UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

PROPOSTA DE MÉTODO PARA AUMENTO DE DESEMPENHO NA DUPLICAÇÃO FORENSE DE DISCOS RÍGIDOS

DANIEL JOSÉ SALOMONI

ORIENTADOR: RAFAEL TIMÓTEO DE SOUSA JÚNIOR

DISSERTAÇÃO DE MESTRADO EM ENGENHARIA ELÉTRICA

PUBLICAÇÃO: PPGENE.MD - 105 A/12

BRASÍLIA / DF: 07/2012

UNIVERSIDADE DE BRASÍLIA
FACULDADE DE TECNOLOGIA
DEPARTAMENTO DE ENGENHARIA ELÉTRICA

PROPOSTA DE MÉTODO PARA AUMENTO DE DESEMPENHO NA DUPLICAÇÃO FORENSE DE DISCOS RÍGIDOS

DANIEL JOSÉ SALOMONI

DISSERTAÇÃO DE MESTRADO PROFISSIONALIZANTE SUBMETIDA AO DEPARTAMENTO DE ENGENHARIA ELÉTRICA DA FACULDADE DE TECNOLOGIA DA UNIVERSIDADE DE BRASÍLIA, COMO PARTE DOS REQUISITOS NECESSÁRIOS PARA A OBTENÇÃO DO GRAU DE MESTRE.

APROVADA POR:

RAFAEL TIMÓTEO DE SOUSA JÚNIOR, Doutor, UnB
(ORIENTADOR)

FLAVIO ELIAS GOMES DE DEUS, Doutor, UnB
(EXAMINADOR INTERNO)

ROBSON DE OLIVEIRA ALBUQUERQUE, Doutor, ABIN
(EXAMINADOR EXTERNO)

DATA: BRASÍLIA/DF, 02 DE JULHO DE 2012.

FICHA CATALOGRÁFICA

SALOMONI, DANIEL JOSÉ
Proposta de Método para Aumento De Desempenho na Duplicação Forense de Discos Rígidos [Distrito Federal] 2012.
xviii, 139p., 297 mm (ENE/FT/UnB, Mestre, Engenharia Elétrica, 2012).

Dissertação de Mestrado – Universidade de Brasília, Faculdade de Tecnologia. Departamento de Engenharia Elétrica.

1. Discos Rígidos 2. Informática Forense
3. Duplicação forense 4. Metodologia

I. ENE/FT/UnB. II. Título (Série)

REFERÊNCIA BIBLIOGRÁFICA

SALOMONI, D. J. (2012). Proposta de Método para Aumento de Desempenho na Duplicação Forense de Discos Rígidos. Dissertação de Mestrado, Publicação PPGENE.DM XX A/12, Departamento de Engenharia Elétrica, Universidade de Brasília, Brasília, DF, 139p.

CESSÃO DE DIREITOS

NOME DO AUTOR: Daniel José Salomoni
TÍTULO DA DISSERTAÇÃO: Proposta de Método para Aumento de Desempenho na Duplicação Forense de Discos Rígidos.
GRAU/ANO: Mestre/2012.

Daniel José Salomoni
Universidade de Brasília
Faculdade de Tecnologia
Departamento de Engenharia Elétrica
CEP 70910-900
Brasília – DF – Brasil

Dedicatória

À minha esposa Daniela, por seu estímulo, suporte e paciência.

À minha filha Rafaela, que simplesmente por existir faz com que eu busque o melhor em mim.

Aos meus pais, irmãos e amigos, que também são vencedores com a conquista de mais esta etapa.

AGRADECIMENTOS

Ao meu orientador Prof. Dr. Rafael Timóteo de Sousa Júnior, pelo constante apoio, incentivo, dedicação e amizade essenciais para o desenvolvimento deste trabalho e para o meu desenvolvimento como pesquisador.

Ao Doutor Robson de Oliveira Albuquerque, da Agência Brasileira de Inteligência, pela ajuda em diversos aspectos deste trabalho.

Aos funcionários do FT-ENE pela colaboração e amizade.

A todos, os meus sinceros agradecimentos.

O presente trabalho foi realizado com o apoio do Departamento Polícia Federal – DPF e do Instituto-Geral de Perícias do Rio Grande do Sul (IGP-RS), com recursos do Programa Nacional de Segurança Pública com Cidadania – PRONASCI, do Ministério da Justiça.

RESUMO

PROPOSTA DE MÉTODO PARA AUMENTO DE DESEMPENHO NA DUPLICAÇÃO FORENSE DE DISCOS RÍGIDOS

Autor: Daniel José Salomoni
Orientador: Rafael Timóteo de Sousa Júnior
Programa de Pós-graduação em Engenharia Elétrica
Brasília, Julho de 2012

O crescimento no tamanho dos discos rígidos demanda um tempo cada vez maior na análise forense dos dados. Uma das etapas da análise forense é a de preservação da evidência digital, sendo que a duplicação forense é a técnica usada para garantir essa preservação. Este trabalho objetiva verificar se o desempenho do processo de duplicação forense pode ser otimizado, com o uso de um método que forneça o cenário de melhor desempenho, baseado no desempenho de ferramentas conhecidas. Os cenários testados incluem a variação do sistema de arquivo destino, do tamanho de setores por bloco e da natureza dos dados de origem. Foram utilizadas as ferramentas LinEn, FTKImager e EWFAcquire, para gravação de imagens no formato E01, e as ferramentas DC3DD e DCFLDD, para o formato RAW. Os resultados mostram que dados de origem muito heterogêneos e o número de setores por bloco não influenciam no desempenho da duplicação forense na criação de imagens em formato RAW. O sistema de arquivos FAT32 apresentou um melhor desempenho para este formato. Para o formato E01, o uso do sistema de arquivos FAT32 e de 512 setores por bloco garantiu um melhor desempenho. Entretanto, quando o formato da imagem for o E01 comprimido, dados heterogêneos influenciam negativamente o desempenho. A partir destes resultados, a aplicação dos passos adicionais propostos apontou para a melhor configuração a ser usada para obter um melhor desempenho na duplicação forense.

ABSTRACT

PROPOSAL OF A METHOD FOR PERFORMANCE IMPROVEMENT IN FORENSIC DUPLICATION OF HARD DRIVES

Author: Daniel José Salomoni
Supervisor: Rafael Timóteo de Sousa Júnior
Programa de Pós-graduação em Engenharia Elétrica
Brasília, July of 2012

The increase in size of the hard disk drive requires a time increasing the forensic analysis of data. One of the stages of forensics is the preservation of digital evidence, and forensic duplication is a technique used to ensure that preservation. This work aims to check if the performance of forensic duplication process can be optimized, through a proposed method that points to the best performance scenario, based on the performance of known tools. The scenarios tested include the variation of the destination file system, the size of sectors per block of the destination image file and the nature of the source data. Were used the tools Linen, FTKImager and EWFAcquire for images in E01 format, and the tools DC3DD and DCFLDD for images in RAW format. The results show that very heterogeneous source data and the number of sectors per block does not influence the performance of forensic duplication for images with RAW format. Also, for the RAW format, the FAT32 file system showed a better performance. For the E01 format, using the FAT32 file system and 512 sectors per block ensured a better performance. However, when the image format used was the compressed E01, heterogeneous data negatively influenced the overall performance. From these results, the application of the proposed additional steps pointed to settings to use for acquiring a better performance in forensic duplication.

SUMÁRIO

LISTA DE TABELAS

LISTA DE FIGURAS

LISTA DE SÍMBOLOS, NOMENCLATURA E ABREVIAÇÕES

ABNT Associação Brasileira de Normas Técnicas.
AFF *Advanced Forensics Format.*
ATA *Advanced Technology Attachment.*
CFTT *Computer Forensic Tool Testing*.
CHS *Cylinder, Head, Sector.*
CRC *Cyclical Redundance Check.*
DCO *Device Configuration Overlay.*
DOS *Disk Operating System.*
ECC *Error Correction Code.*
FBI *Federal Bureau of Investigation.*
FTK *Forensic ToolKit.*
FAT *File Alocation Table.*
HPA *Host Protected Area.*
IDE *Integrated Drive Eletronics.*
IDS *Intrusion Detection System.*
IOCE *International Organization on Computer Evidence*.
LBA *Logical Block Addressing.*
LFN *long file name*.
MTF *Master File Table.*
MD5 *Message Digest Algorithm 5.*
NTFS *New Technologies File System.*
NIST *National Institute of Standards and Technology*.
PATA *Paralell Advanced Technology Attachment.*
PICL *Preservation, Isolation, Correlation and Logging.*
RAID *Rapid Action Imaging Device.*
SATA *Serial Advanced Technology Attachment.*
SCSI *Small Computer System Interface.*
SENASP Secretaria Nacional de Segurança Pública.
SFN *Short File Name*.
SWGDE *Scientific Working Group on Digital Evidence.*
UFS *Unix File System.*

1. INTRODUÇÃO

Segundo (Moore, R. 2005), os crimes de computadores, também chamados de crimes cibernéticos, se referem a qualquer crime que envolva um computador e uma rede de computadores.

Os vestígios de crimes são usados como provas para julgar o responsável pelo mesmo. A detecção de tais vestígios demanda o uso de técnicas de análise em equipamentos computacionais que possam ter sido usados como ferramenta do crime ou que tenham sido o alvo do ataque criminoso (Kruse, W.G. e Heiser, J. G. 2002).

O Código de Processo Penal Brasileiro (Código de Processo Penal, 1941), em seu artigo 158, demanda que sejam realizados exames de corpo de delito quando o crime deixar vestígios. O artigo 159 indica a figura do perito oficial como realizador de tais exames. Na falta de peritos oficiais, o parágrafo 1° desse artigo permite que o exame seja realizado por duas pessoas idôneas, portadoras de diploma de cursos superior preferencialmente na área específica relacionada com a natureza do exame.

De acordo com (Carrier, B., 2006), as análises periciais podem usar técnicas do tipo *live* e/ou *dead*. As técnicas *live* são usadas se o sistema investigado continua em execução e o examinador usa programas do sistema operacional para acessar as informações relevantes. Nas técnicas *dead*, o sistema é desligado, uma cópia do disco rígido é feita, e essa cópia é analisada em laboratório usando aplicações confiáveis.

Em (*National Academy of Sciences*, 2009) é dito que os resultados de exames forenses, enquanto análise científica, devem ser completos e bem embasados; que devem conter, no mínimo, métodos e materiais, procedimentos, resultados, conclusões, e, conforme o caso, fontes bibliográficas e magnitudes de incerteza nos procedimentos e resultados (por exemplo, níveis de confiança).

