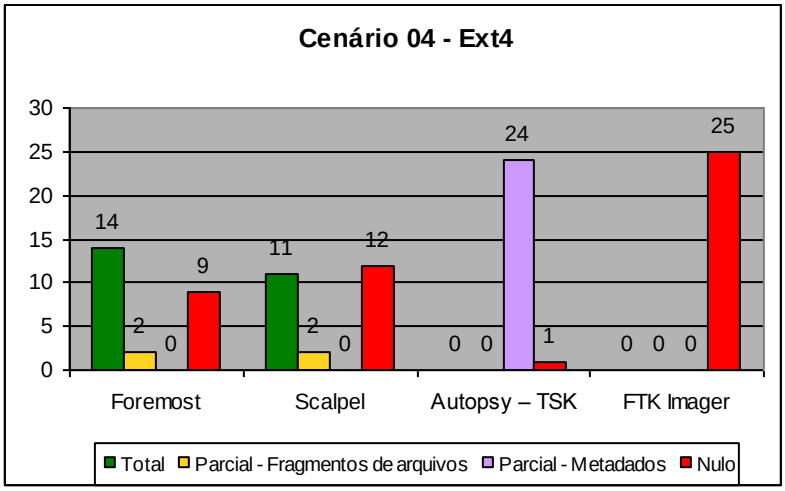

Gráfico 24 – Quantidade de dados recuperados para o Cenário 4, usando o sistema Ext4

Diante da análise da imagem do sistema ReiserFS, nenhuma ferramenta conseguiu recuperar nem metadados nem conteúdo dos arquivos, como pode ser visualizado no Gráfico 25.

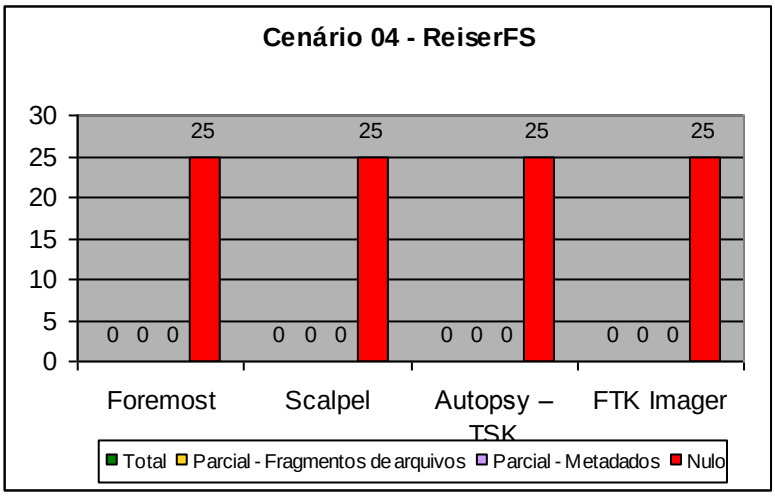

Gráfico 25 – Quantidade de dados recuperados para o Cenário 4 e o sistema ReiserFS

O sistema FAT16 mostra um comportamento, como visto no Gráfico 26, que se assemelha ao dos sistemas Ext2 e Ext3 no que diz respeito à recuperação de dados. Assim como para aqueles, apenas as ferramentas Autopsy e FTK Imager conseguiram recuperar os metadados (nomes) dos arquivos. Não houve recuperação de conteúdo.