A padronização e aplicação de métodos e métricas na perícia forense de evidências digitais é um meio para que os resultados destas perícias possam ser mensurados e considerados trabalhos científicos, cujos resultados possam ser reproduzidos e validados. De acordo com (Hankins *et all*, 2011) houveram algumas tentativas de criar metodologias de computação forense a partir do estudo do outros campos de desenvolvimento, especialmente a partir da ciência forense. Apesar dessa base sólida, as pouco mais de duas décadas de existência da

computação forense ainda não foram suficientes para que fossem desenvolvidas metodologias consistentes.

A iniciativa da *International Organization on Computer Evidence* (IOCE), na busca de um método para computação forense, enunciou princípios básicos para a realização de perícias em evidências digitais. Um dos princípios é "Ao lidar com evidência digital, as ações tomadas não devem alterar tal evidência.". A solução usada para evitar alterações na evidência digital é a duplicação forense, processo pelo qual todos os bits do dispositivo de origem são copiados para um dispositivo de destino. Toda a análise será realizada sobre essa cópia da evidência.

Atualmente, os meios mais usados e com maior capacidade de armazenamento de dados são os discos rígidos. O processo de duplicação forense de discos rígidos também é chamado de clonagem, espelhamento ou criação de imagem de disco. Entretanto, existem algumas particularidades na duplicação forense que devem ser observadas. Existe a possibilidade de contaminação da cópia, quando a cópia de disco para disco sofre contaminação de dados, dependendo do nível de abstração de dados usados. Podem existir também volumes de dados não acessíveis, quando as estruturas lógicas do disco são alteradas com o intuito de ocultar informações importantes. Por último há risco de alterações não intencionais da evidência. Apesar de um disco rígido ser classificado como memória secundária não volátil, ele é um equipamento eletrônico que pode sofrer com a ação do tempo, umidade, campos magnéticos, causando a alteração involuntária de seu conteúdo.

Além disso, a crescente capacidade de armazenamento de informação de mídias digitais leva ao aumento de tempo necessário ao processo de duplicação forense, impactando no tempo de realização da análise forense da evidência digital.

Para atender a essas duas necessidades, o presente trabalho apresentará um método para duplicação forense, em adição aos métodos já existentes, que busca orientar a escolha de parâmetros que possam melhorar o desempenho desse processo. Os parâmetros estudados são relacionados a ferramentas, sistemas de arquivos e tamanho do bloco de leitura do disco. Além disso, serão feitas considerações acerca da influência da natureza dos dados de origem e do tamanho do disco rígido na velocidade de aquisição da evidência digital.

Existem diferentes ferramentas de hardware e de software que podem ser usadas para realizar a duplicação forense. Em conjunto com o método criado, serão realizados experimentos de duplicação forense com essas ferramentas, obtendo dessa forma resultados sobre o desempenho das mesmas com a variação dos parâmetros estudados. Os resultados obtidos, em

conjunto com a aplicação do método proposto, devem fornecer subsídios que permitam aumentar a eficácia do processo.

1.1. OBJETIVO

O objetivo deste trabalho é a definição de parâmetros que, em conjunto com um método definido, aumentem a eficiência da duplicação forense de discos rígidos, levando em conta os princípios de preservação de evidência digital exigidos pelos processos criminais.

Os objetivos específicos buscados por este estudo são:

- Levantamento de características de discos rígidos e sistemas de arquivos;
- Proposta do método para duplicação forense;
- Realização de testes comparativos entre ferramentas de duplicação forense.

1.2. HIPÓTESE

O presente estudo trabalha com a hipótese de que existem parâmetros do processo de duplicação forense que, usados em conjunto com um método para esse processo, são capazes de acelerar o processo de duplicação forense.

1.3. JUSTIFICATIVA

O contínuo desenvolvimento tecnológico impele o aumento da capacidade de armazenamento de dados digitais, a diversificação dos tipos de dispositivos de armazenamento e a diminuição de seus custos de produção. Segundo (Tessmann, C.R., 2012 apud Hoelz, B., 2009), o aumento da demanda por análises periciais de informática na Polícia Federal Brasileira cresceram quase que exponencialmente de 2005 a 2009, e que este dado também é verificado em outros órgãos. No Instituto-Geral de Perícias do Rio Grande do Sul, por exemplo, o tamanho médio dos discos rígidos examinados pelo órgão em 2011 chegou a 198,12 GB (Tessmann, C.R., 2012). Essa evolução é mostrada na Figura 1-1.

A manipulação descuidada de evidências digitais pode tirar seu valor enquanto prova de um crime. Para que os resultados da análise pericial sejam aceitos na área criminal e que não sejam facilmente refutados ou invalidados, a evidência digital não pode ter sido alterada.

As soluções de duplicação forense por software tem baixo custo. Porém, o conhecimento sobre a ferramenta deve ser profundo, para que as evidências digitais não sejam alteradas ou até destruídas.

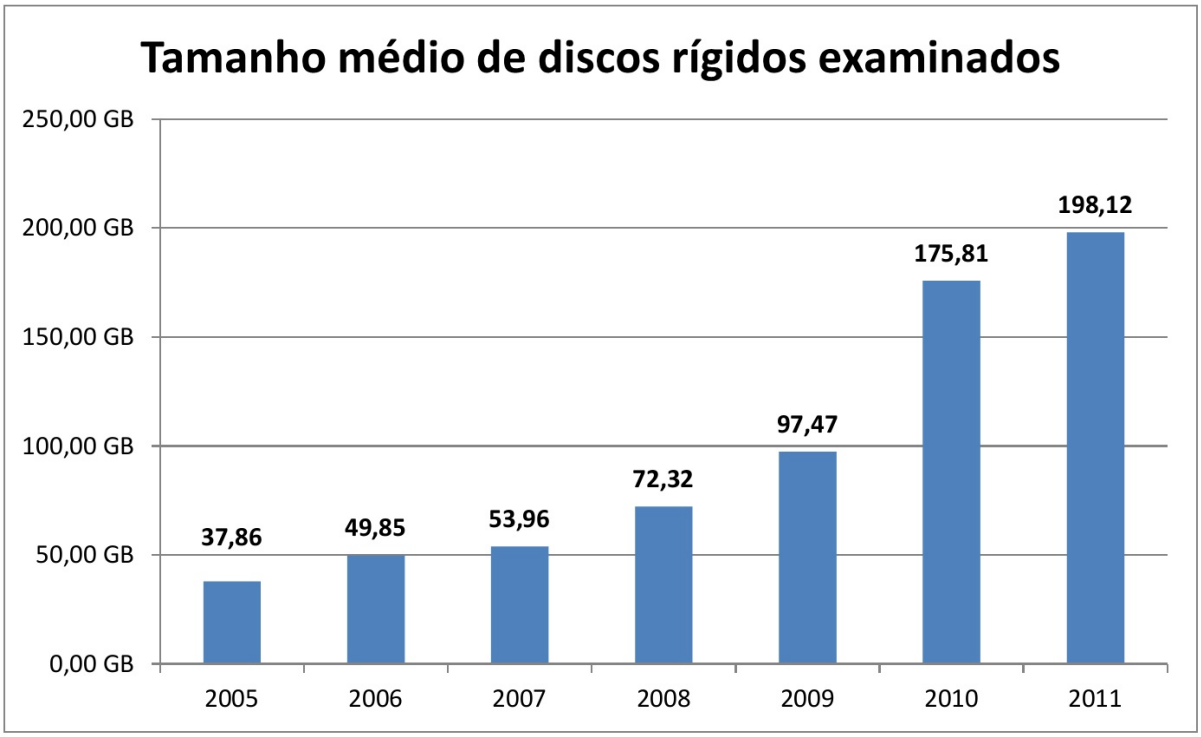

Figura 1-1: Tamanho Médio de discos Rígidos Examinados, adaptada de (Tessmann, C.R., 2012)

O aumento do volume de evidências digitais gera uma demanda que parece sobrecarregar a capacidade dos órgãos responsáveis por analisá-las. É necessário que o método aplicado a essa análise também favoreça a agilidade da análise pericial, sem prejudicar sua precisão.

Assim, o presente trabalho vai ao encontro das necessidades de maior agilidade no exame forense de discos rígidos, focando na etapa de Duplicação Forense, e de um método que garanta a legalidade e valor jurídico da evidência examinada.

1.4. ORGANIZAÇÃO DA DISSERTAÇÃO

O capítulo 2 irá apresentar os estudos teóricos realizados, divididos nos seguintes temas: aspectos jurídicos dos crimes cibernéticos; estrutura e funcionamento de discos rígidos; estrutura dos sistemas de arquivos; características de arquivos de imagem forense; e ferramentas de duplicação forense.

No capítulo 3 serão apresentadas as metodologias utilizadas na etapa de aquisição de evidências digitais.

O capítulo 4 mostra os experimentos realizados, com a consolidação e análise dos resultados encontrados.

O capítulo 5 apresenta as conclusões do presente estudo.

2. REFERENCIAL TEÓRICO

Este capítulo apresentará os elementos teóricos usados para abordar o problema em estudo, discutindo como estes se integram dentro da proposta de trabalho desta dissertação. As seções a seguir apresentarão o embasamento legal da preservação de evidências, o funcionamento e organização dos discos rígidos, os métodos de duplicação forense estudados e ferramentas de duplicação forense.

2.1. ASPECTOS JURÍDICOS DOS CRIMES CIBERNÉTICOS

Uma das definições de crime, de acordo com (Marques, J. F., 2002), é: "fato típico, antijurídico e culpável", onde "típico" refere-se ao fato delituoso que é definido em lei.

Os crimes de informática não estão tipificados, ou seja, não são definidos como crime no Código Penal Brasileiro. Há dois projetos de lei (PL), de números 84/1999 e 2793/2011, que alteram o Código Penal, incluindo nele a tipificação dos crimes de informática. Tais projetos se encontram em discussão na Câmara e no Senado Federal. Assim, Os tipos penais usados para nos crimes de informática são os que apresentam os mesmos efeitos de crimes já tipificados, como calúnia, pedofilia, por exemplo. Esses tipos de crime de informática são classificados como crimes comuns, de acordo com (Carvalho, A. C. A. P, *et all*, 2008). A Tabela 2-1 apresenta esta classificação.