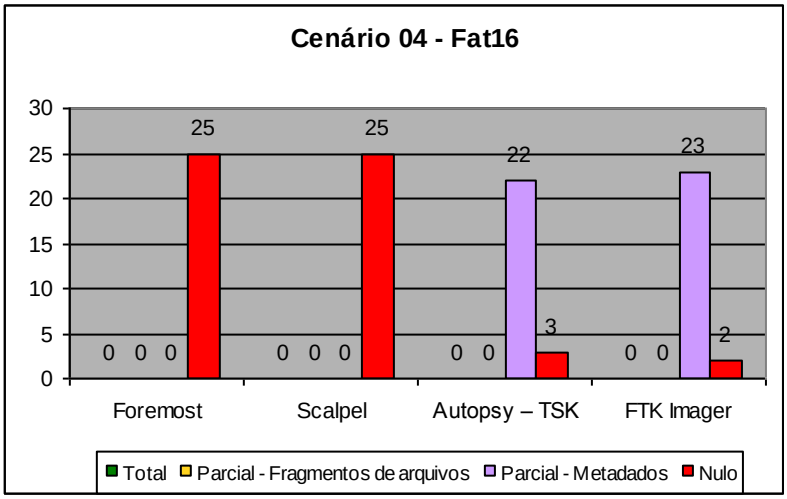

Gráfico 26 – Quantidade de dados recuperados para o Cenário 4, usando o sistema FAT16

Uma melhora bem mais significativa para o sistema FAT32, comparando-se com os resultados do Cenário 2, é mostrada no Gráfico 27. Quase todos os arquivos foram recuperados tanto em conteúdo como em metadados, sendo que as ferramentas Autopsy e FTK Imager tiveram um melhor desempenho com relação às Foremost e Scalpel, lembrando que estas últimas não se mostram capazes de recuperar metadados.

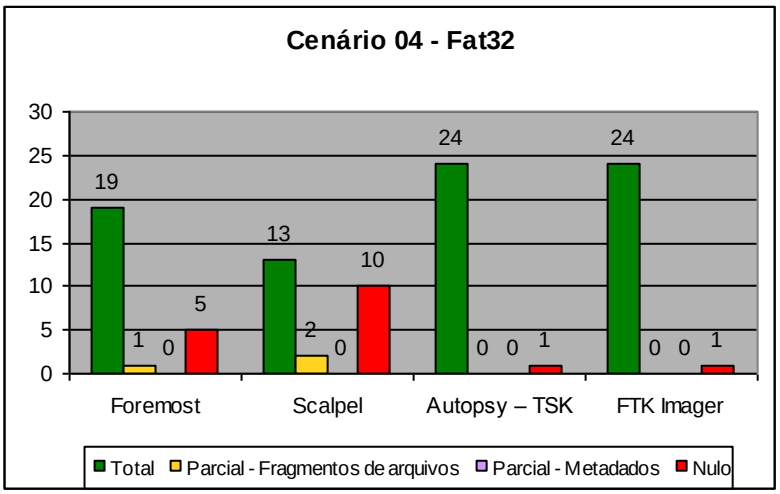

Gráfico 27 – Quantidade de dados recuperados para o Cenário 4, usando o sistema FAT32

Os resultados para a imagem do sistema NTFS, mostrados no Gráfico 28, são bastante singulares, não deixando que sejam comparados com as demais imagens analisadas. A ferramenta Foremost foi a que obteve um melhor desempenho neste sistema. Menos da metade dos arquivos foram recuperados pelas ferramentas Foremost, Scalpel e Autopsy. O FTK Imager teve o desempenho menos satisfatório e conseguiu recuperar apenas o nome de um dos arquivos. O Autopsy recuperou os metadados (nome, tamanho, data de modificação)

de apenas um arquivo que não estava entre os cinco que tiveram seu conteúdo recuperado por esta ferramenta.