Tabela 2-1: Classificação de Crimes de Informática de Acordo com (Carvalho et all, 2008)

TIPO DE CRIME DE INFORMÁTICA	DESCRIÇÃO	EXEMPLO
PURO	O agente visa o sistema de informática, em todas as suas formas ou manifestações.	Acesso indevido aos dados e sistemas contidos no computador.
MISTO	O agente não visa o sistema de informática, mas a informática é instrumento indispensável para consumação da ação criminosa.	Transferência de fundos de uma conta bancária para outra (pressupondo que os registros bancários existem somente na forma de dados de sistemas informatizados).
COMUM	O agente não visa o sistema de informática, mas usa a informática como instrumento (não essencial, poderia ser outro o meio) de realização da ação.	Distribuição de pornografia infanto-juvenil pela internet.

O artigo 170 do Código Penal exige que "Nas perícias de laboratório, os peritos guardarão material suficiente para a eventualidade de nova perícia." (Código Penal, 1941).

2.2. DISCOS RÍGIDOS

O dispositivo de armazenamento secundário de informações digitais mais comum é o disco magnético (Silberschatz A. *et all*, 2000). O disco rígido é um dispositivo de entrada e saída de dados digitais, formado por um conjunto de discos magnéticos metálicos, cabeças de leitura e gravação (uma para cada face do disco) e o motor de rotação. O motor de rotação gira os discos a uma velocidade constante, e as cabeças de leitura/gravação se movem acima do disco, suspensas sobre uma camada de ar de poucos milionésimos de polegada. Essa camada de ar é produzida pelo movimento de rotação dos discos.

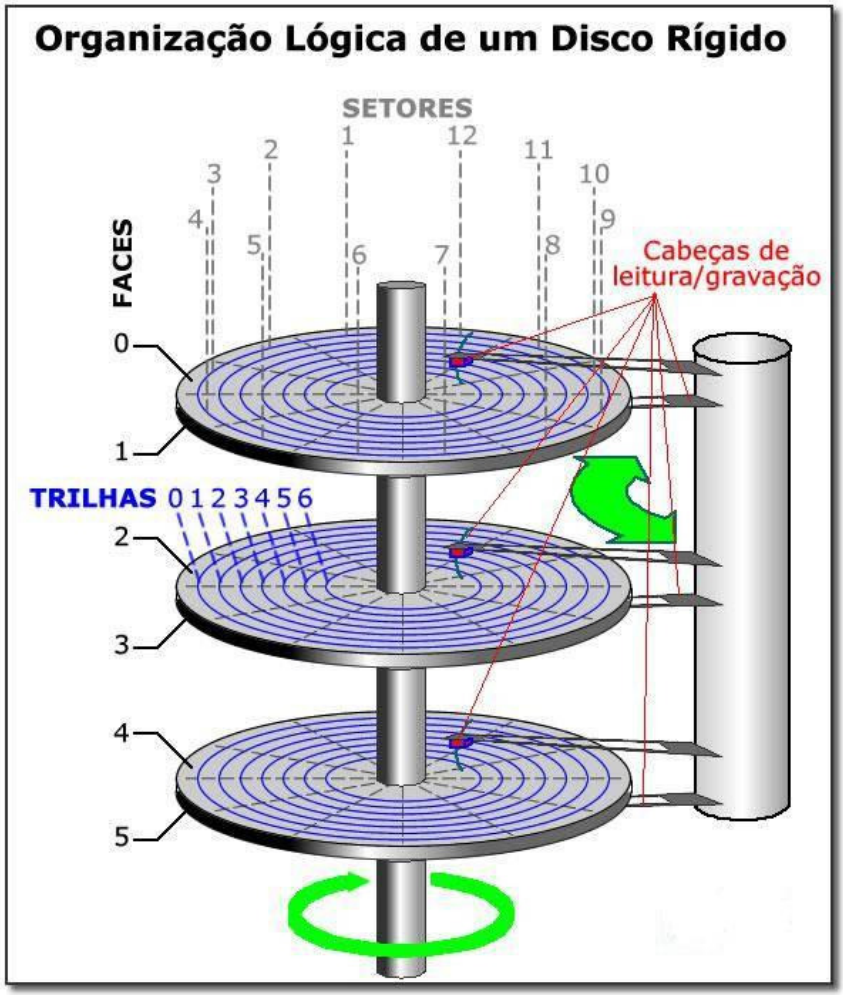

Figura 2-1: Diagrama de um Disco Rígido, adaptado de (Piropo, B., 2007)

As estruturas de dados básicas em um disco são as trilhas e os setores. As trilhas são anéis concêntricos, endereçadas de fora para dentro no disco, começando com Zero, como mostrado na Figura 2-1. As trilhas são divididas em setores de mesmo tamanho de bytes, como

mostrado na Figura 2-2. Entretanto, as trilhas mais externas podem conter mais setores que as trilhas mais próximas ao centro do disco. O conjunto de trilhas de mesma posição nos diferentes discos internos é chamado de cilindro, cujo endereçamento também começa em Zero.

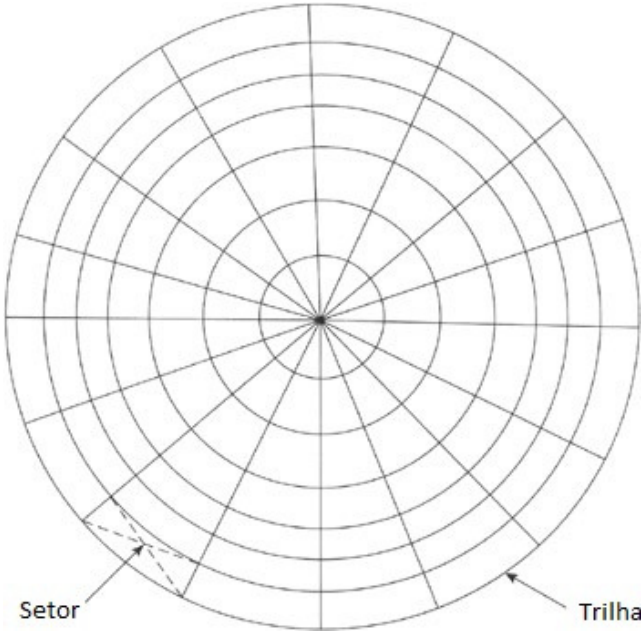

Figura 2-2: Trilha e Setor em uma Face do Disco Rígido

O setor contem as seguintes informações (Figura 2-3):

- GAP ou intervalo: espaço que separa os setores;
- CAB ou Cabeçalho do setor: dados de trilha/cilindro, lado do disco e número do setor (começando em Um);
- Dados: usualmente tem tamanho de 512 bytes;
- CRC (*Cyclical Redundance Check*): controle de erros;
- ECC (*Error Correction Code*): Dados de correção de erros. Quando um setor é gravado, o ECC é atualizado com um valor calculado a partir dos bytes da área de dados. Quando o setor é lido, o ECC é recalculado e comparado com o valor armazenado (Silberschatz A. *et all*, 2000).

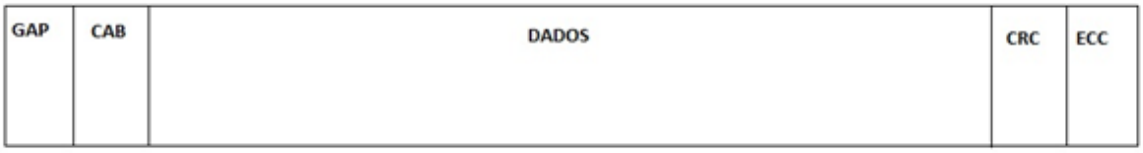

Figura 2-3: Diagrama de um Setor

Uma especificação de sistema de arquivos chamada de AF (*Advanced Format*) usa um tamanho do setor de Quatro Kbytes. Entretanto, os cálculos de ECC também são modificados a fim de oferecer maior confiabilidade. Além disso, oferece compatibilidade para acesso a setores lógico de 512 bytes, mesmo com setores físicos de Quatro Kbytes. Porém, há queda de desempenho quando usado neste modo.

Os discos rígidos realizam operações de leitura/escrita com vários setores de cada vez. O número de setores que podem ser processados em uma operação é chamado de bloco, e essa característica é chamada de *blockmode*.

Um conceito semelhante é o de cluster. Um cluster (também chamado de agrupamento) é a menor parte reconhecida pelo sistema operacional, e pode ser formado por vários setores.

Um arquivo com um número de bytes maior que o tamanho do cluster, ao ser gravado no disco, é distribuído em vários clusters. Um cluster não pode conter dados de mais um arquivo.

Considerando que um cluster associado a um arquivo foi encontrado, todos os seus setores podem ser lidos sem necessidade de um acesso adicional.

Essa organização por setores/clusters é gerenciada através de uma Tabela de Alocação de Arquivos (*File Alocation Table* - FAT). Nessa tabela, cada entrada dá a localização física do cluster associado a cada arquivo lógico.

Segundo (Botchek, R., 2008), entre as métricas usadas para medir desempenho de discos rígidos estão taxa de transferência da interface, tamanho do *buffer*, tempo médio de busca, latência de giro (ou rotacional), tempo de busca trilha-a-trilha.

A métrica “tempo de acesso” é calculada usando três fatores (Zelenovsky R. e Mendonça, A., 2006):

- Tempo de posicionamento ou *seek time*: tempo gasto para movimentar as cabeças entre um cilindro e outro;

- Tempo de chaveamento da cabeça: tempo gasto para selecionar o prato a ser acessado;
- Latência de giro: tempo gasto até que a cabeça se posicione sobre o setor desejado.

Em uma duplicação forense de disco rígido, o disco inteiro é lido sequencialmente, do início ao fim. Assim, o tempo de acesso não é relevante para o desempenho. A natureza sequencial da cópia também elimina benefícios de taxas de transferência de interface e de cachês de disco maiores. Assim, o parâmetro de desempenho mais importante é a taxa de transferência de mídia, que é a taxa em que o disco rígido realmente lê ou grava dados na mídia física (Botchek, R., 2008). As características que afetam esse parâmetro são mostradas na Tabela 2-2.

Tabela 2-2: Características que afetam a Taxa de Transferência de Mídia. (Botchek, R., 2008)

Característica	Descrição
Rotações por Minuto	Quanto mais rápido o número de rotações por minuto de um disco rígido, mais rápida a taxa de transferência de mídia. Os discos com 7200 RPM giram 33% mais rápido que discos de 5400 RPM, e devem ter uma taxa de transferência de mídia 33% maior.
Tecnologia de gravação e densidade de bits	Com o aumento do número de bits que podem ser gravados numa área da superfície do disco, causado pela diminuição do espaço entre os bits de um setor ou diminuição do espaço entre trilhas, a taxa de transferência de mídia aumenta.
Evolução da tecnologia dos componentes	A tecnologia das cabeças de leitura/gravação deve suportar uma maior taxa de rotação e maior densidade de bits do disco.
Diferença entre tempo de leitura e gravação	Alguns discos leem dados 20% mais rápido do que gravam dados no mesmo.
Posição de dados no disco rígido	Os dados nas trilhas mais externas do disco são mais numerosos que as trilhas mais internas. Assim, a taxa de transferência diminui à medida que as trilhas mais externas são lidas.

2.2.1. Interfaces de Disco

As primeiras interfaces com discos rígidos usavam o padrão IDE (*Integrated Drive Eletronics)* (Zelenovsky, R. e Mendonça, A., 2006). Esse padrão foi batizado posteriormente como ATA (*Advanced Technology Attachment*). Em 2003, surgiu a proposta de um ATA serial, denominada SATA (*Serial ATA*), enquanto que o padrão ATA/IDE também passou a ser designado como PATA (*Paralell ATA*). As interfaces seriais SATA e FireWire800 usam a codificação “8b/10b”, onde 1 byte é igual a 10 bits transferidos pela interface serial.

A Tabela 2-3 mostra as velocidades máximas das principais interfaces usadas.

Tabela 2-3: Velocidades Máximas das Principais Interfaces de Discos Rígidos. (adaptado de Botchek, R., 2008)

Interface	Versão	Pico de Taxa de dados (bits/segundo)	Pico de Taxa de Dados (Mbytes/segundo)
USB	USB 1.1	-	1.5
	USB 2.0	480 Mbits/s	48
	USB 3.0	-	625
IDE/PATA	IDE/EIDE PIO Mode 0	-	3.3
	IDE/EIDE PIO Mode 1	-	5.2
	IDE/EIDE PIO Mode 2	-	8.3
	IDE/EIDE PIO Mode 3	-	11.1
	IDE/EIDE PIO Mode 4	-	16.7
	UltraDMA/33	-	33
	UltraDMA/66	-	66
	UltraDMA/100	-	100
	UltraDMA/133	-	133
Firewire	IEEE1394A (Firewire400)	400 Mbits/s	40
	IEEE1394A (Firewire800)	800 Mbits/s	80
SCSI	SCSI-1	-	5
	SCSI-2	-	10
	SCSI-2 Wide3	-	20
	Ultra SCSI	-	20
	Ultra Wide SCSI	-	40
	Ultra2 SCSI	-	80
	Ultra3 SCSI	-	160
	Ultra SCSI 320	-	320
SATA	SATA I	1.5 Gbits/segundo	150
	SATA II	3.0 Gbits/segundo	300
	eSATA	Mesmo que SATA I ou SATA II	Mesmo que SATA I ou SATA

A interface de discos rígidos SCSI (Small Computer System Interface) permite a conexão de até 14 dispositivos simultaneamente (Zelenovsky R., Mendonça, A. 2006). O seu custo mais elevado restringiu seu uso a computadores de alto desempenho.

2.2.2. Endereçamento

Para acessar um determinado setor no disco rígido, é necessário saber sua localização. Assim, cada setor tem um endereço físico, relativo ao início do disco físico e um endereço lógico, criado posteriormente para atender ao aumento da capacidade dos discos. Existem dois tipos de endereçamento físico. Os discos rígidos mais antigos usam a geometria do disco e o método CHS, onde um endereço é formado pelo número de Cilindro, Cabeça e Setor. O

esquema de endereçamento CHS atingiu seu limite de endereçamento, e não é mais usado. A especificação ATA original usava um valor de 16 bits para o número do cilindro (2 ^ 16 = 65526 cilindros), 4 bits para o número da cabeça e 8 bits para setor (2 ^ 8 = 32000). Mas os programas de BIOS usavam 10 bits do cilindro, 8 bits da cabeça e 6 bits do setor. Portanto, para comunicar com o disco rígido através do BIOS, o menor tamanho para cada valor tinha de ser usado, o que permitiu um disco com tamanho máximo de 504 MB (Carrier, B., 2005).

Para contornar o limite de 504 MB, foram desenvolvidas novas BIOS que traduziam os intervalos de endereços das BIOS antigas para a especificação ATA. Por exemplo, em uma solicitação de leitura do cilindro=8, cabeça=4, e setor= 32, a BIOS traduz essa solicitação para cilindro= 26, cabeça=2, setor= 32 do disco. Porém, o processo de tradução não funcionaria para discos com mais de 8.1GB.

Para superar esse novo limite, os endereços CHS foram eventualmente abandonados, e o endereçamento de bloco lógico (LBA) se tornou padrão. LBA usa um único número, começando em 0, para tratar de cada setor. Com LBA, o software não precisa saber sobre a geometria do disco, ele precisa saber apenas um único número. O endereço LBA 0 é CHS 0,0,1 e endereço LBA 1 é CHS 0,0,2. A fórmula de conversão é LBA = (((Cilindro * Cabeças_por_cilindro) + cabeça) * setores_por_trilha) + Setor – 1, onde as variáveis devem ser substituídas pelos respectivos valores CHS. Por exemplo, considere um disco com 16 cabeças por cilindro e 63 setores por trilha. Um endereço CHS de cilindro=2, cabeça=3, e setor=4 seria, em LBA, o seguinte:

$$(((2 * 16) + 3) * 63) + 4\text{-}1 = 2208$$

Com base no endereçamento lógico LBA, cada sistema de arquivos criou conceitos de endereçamento próprios. Esses conceitos são mostrados no item “2.3 Sistemas de Arquivos”.

2.2.3. Formatação Física e Formatação Lógica

A formatação de baixo nível ou formatação física do disco rígido é a divisão da superfície do disco em trilhas e setores, e é usualmente realizada como parte do processo de fabricação (Silberschatz A. et all, 2000). Nessa formatação é possível definir o tamanho do setor. Setores com tamanhos maiores, como 1024 bytes, indicam que menos cabeçalhos e apêndices serão gravados, aumentando o espaço para dados de usuário, mas diminuindo as chances de recuperação de erros do ECC.

A formatação lógica é a preparação dos setores para uso pelo sistema operacional. A estrutura dessa formatação varia de acordo com o sistema operacional usado. A primeira etapa dessa formatação é a divisão do disco em grupos de cilindros, chamados de volumes. Há dois conceitos nessa etapa, segundo (Carrier, B., 2005): Particionamento e Montagem. No particionamento, um volume é dividido em vários volumes menores, e na montagem, múltiplos volumes podem ser agrupados em um volume maior.

A segunda etapa é a criação do sistema de arquivos, quando o sistema operacional armazena as estruturas de dados do sistema de arquivos inicial no disco. Essas estruturas de dados podem incluir mapas de espaço livre e alocado e um diretório inicial vazio (Silberschatz A. *et all*, 2000).

2.2.4. Dados Ocultos

Existem duas funcionalidades de endereçamento de dados no nível de volumes que devem ser levadas em conta na duplicação do disco (Carrier, B., 2005). A HPA (*Host Protected Area*) e o DCO (*Device Configuration Overlay*).

A HPA é uma área especial do disco que pode ser usada para gravar dados. Foi adicionada na especificação ATA-4, com a motivação que os fabricantes pudessem gravar dados que não fossem apagados por operações de formatação do disco pelo usuário. A HPA está no final do disco e, quando usada, só pode ser usada através da reconfiguração do disco rígido. O disco com HPA mostra como final de disco um setor antes do início da HPA.

Para acessar essas informações, o modo de acesso direto ao disco deve ser usado. Nesse modo, comandos são enviados diretamente para a controladora do disco. O comando READ_NATIVE_MAX_ADDRESS irá retornar o último endereço físico, e o comando IDENTIFY_DEVICE irá retornar o último endereço que pode ser acessível pelo usuário. Se o último endereço acessível pelo usuário for maior que o último endereço físico, existe uma área HPA.

O DCO foi adicionado na especificação ATA-6, para permitir a capacidade de limitação de tamanho do disco rígido. O DCO pode esconder setores no final do disco, em adição aos setores escondidos por uma HPA, forçando o comando IDENTIFY_DEVICE a retornar um endereço anterior ao último endereço físico. Para detectar um volume DCO, é usado o comando DEVICE_CONFIGURATION_IDENTIFY, que retorna o tamanho do disco. Esse

tamanho pode ser menor que os valores fornecidos por IDENTIFY_DEVICE e READ_NATIVE_MAX_ADDRESS, acusando o uso de DCO.

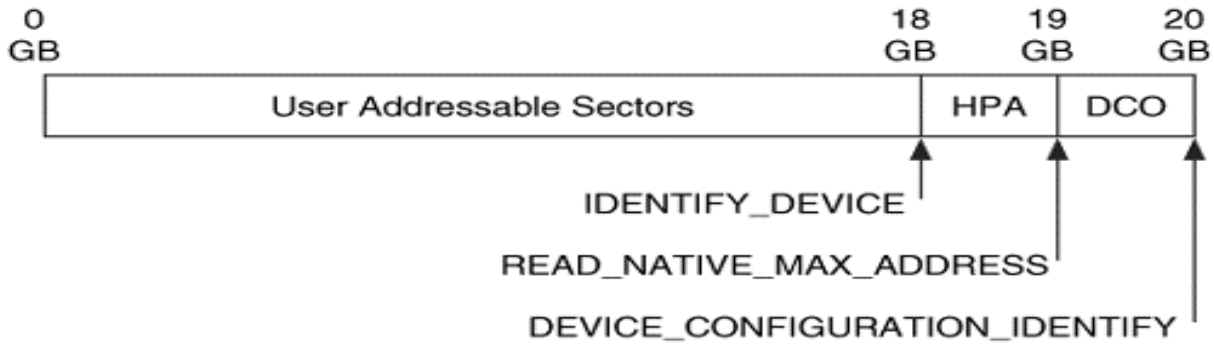

Figura 2-4: Uso de HPA e DCO, adaptada de (Carrier, B., 2005)

Um exemplo de uso de HPA e DCO é ilustrado na Figura 2-4, onde um disco de 20 GB tem apenas 18 GB acessíveis ao usuário, um volume HPA de 1 GB e um volume DCO de 1 GB.