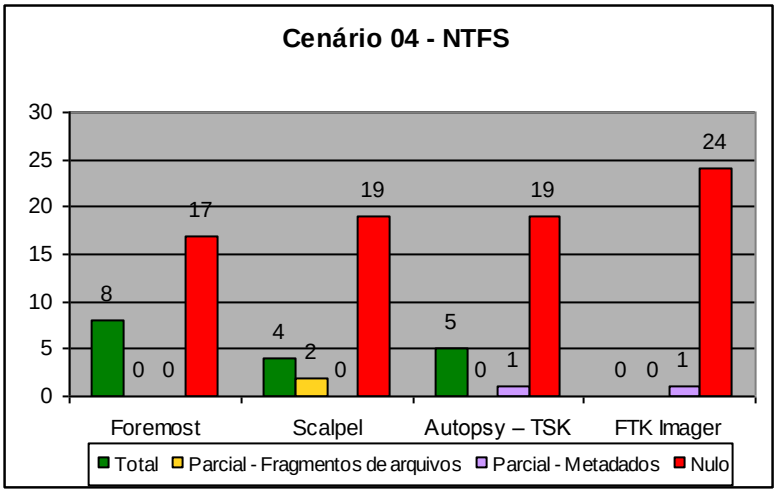

Gráfico 28 – Quantidade de dados recuperados para o Cenário 4, usando o sistema NTFS

Para o Cenário 4, a abordagem de usar mais de uma ferramenta para otimizar os resultados funcionaria apenas no que diz respeito ao sistema Ext4. Pra os demais sistemas, utilizar a ferramenta que obteve o melhor desempenho já é o suficiente para que o melhor resultado seja alcançado.

Ao comparar os resultados do Cenário 4 e os do Cenário 2, observam-se melhoras para o sistema Ext4 ao usar as ferramentas Scalpel ou Foremost, dado que desta vez, cerca de metade dos arquivos foram recuperados. O FAT32 obteve melhoras muito mais significativas com relação à recuperação de dados quando todas as ferramentas conseguiram recuperar o conteúdo de diversos arquivos, com destaque para as ferramentas Autopsy e FTK Imager, que conseguiram recuperar conteúdo e metadados de quase todos os arquivos. Os resultados para o sistema NTFS também obtiveram melhoras, pois, apesar da baixa recuperação demonstrada pelas ferramentas no Cenário 4, as recuperações do Cenário 2 foram nulas para todas elas.

Com esta comparação entre cenários e sistemas de arquivos, pode-se concluir que o comportamento das ferramentas é alterado de acordo com o sistema de alocação de dados usado pelos sistemas de arquivos.

A seção 3.4.5 traz um resumo dos resultados obtidos pelas ferramentas forenses para cada sistema de arquivos.

3.4.5 Visão geral dos resultados

Revendo os resultados de todas as imagens para todos os cenários, podemos ter uma idéia de que abordagem usar para cada sistema de arquivos a ser analisado. Um resumo de que procedimento é aconselhado se adotar é apresentado a seguir.

Sistema Ext2: O melhor desempenho foi da ferramenta FTK Imager, que recupera conteúdo e metadados, apesar de separadamente. Com relação às ferramentas *opensource*, a melhor abordagem é usar mais de uma ferramenta para poder recuperar dados e metadados e, ainda assim, poucos arquivos têm tanto conteúdo como metadados recuperados.

Sistema Ext3: As ferramentas Autopsy e FTK Imager obtiveram o mesmo desempenho independentemente do cenário, ou seja, apenas recuperaram metadados. Assim, a melhor abordagem para este sistema é usar mais de uma ferramenta, incluindo também o Foremost ou Scalpel, e buscar fazer associações entre nomes e conteúdos. Mesmo assim, poucos arquivos são recuperados.

Sistema Ext4: Apesar da ferramenta FTK Imager não reconhecer este sistema e o Autopsy recuperar apenas metadados, as Foremost e Scalpel conseguiram recuperar uma quantidade maior de arquivos ao comparar estes resultados com os dos outros sistemas Linux. Logo essas duas ferramentas, Foremost e Scalpel, são recomendadas quando se tratar de recuperar arquivos no sistema Ext4.

Sistema ReiserFS: As ferramentas Autopsy e FTK Imager não funcionaram para este sistema em nenhum cenário e as ferramentas Foremost e Scalpel conseguiram recuperar o conteúdo de alguns arquivos apenas nos Cenários 1 e 3, ou seja, quando não havia dados sobrescritos.