A seção que descreve as ferramentas utilizadas nos experimentos aponta as limitações das ferramentas em relação à detecção de volumes ocultos.

2.3. SISTEMAS DE ARQUIVOS

Os sistemas de arquivos têm procedimentos e estruturas que são usadas para organizar arquivos em uma mídia digital. Existem diversos sistemas de arquivos, usados por diferentes tipos de sistemas operacionais. Os mais usados são os sistemas FAT e NTFS, usados pelos sistemas operacionais Windows, e EXT3, pelos sistemas Linux. A Tabela 2-4 apresenta características de outros tipos de sistemas de arquivos.

Tabela 2-4: Características de Sistemas de Arquivos, adaptado de (Wikipedia [B], 2012)

Sistema	Características
UFS	O Sistema de Arquivos Unix, do inglês *Unix File System* (UFS), é um sistema de arquivos usado por muitos sistemas operacionais Unix e assemelhados. Também é conhecido como *Berkeley Fast File System* (Sistema de Arquivos Rápido da Berkeley), *BSD Fast File System* ou simplesmente FFS.
Minix	Usado no sistema operacional Minix, precursor do Linux. O limite da partição era 64MB, e usava nomes de arquivos curtos.
Ext2	Sistema de disco de alto desempenho usado pelo Linux para discos rígidos e mídias removíveis. O sistema de arquivos *second extended* foi desenhado como uma extensão do *extended* (Ext).

Sistema	Características
Xiafs	Foi desenhado e implementado para ser um sistema de arquivos estável e seguro, e para estender o código do sistema de arquivos Minix. Ele possui a maioria das características requisitadas, sem um complexidade exagerada.
Msdos	Sistema de arquivos usado pelo DOS, Windows, e alguns computadores OS/2. Nomes de arquivos no msdos podem ter 8 caracteres no nome, seguido de um ponto, e 3 caracteres de extensão.
Umsdos	É um sistema de arquivos DOS estendido usado pelo Linux. Ele adiciona habilidades para nome de arquivos longos, UID/GID, permissões POSIX, e arquivos especiais (*devices*, *named pipes*, etc.) em cima do sistema de arquivos DOS, sem sacrificar a compatibilidade com o DOS.
Vfat	Sistema de arquivos DOS estendido usado pelo Microsoft Windows95 e Windows NT. VFAT adiciona capacidade para nome de arquivos longos para o sistema de arquivos DOS.
Iso9660	Sistema de arquivos para CD-ROM de acordo com o padrão ISO 9660.
High Sierra	O Linux suporta High Sierra, o precursor do sistema de arquivos ISO 9660 para CD-ROM. Ele é automaticamente reconhecido como sistema de arquivos iso9660 suportado pelo Linux.
Rock Ridge	O Linux também suporta o *System Use Sharing Protocol records* especificados pelo protocolo de Intercâmbio Rock Ridge. Eles são usados para promover a descrição dos arquivos no sistema de arquivos iso9660 para máquinas UNIX, e fornece informações como nome de arquivos longos, UID/GID, permissões POSIX, e dispositivos. Ele é automaticamente reconhecido como sistema de arquivos iso9660 suportado pelo Linux.
Proc	Sistema de arquivos virtual que é usado como uma interface para estruturas de dados do *kernel*.
Hpfs	O HPFS é o sistema de arquivos utilizado pelo OS/2 da IBM, com recursos que se aproximam muito dos permitidos pelo NTFS, como nome de arquivos com até 254 caracteres incluindo espaços, partições de até 512 GB e unidades de alocação de 512 bytes. Embora muito eficiente, este sistema de arquivos caiu em desuso juntamente com o OS/2.
Sysv	Implementação do sistema de arquivos SystemV/Coherent para Linux. Ele implementa todos os sistemas de arquivos Xenix, SystemV/386, e Coherent.
smb	Para redes que suportam o protocolo SMB, usado pelos Windows for Workgroups, Windows NT, e Lan Manager.
nfs	Sistema de arquivos para redes, usado para acessar discos localizados em computadores remotos.
ncpfs	Sistema de arquivos para redes que suportam o protocolo NCP, usado pelo Novell NetWare.
HFS	Sistema de arquivos desenvolvido pela Apple Computer para uso em computadores rodando o Mac OS. Originalmente projetado para uso em discos rígidos e flexíveis, ele também pode ser encontrado em suporte *read-only* como CD-ROMs. HFS também pode ser referenciado como HFS Standard e Mac OS Standard, sendo que o seu sucessor HFS+ também é chamado HFS Extended ou Mac OS Extended.
HFS+	HFS+ ou HFS Plus é o sistema de arquivos desenvolvido pela Apple, Inc. para substituir o antigo *Hierarchical File System* (HFS) como sistema de arquivos primário usados em computadores OS X e iOS, é um dos formatos, por exemplo, do iPod.
ReiserFS	O ReiserFS foi o primeiro sistema de arquivos com suporte a *journaling* incluído no núcleo Linux 2.4.
exFAT	exFAT (*Extended File Allocation Table*, em português "Tabela de Alocação de Arquivos Estendida", também conhecido como FAT64) é um formato de sistema de arquivos utilizado principalmente em discos de memória flash, introduzido com o Windows Embedded CE 6.0. A utilização do exFAT é uma alternativa para evitar o extensivo uso do sistema de arquivos NTFS.

As seções abaixo apresentam alguns aspectos dos sistemas de arquivos FAT, NTFS e EXT3, com o intuito de tecer um panorama geral sobre seu uso.

2.3.1. FAT

O sistema de arquivo FAT (*File Alocattion Table*) foi usado primeiramente no DOS (*Disk Operating System*) e nas versões do Microsoft Windows 9x. Ainda é bastante usado em

cartões de memória e pendrives USB. Usa poucos tipos de estruturas de dados, sendo a Tabela de alocação de arquivos e a tabela de diretórios as mais importantes. Uma entrada de diretório é alocada para cada arquivo, e contém o nome, tamanho, endereço inicial do arquivo e outros metadados como data e hora. O conteúdo dos arquivos e diretórios é organizado em agrupamentos de setores chamados de clusters. Se um arquivo precisa de mais de um cluster, os clusters seguintes são localizados usando a estrutura FAT. Esta estrutura também é usada para identificar se o cluster está ou não sendo usado. O tamanho do cluster deve ser uma potência de 2, com tamanho máximo de 32 KB. Há três diferentes versões do sistema, FAT12, FAT16 e FAT32, sendo que a maior diferença entre elas está no tamanho dos registros da FAT (Costa, M.A.S.L., 2011).

Há três áreas físicas da FAT, a área reservada, a área da FAT e a área de dados. A área reservada contém o setor de boot, que está localizado no primeiro setor do volume. O setor de boot do FAT32 contém dados adicionais, como o endereço do setor que contém uma cópia backup da FAT.

A área reservada inicia no setor Zero, e tem tamanho de Um setor para FAT12/16 e de vários setores na FAT32.

A área da FAT contém uma ou mais estruturas FAT, e inicia no setor subsequente à área reservada. Seu tamanho é calculado multiplicando o tamanho da FAT pelo número de FATs existentes. A FAT tem basicamente uma entrada para cada cluster do sistema. Dependendo da versão, pode assumir os valores mostrados na Tabela 2-5, indicando que está danificado ou não alocado. Qualquer outro valor indica o número do cluster alocado.

Tabela 2-5: Diferenças entre Versões do Sistema de Arquivos FAT

Característica	**Versão FAT**		
	FAT12	**FAT16**	**FAT32**
Tamanho da Área Reservada (em setores)	1	1	N
Tamanho da entrada FAT (em bits)	12	16	32[1]
Indicação de Cluster não alocado ou danificado (base 16)	0xff7	0xfff7	0xfff fff7
Esquema de nome de arquivos permitidos	SFN (8.3)	SFN (8.3)	SFN e LFN
Tamanho Máximo de Cluster (em Kbytes)	<= 64	<= 64	<= 4
Tamanho Máximo de Arquivo	32MB	2GB	4GB
Número Máximo de Arquivos	4077	65517	268.435.437
Tamanho Máximo do Volume	32 MB	2 ou 4GB	2 TB

[1] O FAT32 usa apenas 28 bits dos 32 bits da entrada de diretórios, podendo armazenar no máximo 268.435.456 clusters.

Outras diferenças entre as versões, mostradas na Tabela 2-5, são o tamanho da área reservada e o tamanho da entrada FAT.

Os nomes de arquivos usam o formato 8.3, ou seja, oito caracteres para nome e três caracteres para extensão. No FAT32, se um arquivo tem nome maior que oito caracteres, ou que contém caracteres especiais, uma entrada de diretório do tipo LFN (*long file name*) é criada, contendo apenas o nome do arquivo, além da entrada SFN (*Short File Name*), que contém os metadados adicionais.

A área de dados contém os clusters que irão armazenar os conteúdos dos arquivos e diretórios. Essa área inicia no setor seguinte à área FAT. O número de setores por cluster é informado no setor de boot. O endereço do primeiro cluster é dois.

2.3.2. NTFS

O sistema NTFS (*New Technologies File System*) é usado em sistemas Windows NT, 2000, XP, 7 e 8. A grande diferença em relação ao FAT é que, com exceção dos primeiros setores que contém o código de inicialização (*boot code*), todo o disco é considerado área de dados, e os dados tipicamente escondidos do FAT estão em arquivos dessa área de dados (Carrier, B., 2005).

A estrutura principal do NTFS é a MTF (*Master File Table*). Cada arquivo e diretório tem pelo menos uma entrada na MFT, e cada entrada tem tamanho de 1 KB, com endereço de 48 bits começando em Zero. Esse endereço é chamado pela Microsoft de número do arquivo.

A própria MFT é um arquivo. Assim, a primeira entrada da MFT é o $MFT, que contém o endereço da MFT no disco. O cluster inicial da MFT é encontrado no setor de inicialização.

As 16 primeiras entradas da MFT são usadas para armazenar metadados do sistema de arquivos. A Tabela 2-6 descreve essas entradas.

Tabela 2-6: Descrição dos Metadados da MFT, adaptado de (Carrier, B., 2005)

Nome da entrada MFT	Registro	Descrição
$MFT	0	A entrada para a própria MFT (Master File Table).
$MFTMIRR	1	Cópia backup dos 16 primeiros registros do MFT.
$LOGFILE	2	Arquivo de registro de transações (*journal*).
$VOLUME	3	Informações sobre o volume, como rótulo (*label*), identificador e versão.
$ATTRDEF	4	Definição de atributos.

Nome da entrada MFT	Registro	Descrição
.	5	Diretório raiz do disco.
$BITMAP	6	Mapeamento e estado de alocação dos clusters do disco.
$BOOT	7	Setor de inicialização e código de inicialização.
$BADCLUS	8	Lista de clusters que tem setores danificados no disco.
$SECURE	9	Informações sobre segurança e controle de acesso a arquivos.
$UPCASE	10	Mapeia cada caractere UNICODE minúsculo para o correspondente maiúsculo.
$EXTEND	11	Diretório que contém arquivos para extensões opcionais.

A estrutura da MFT, mostrada na Figura 2-5, armazena atributos, que são estruturas de dados de com formatos diferentes para cada tipo de atributo.

A maioria das entradas da MFT tem um atributo do tipo $FILE_NAME, que contém o nome do arquivo (em UNICODE), além de tempo de criação, de último acesso e da última gravação, e um atributo do tipo $STANDARD_INFORMATION, que contém informações temporais, informações sobre o proprietário do arquivo e outras informações de segurança, além do atributo $DATA, com o conteúdo do arquivo. Cada diretório tem um atributo $INDEX_ROOT, com informações sobre os arquivos e subdiretórios armazenados abaixo dele.

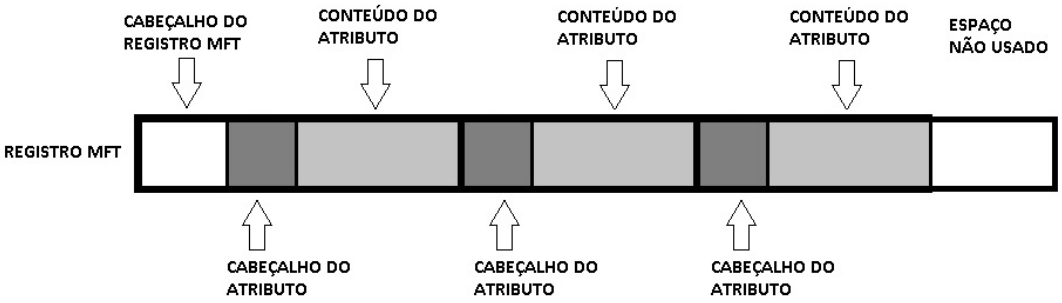

Figura 2-5: Estrutura da MFT

O NTFS é um padrão proprietário da Microsoft. Seu uso no sistema Linux não é nativo, mas sim realizado por *drivers* "NTFS-3G", versão gratuita dos *drivers* TUXERA NTFS da empresa Tuxera Inc., que já estavam disponíveis no HELIX3 (E-fense Helix3, 2009). Esses drivers utilizam uma interface chamada FUSE (*Filesystem in Userspace*), que é um programa que executa as leituras/gravações no sistema de arquivos NTFS.

2.3.3. EXT3

O EXT3, evolução do EXT2, é o sistema de arquivos padrão de muitas distribuições do Linux, e foi baseado no UFS (*Unix File System*). A estrutura do sistema inicia em uma área reservada, e o restante do sistema é dividido em seções, chamados de grupos de blocos (Costa, M.A.S.L., 2011). Todos os grupos tem o mesmo tamanho, exceto pelo último. Os grupos de blocos armazenam nomes de arquivos, metadados e conteúdos dos arquivos, e cada grupo tem seu descritor de grupo, armazenado em uma tabela de descritores de bloco, localizados no setor seguinte ao fim do superbloco.

O superbloco fica 1024 bytes depois do início do sistema de arquivos, tem tamanho de 1024 bytes e guarda informações sobre a estrutura do sistema, como tamanho de bloco, número total de blocos e número de blocos por grupo de bloco. Também contém o número total de inodes e o número de inodes por grupo de blocos. Os inodes são estruturas que armazenam os metadados de arquivos e diretórios. Os inodes tem tamanho fixo e são armazenados em tabelas de inodes, sendo que para cada grupo de bloco há uma tabela de inodes. O conteúdo de arquivos é armazenado em blocos, que são grupos de setores consecutivos. A Figura 2-6 ilustra a relação entre entradas de diretório, inodes e grupos de blocos.

O nome do arquivo é armazenado em uma entrada de diretório, que está localizada nos blocos alocados para o diretório "pai" desse arquivo. A entrada de diretório contém o nome do arquivo e um ponteiro para o inode desse arquivo.

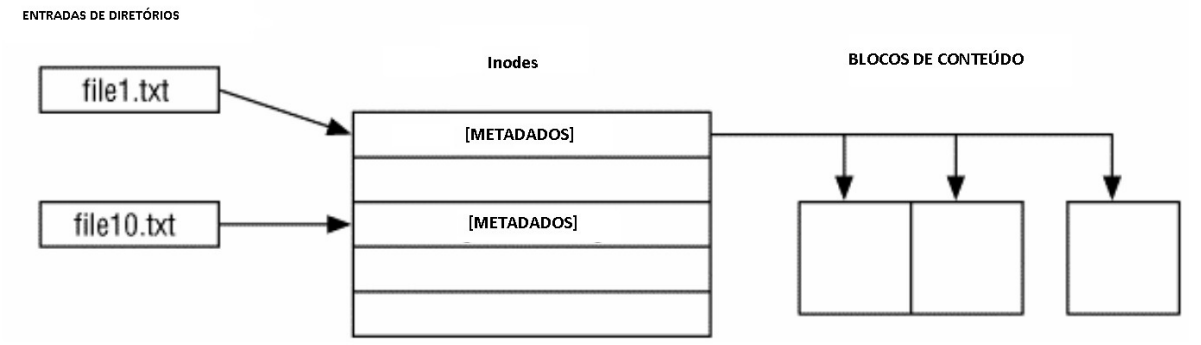

Figura 2-6: Estrutura de Dados do EXT3

O estudo aprofundado sobre esses sistemas de arquivos é necessário para a fase de análise da evidência. Para a etapa de duplicação forense é importante conhecer cada sistema e suas limitações, a fim de escolher o sistema que melhor atende as necessidades dessa etapa.

2.4. FORMATAÇÃO DOS DADOS ADQUIRIDOS

Segundo (Carrier, B., 2005), os dados digitais em um disco rígido podem ser interpretados em camadas de abstração, como disco, volume, arquivos e aplicação. A regra geral é que sejam adquiridos dados na camada mais baixa em que a evidência possa ser achada. Na maioria dos casos, todos os setores do disco devem ser adquiridos/copiados.

Como exemplo, quando apenas um volume do disco rígido é adquirido, com a cópia de todos os setores do volume ou partição, será possível realizar procedimentos de recuperação de arquivos apagados. Entretanto, caso a partição tenha sido redimensionada, os setores que anteriormente pertenciam àquela partição não estarão inclusos na cópia.

No caso de cópia lógica de arquivos, com em um backup de uma empresa, onde apenas os arquivos visíveis nas tabelas de alocação são copiados, não será possível realizar um procedimento de recuperação de arquivos.

Em alguns casos, como em análises de IDS, pode ser necessário copiar somente arquivos lógicos, como registros de operações (logs).

O resultado da duplicação forense pode ser uma cópia forense ou uma imagem forense (Costa, M.A.S.L., 2011). Na primeira, todo o conteúdo da evidência é copiado bit-a-bit para uma mídia de destino, enquanto que na segunda o conteúdo é copiado para um arquivo que guarda as informações relativas à geometria do conteúdo. A imagem forense pode ser do tipo simples, encapsulada ou simples com dados adicionais (Carrier. B., 2005), mostrados na Figura 2-7. Os itens abaixo descrevem esses formatos.

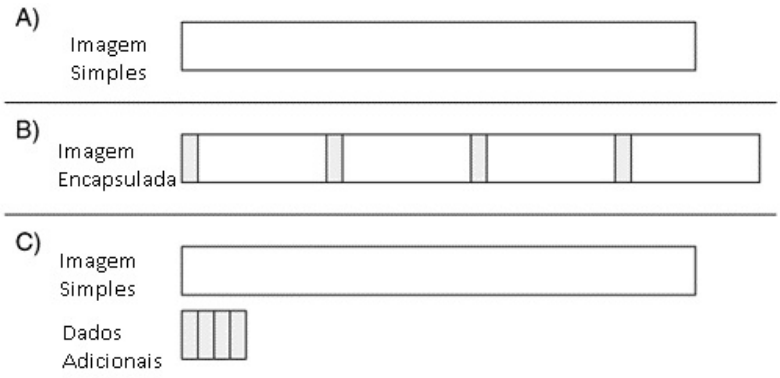

Figura 2-7: Formatos de Arquivos de Saída, adaptado de (Carrier, B., 2005)

Cada formato tem seu uso específico, dependendo do objetivo da perícia. Em casos de recuperação de documentos apagados e busca de palavras chave, o encapsulamento é mandatório para evitar contaminação com dados do disco destino. Se o objetivo é executar o sistema operacional na máquina física, o encapsulamento não deve ser usado. A execução virtual do sistema operacional da máquina pode usar um arquivo encapsulado, entretanto o desempenho da execução é inferior ao da execução na máquina física.

Assim, deve-se levar em conta, na escolha do formato de saída do arquivo de cópia, a maneira que o mesmo será usado, e qual ferramenta de análise irá acessá-lo. Os programas Encase para Windows ou FTK (*Forensic ToolKit*) apresentam emulação de disco, onde as informações do arquivo de imagem podem ser acessadas pelo sistema operacional como se fosse um disco de sistema, ou usadas para gravar um clone do disco origem.

2.4.1. Imagem Simples

Uma imagem simples, ou bruta (*Raw image*) contem apenas os dados do disco origem, e é fácil compará-lo com os dados de origem. Uma imagem simples pode ser copiada integralmente para um único arquivo ou segmentada em vários arquivos de mesmo tamanho. Uma grande desvantagem desse tipo de imagem é a possibilidade da ocorrência de contaminação da cópia, quando dados anteriores do disco destino são misturados aos dados de origem. Para evitar a contaminação, é necessário que o disco destino seja sanitizado usando o processo de *wipe*, onde um valor, tipicamente Zero ou Um, é gravado em todos os setores do disco.

Esse tipo de imagem é amplamente usado, pois pode ser lida por diferentes ferramentas de análise forense (Garfunkel, S., *et all*, 2006). Entretanto, esse formato não é comprimido, exigindo que o disco destino tenha pelo menos tanto espaço quanto o disco origem.