Sistema FAT16: As ferramentas Autopsy e FTK Imager conseguiram recuperar todos os dados para os Cenários 1 e 3 e as Scalpel e Foremost, alguns deles. No entanto, para os demais Cenários 2 e 4, apenas o Autopsy e o FTK Imager conseguiram recuperar os metadados de alguns arquivos e as outras ferramentas efetuaram nenhuma recuperação.

Sistema FAT32: Quando os dados são totalmente sobrescritos, apenas metadados são recuperados usando-se as ferramentas Autopsy e FTK Imager. Para os demais cenários, a

recuperação é quase sempre total usando uma ou outra ferramenta, sendo recomendado utilizar o Autopsy ou o FTK Imager, dado a maior quantidade de arquivos recuperados e o fato de que ambas recuperam também as informações relacionadas a eles.

Sistema NTFS: Nenhuma ferramenta conseguiu ter sucesso ao recuperar metadados. Apenas um arquivo teve os seus metadados recuperados para todos os cenários. Com relação ao conteúdo dos arquivos, estes são recuperados quase totalmente quando não há indício de terem sido sobrescritos. Quando totalmente sobrescritos, nada é recuperado e com os dados parcialmente sobrescritos, a ferramenta Foremost foi a que apresentou o melhor desempenho ao recuperar a maior quantidade de arquivos. A ferramenta FTK Imager, para todos os cenários, conseguiu recuperar apenas os metadados de um arquivo, sem recuperar o conteúdo de nenhum deles.

CONCLUSÃO

Com o aumento da capacidade de armazenamento dos equipamentos eletrônicos, a diminuição de seus preços, junto à massificação de seu uso, a probabilidade de serem perdidos dados importantes por simples descuido cresce na mesma proporção. Pelo mesmo motivo, os crimes usando estes equipamentos também aumentam a cada dia.

Pensando nisso, diversas ferramentas são desenvolvidas com o intuito de recuperar estes dados que, apesar de apagados do disco, ou mesmo por este motivo, podem vir a ser importantes. Neste trabalho, foi testada uma parcela destas ferramentas.

Ao submeter estas ferramentas a quatro cenários diferentes de perda de dados, pode-se concluir que é relativamente simples recuperar dados que tenham sido apagados imediatamente do disco. No entanto, à medida que o sistema é usado e novos dados são inseridos nele, fica cada vez mais difícil fazer esta recuperação de forma simples e medidas mais sofisticadas devem ser tomadas para chegar a este objetivo.

Como apresentado neste trabalho, nenhuma ferramenta é totalmente eficiente para todos os sistemas de arquivos ou cenários em que os seus dados possam estar inseridos. A melhor abordagem é, antes de iniciar o procedimento de recuperação, observar qual o sistema de arquivos em que a partição analisada é formatada e verificar se o sistema tinha sido muito utilizado desde que o arquivo procurado foi apagado do disco.

Para os sistemas de arquivos mais comuns nos sistemas operacionais do Windows, as ferramentas que apresentaram melhores resultados na recuperação de conteúdo de arquivos apagados foram o Autopsy e a FTK Imager. O Autopsy alcançou algum resultado positivo para todos os sistemas Windows em todos os cenários. O FTK Imager se igualou ao Autopsy em todos os resultados com exceção da análise de um sistema NTFS para o Cenário 4, onde o FTK Imager não recuperou arquivo algum. Excepcionalmente, para este último resultado a melhor ferramenta foi a Foremost.

Para os sistemas de arquivos mais utilizados em sistemas operacionais Linux, quando há alguma recuperação de conteúdo dos arquivos, as ferramentas que se mostraram melhores foram o Foremost e o Scalpel. Para estes casos, recomenda-se o uso adicional da ferramenta

Autopsy para tentar recuperar também os nomes dos arquivos e correlacionar nomes com conteúdo. A exceção neste caso fica por conta do desempenho do FTK Imager para o sistema Ext2 nos Cenários 1 e 3, quando a ferramenta conseguiu recuperar todos os arquivos com conteúdo e nome, apesar de fazê-lo em pastas diferentes de sua interface gráfica.