2.4.2. Imagem Incorporada

Uma imagem incorporada ou encapsulada (*Embedded image*), contem os dados do disco de origem e dados adicionais sobre a aquisição, como valores de *hash*, informações temporais e de setores defeituosos detectados na aquisição do disco de origem. Também é possível segmentar esse tipo de imagem em vários arquivos. Uma imagem encapsulada não sofre com o problema de contaminação, entretanto a sanitização do disco destino é sempre recomendada, para evitar o uso do argumento legal que a cópia possa estar contaminada.

Em alguns formatos, é possível realizar compressão do arquivo de saída, obtendo assim uma economia de espaço no disco de destino. Entretanto, esse procedimento não será efetivo se os dados do disco origem forem muito variáveis, como vídeos ou imagens. Outro ponto importante é que a ferramenta usada na etapa de análise da perícia deve suportar compressão de arquivo, e que o hardware do computador seja poderoso o bastante para que o atraso no acesso ao arquivo comprimido, causado pela descompressão dos dados, seja desprezível. Existem formatos que independem da ferramenta de análise, como o *AFF (Advanced Forensic Format), o AFF4 (Advanced Forensic Format 4) e o gfzip (Generic Forensic Zip).* Outros formatos são usados em ferramentas de análise específicas. Alguns formatos específicos de ferramentas de análise são: *Expert Witness, ASR, IDIF, ProDiscover, Pyflag, RAID, Safeback e SDi32* (Garfunkel, S., *et all*, 2006). A Tabela 2-7 mostra a ferramenta de análise associada a cada padrão, se o formato é proprietário (se a especificação do formato é disponível publicamente) e se o formato comprimido pode ser pesquisado sem a necessidade de descompressão de todo o arquivo.

Tabela 2-7: Características dos formatos de imagens, adaptada de (Garfunkel, S., *et all*, 2006)

Nome do formato	Ferramenta de Análise	É Proprietário?	Compressão de Dados é
Expert Witness	Encase	Sim	Sim
ASR	SMART	Não	Não
IDIF, IRBF, IEIF	ILook Investigator	Sim	Sim
ProDiscover Image File	Prodiscover	Não	Sim
sgzip	Pyflag	Não	Sim
RAID	DIBS USA´s RAID	Não	Não

Nome do formato	Ferramenta de Análise	É Proprietário?	Compressão de Dados é
Safeback	Safeback	Sim	-
SDi32	Vogon International´s SDi32	Não	Não

O formato Expert Witness, usado pelo Encase da Guidance Software, talvez seja o padrão *de facto* para análise forense. É composto de um cabeçalho com informações sobre o caso, seguido de blocos de dados com códigos CRC para cada bloco de 64 setores, e finaliza com o código hash dos dados (Garfunkel, S., *et all*, 2006).

2.4.3. Imagem Simples com Dados Adicionais

Há também a possibilidade de criação de uma imagem simples, e que os dados adicionais sejam gravados em um arquivo separado. Uma desvantagem desse formato, alerta (Garfunkel, S., *et all*, 2006), é que há a possibilidade desses arquivos sejam perdidos ou confundidos com dados de outro disco rígido.

2.5. TRATAMENTO DE ERROS

O arquivo que contém a cópia da evidência deve ser íntegro, sem alterações em relação à evidência original. Nos casos em que o disco rígido com a evidência original apresentar erros de leitura, que impeçam a leitura de parte da evidência, procedimentos de tratamento de erros devem ser aplicados. Segundo (Carrier, B., 2005), um tratamento de erros aceitável é que o endereço do setor com erros seja reportado, e que os dados não lidos sejam substituídos por zeros.

O tratamento de erros pode variar de acordo com a ferramenta de duplicação forense usada.

Por exemplo, se a ferramenta dd encontrar um erro de leitura no dispositivo de entrada, a ação default é interromper a cópia. Esse comportamento é contornado usando o parâmetro "conv=noerror". Dessa forma, os setores defeituosos serão ignorados, gerando uma imagem com número de setores diferente da evidência original. Para preencher os setores defeituosos com zeros, evitando a alteração do número de setores, é usado o parâmetro "conv=sync, noerror". Esse parâmetro força a gravação em porções do tamanho do bloco escolhido. Se não há dados suficientes para gravar um bloco inteiro, o bloco é preenchido com zeros. A

desvantagem do uso do "sync" é que a imagem terá um tamanho múltiplo do tamanho do bloco, que pode não ser o tamanho real da evidência original.

Na ferramenta dd_rescue de Kurt Garloff (http://www.garloff.de/kurt/linux/ddrescue) o tratamento de erros é habilitado por default. Se um erro é encontrado, o tamanho do bloco é diminuído, e os blocos que não puderam ser lidos são zerados.

A Tabela 2-8 mostra resultados de tratamento de erros em alguns testes realizados pelo CFTT/NIST (CFTT/NIST, 2012). Tratam-se, porém, de exceções no comportamento destas ferramentas.

Tabela 2-8: Resultados de Tratamento de Erros de Aquisição, adaptado de (CFTT/NIST, 2012)

Ferramenta	Comportamento
Encase Linen 6.01	Até sete setores contíguos ao setor defeituoso podem ser preenchidos com zeros.
FTK Imager 2.5.3.14	O endereço de dados corrompidos no arquivo de imagem não foi reportado.
DCCIDD 2.0	Até sete setores contíguos ao setor defeituoso podem ser preenchidos com zeros.

2.6. INTEGRIDADE DA CÓPIA

A verificação de integridade da cópia da evidência original pode ser feita através da comparação bit-a-bit da cópia com o original, ou através de um valor calculado por uma função criptográfica de *hash*.

Segundo (Ruback, M. C., 2011), as funções criptográficas de *hash* são utilizadas na área de segurança da informação para auxiliar em tarefas como verificação de corrupção de dados, autenticação de mensagens e assinatura digital, entre outras.

De acordo com (Schneier, B., 1995), uma função unidirecional de *hash*, *H(M)*, opera sobre uma mensagem *M* – também chamada de pré-imagem – de tamanho arbitrário e deve retornar um valor de *hash h*.

A função criptográfica de *hash* MD5 (*Message Digest Algorithm* 5) recebe mensagens de tamanho variável e retorna o valor de *hash* correspondente de tamanho fixo de 128 bits. Qualquer mudança na mensagem de entrada altera o resultado do *hash*. Entretanto, a função MD5 está em desuso, visto que foram encontradas colisões de *hashes*, ou seja, foram

encontradas duas mensagens de entrada diferentes que produziram o mesmo valor hash. Assim, é aconselhável o uso de outras funções de cálculo de *hash,* como SHA1, SHA128 e SHA256.

O cálculo de *hash* deve ser efetuado usando todos os bits do disco de origem como mensagem M. Assim, um valor *hash* será obtido. Caso algum bit do disco de origem seja alterado, o valor *hash* sofrerá alteração.

O cálculo de *hash*es para segmentos do disco origem podem ser úteis para garantir o uso de partes do disco como evidência mesmo em caso de alteração. Assim, apenas os segmentos que não apresentarem alterações poderão ser usados para análise.

Com o aprendizado dos métodos usados para duplicação forense e escolha das ferramentas a ser utilizadas, foram definidos os experimentos necessários, mostrados no capítulo 4.

2.7. FERRAMENTAS UTILIZADAS NA DUPLICAÇÃO FORENSE

A duplicação forense pode ser realizada por ferramentas de Hardware, com o uso de hardware dedicado ou de bloqueadores de escrita por hardware, ou por ferramentas de Software executadas em computadores. As seções abaixo descrevem estas duas categorias de ferramentas.

2.7.1. Ferramentas de Hardware

O desempenho e a segurança são duas características dos hardwares dedicados, equipamentos específicos para duplicação forense, que independem de computadores para realizar a duplicação. Entretanto, seu alto custo impede seu uso generalizado. Além disso, com novas interfaces surgindo, os equipamentos podem se tornar desatualizados muito rapidamente. Os discos rígidos origem e destino são conectados nesses equipamentos por suas interfaces nativas. Apresenta também funcionalidades de sanitização do disco destino e cálculo parcial e integral de códigos *hash*.

O hardware bloqueador de escrita é conectado ao computador usado na análise e ao disco rígido de origem, e impede que operações de escrita solicitadas pelo sistema operacional do computador usado sejam enviadas à controladora de disco. Seu custo é menor que o hardware dedicado. Dependendo do modelo, usa interfaces eSata, USB ou Firewire para conexão com o

computador, e se conecta com disco rígido através da interface do disco, como SATA e PATA, ou por USB, sendo necessário uma adaptador da interface do disco rígido para USB.

As ferramentas de hardware são especificamente projetadas para realizar duplicação forense. Entretanto, o custo de aquisição destas ferramentas é bastante elevado.

2.7.2. Ferramentas de Software

O uso do bloqueio de escrita por Software deve ser usado com cautela, pois algumas aplicações podem burlar esse bloqueio. É necessário que o software seja certificado.

Por último, pode ser usado um computador pessoal com software específico para duplicação de evidências digitais. Essas ferramentas de software são executadas em distribuições LINUX específicas para investigações forenses, pois apresentam configurações que impedem a alteração da evidência digital.

Em alguns casos, é necessário usar o próprio equipamento suspeito para a cópia da evidência. Como exemplo, temos o caso de mais de um disco rígido no computador questionado, configurados como RAID, ou quando novas interfaces de disco ainda não são suportadas pelos equipamentos do examinador.

3. MÉTODOS DE ANÁLISE FORENSE

Há esforços internacionais para a padronização de procedimentos em análises forenses digitais. Em 1998, a IOCE, composta por forças policiais dos países membros do grupo G8, foi indicado para elaborar princípios para procedimentos relativos a evidências digitais. Os princípios propostos, apresentados em 2002 (IOCE, 2002), são mostrados na Tabela 3-1.

Em 1998 foi criado nos Estados Unidos o *Scientific Working Group on Digital Evidence* (SWGDE), um dos grupos de trabalho que dão suporte aos esforços de padronização da IOCE. Este grupo apresentou um documento com uma proposta de desenvolvimento de procedimentos padrão em evidência digital (SWGDE, 1999). Nos Estados Unidos, este documento foi eleito como base para geração de procedimentos padrão pelo *Federal Bureau of Investigation* (FBI).