Outra conclusão que pode ser retirada é o quão difícil é apagar definitivamente um dado de um disco. Informações confidenciais existem e discos ultrapassados têm seu uso suspenso diariamente em grandes empresas. Para que estes dados tenham sua confidencialidade respeitada, procedimentos cautelosos devem ser tomados para que estes discos sejam descartados com segurança e informações sensíveis não corram o risco de serem divulgadas sem a devida autorização de seus responsáveis.

Como trabalhos e estudos futuros são indicados:

- A inclusão de mais ferramentas de recuperação de dados, tanto proprietárias, como de código aberto;
- Testar o desempenho das ferramentas forenses por tipo de arquivos apagados, como multimídia, documentos, planilhas, entre outros;
- O estudo mais aprofundado sobre o comportamento de sistemas operacionais no que diz respeito à gravação e sobrescrição de dados em disco.

REFERÊNCIAS BIBLIOGRÁFICAS

ACCESSDATA Product Downloads. Disponível em: <http://www.accessdata.com/downloads.html >. Acesso em: 2 mar. 2010.

CARRIER, Brian. **File System Forensic Analysis.** Addison-Wesley, 2005.

CHISUM, W.J.; Turvey, B. Evidence Dynamics: Locard's Exchange Principle & Crime Reconstruction. **Journal of Behavioral Profiling**, v1, n 1, jan. 2000.

COURTNEY, Scott. **An In-Depth Look at Reiserfs**. Disponível em: <http://www.linuxplanet.com/linuxplanet/tutorials/2926/4/>. Acesso em: 26 fev. 2010.

DFLABS. **PTK an alternative computer forensic framework**. Disponível em: <http://ptk.dflabs.com/>. Acesso em: 2 mar. 2010.

FOREMOST. Disponível em: <http://foremost.sourceforge.net/>. Acesso em: 2 mar. 2010.

GIL, A. C. **Como elaborar projetos de pesquisa**. 4a. ed. São Paulo: Atlas, 2002.

IEONG, Ricci S.C. FORZA – Digital forensics investigation framework that incorporate legal issues. **Digital Investigation**, Amsterdam, v. 3, s.1, p. 29-36, set. 2006.

JONES, M. Tim. **Anatomia do Sistema de Arquivos do Linux**. Disponível em: <http://www.ibm.com/developerworks/br/library/l-linux-filesystem/index.html>. Acesso em: 30 jan. 2010.

JONES, Tim M. **Anatomia do Ext4**. Disponível em: <http://www.ibm.com/developerworks/br/library/l-anatomy-ext4/index.html>. Acesso em: 10 fev. 2010.

MELO, Sandro. **Computação Forense com Software Livre**: Conceitos, Técnicas, Ferramentas e Estudos de Casos. 1a ed. Rio de Janeiro: Alta Books, 2009.

MORIMOTO, Carlos E. **Hardware, o Guia Definitivo**. Disponível em: <http://www.gdhpress.com.br/hardware/>. Acesso em: 3 fev. 2010.

MOURA, M. **hard_13hd4.png**. Curso de Administração – Computação I. 2009. Disponível em: <http://computacao.web44.net/adm_comp/tutoriais/hard3.html>. Acesso em: 10 fev. 2010.

SCALPEL: A Frugal, High Performance File Carver. Disponível em: <http://www.digitalforensicssolutions.com/Scalpel/>. Acesso em: 2 mar. 2010.

THE SLEUTH KIT (TSK) & Autopsy: Open Source Digital Investigation Tools. Disponível em: <http://www.sleuthkit.org/>. Acesso em: 2 mar. 2010.

VACCA, John R. **Computer Forensics**: Computer Crime Scene Investigation. 2a ed. Boston: Charles River Media, 2005.