Tabela 3-1 - Princípios da IOCE para tratamento de Evidências Digitais

#	Princípio
1	Ao lidar com evidência digital, todos os procedimentos das ciências forenses devem ser aplicados.
2	Ao lidar com evidência digital, as ações tomadas não devem alterar tal evidência.
3	Quando for necessário acessar evidência digital original, o operador deve ser treinado para esse propósito.
4	Toda a atividade relativa à apreensão, acesso, armazenamento ou transferência de evidência digital deve estar totalmente documentada, preservada e disponível para fins de revisão.
5	Um operador é responsável por todas as ações tomadas a respeito da evidência digital enquanto esta estiver sob sua responsabilidade.
6	Qualquer agência que seja responsável pela apreensão, acesso, armazenagem ou transferência de evidência digital é responsável pela observância destes princípios.

A Tabela 3-2 apresenta a nomenclatura definida pelo SWGDE, referente à aquisição de evidências digitais, usadas neste estudo.

A iniciativa Brasileira para padronização dos procedimentos periciais é resultante de acordo firmado entre a Secretaria Nacional de Segurança Pública (SENASP) do Ministério da Justiça, ABNT e Inmetro. A partir desse acordo, foi criado o Comitê Técnico de Ciências Forenses (ABNT/CEE-137 – Ciências Forenses), formado por 18 subcomitês, cada relacionado a uma área de perícia. O subcomitê técnico de Perícias em Informática irá discutir e propor normas técnicas para as perícias de Informática.

Tabela 3-2: Nomenclatura definida pelo SWGDE (SWGDE, 1999)

Termo	Definição
Aquisição de Evidência Digital	Coleta e armazenamento de informação ou itens físicos para análise.
Objetos de Dados	Objetos ou informações com valor probatório potencial que são associados com itens físicos. Objetos de dados podem estar em diferentes formatos sem alterar a informação original.
Evidência Digital	Informação com valor probatório armazenado ou transmitido em formato digital.
Itens físicos	Itens onde os objetos de dados de informações são armazenados ou transmitidos em formato digital.
Evidência Digital Original	Itens físicos e objetos de dados associados com estes itens no momento do procedimento de apreensão ou aquisição.
Duplicação de Evidência Digital	Reprodução exata de todos os objetos de dados contidos em um item físico original.
Cópia	Reprodução exata de uma informação contida em um item físico original, independente do mesmo.

No âmbito internacional, a Convenção de Budapeste, ou Convenção do Conselho Europeu sobre o Crime Cibernético, define os crimes praticados por meio do computador e as formas de persecução penal nos países signatários. Até o presente momento o Brasil não é signatário desse tratado, pela falta de legislação específica e diferenças entre as metodologias empregadas.

Dessa forma, para realização de uma análise forense, usualmente são seguidos os métodos e padrões internacionais propostos pela IOCE e pelo SWGDE.

3.1. ETAPAS DA ANÁLISE FORENSE

Com base nos princípios da IOCE, (Costa, M.A.S.L., 2011) dividiu os procedimentos básicos em um exame forense em computadores em quatro etapas distintas: Identificação, Preservação, Análise e Apresentação. A Figura 3-1 mostra os procedimentos realizados em cada etapa.

Já o método exposto por (Carrier, B., 2005) é chamada de PICL (*Preservation, Isolation, Correlation and Logging),* traduzido como Preservação, Isolamento, Correlação e Registro, que são os passos básicos em uma análise forense. Na fase de Preservação, são adquiridos os dados disponíveis para posterior análise. O procedimento de Isolamento determina que o equipamento não possa se comunicar com outros equipamentos externos. Nesse passo, é necessário desconectar os meios e desabilitar todas as possíveis interfaces de acesso, como placas de rede cabeadas ou sem fio e serviços Bluetooth.

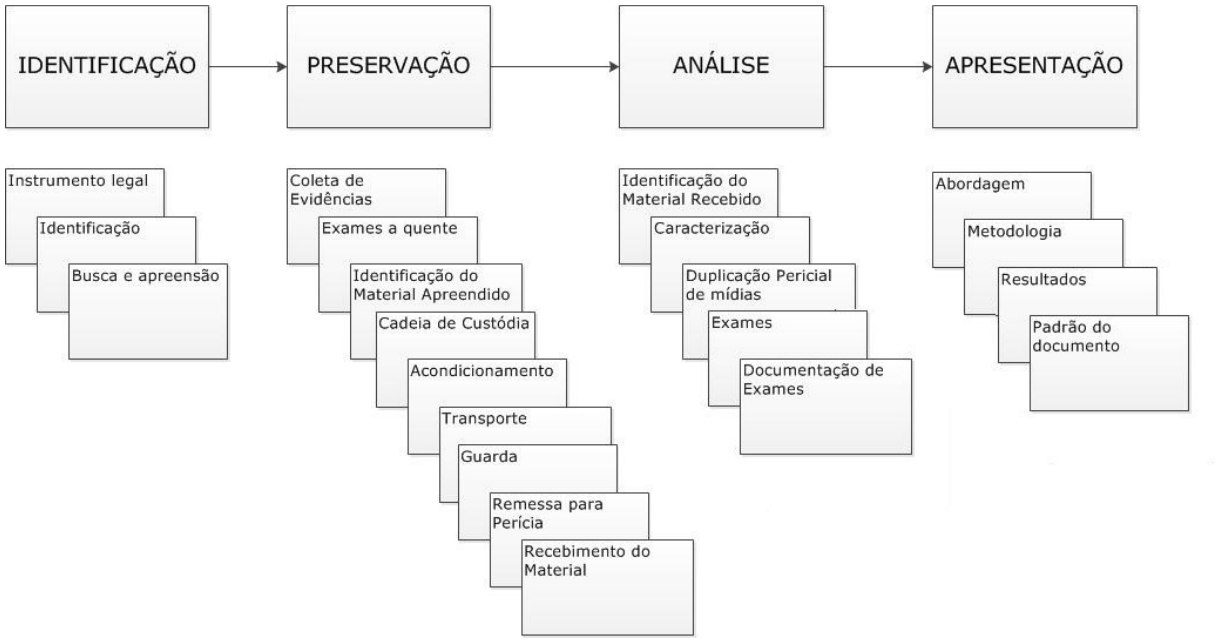

Figura 3-1: Etapas do Exame Forense adaptada de (Costa, M.A.S.L., 2011)

Na fase de Correlação, os dados devem ser validados para determinar se houve tentativa de falsificação. Um dos meios de realizar essa validação é comparar os dados de tempo dos arquivos com outros arquivos de registros de operações.

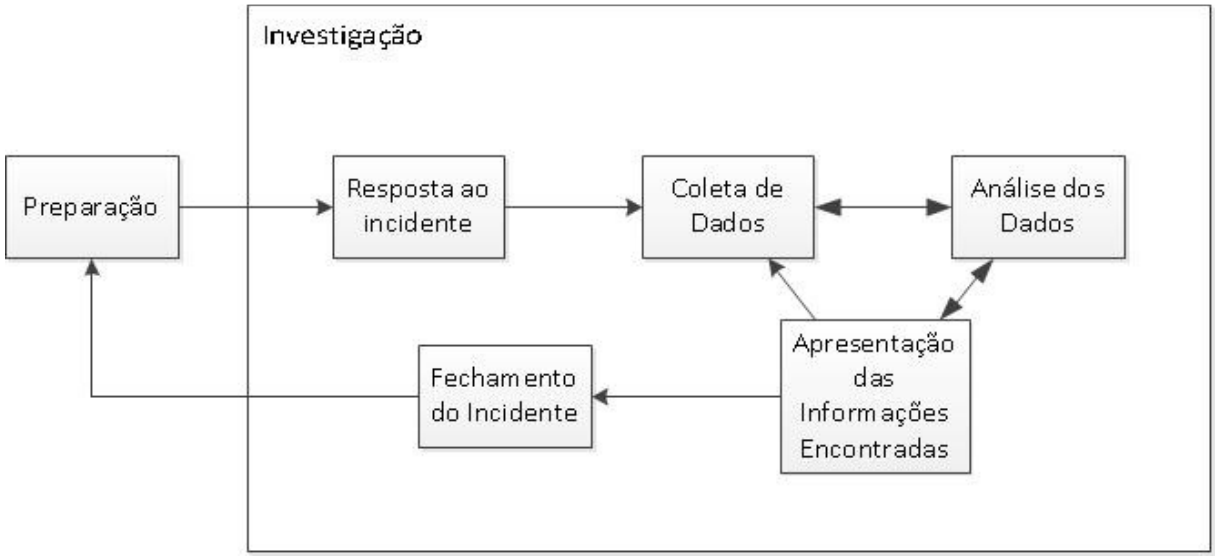

Figura 3-2: Fases da Investigação, de acordo com (Beebe, N. L., Clark, J. G., 2005)

Por fim, é importante o registro e documentação de todas as operações executadas, principalmente as ações que podem gerar modificações em um sistema que está sendo executado.

O método proposto por (Beebe, N. L., Clark, J. G., 2005) também divide o processo em fases e, especificamente na fase de análise, em subfases. As fases são mostradas na Figura 3-2.

A fase de Preparação, onde são desenvolvidos os procedimentos para atendimento aos incidentes, é realimentada com os resultados de cada investigação. Na fase de resposta a incidentes, além da detecção do incidente e coordenação dos recursos usados na investigação, há a formulação do plano de investigação para coleta e análise de evidências.

A seguir são apresentados os métodos para duplicação forense extraídos dos métodos de análise forense estudados.

3.2. MÉTODOS DE DUPLICAÇÃO FORENSE

Os métodos de análise forense apresentam diferentes maneiras de abordar o procedimento de duplicação forense. A abordagem de (Costa, M.A.S.L., 2011) situa o procedimento de duplicação forense na etapa de análise, subetapa "Duplicação Pericial de Mídias". No método PICL de (Carrier, B., 2005), a duplicação é feita na fase de preservação, enquanto que (Beebe, N. L., Clark, J. G., 2005) situam os procedimentos de preservação de evidência na fase de Coleta de Dados.

No procedimento de preservação da evidência segundo (Carrier, B., 2005), os passos a serem seguidos são os seguintes:

- Copiar os dados importantes, colocar os dados em um lugar seguro, e analisar apenas a cópia, para que os dados originais possam ser restaurados caso os dados da cópia sejam modificados;
- Calcular os *hash*es MD5 e SHA dos dados importantes, para que possa ser provado posteriormente que os dados originais não foram modificados;
- Usar um dispositivo bloqueador de escrita durante procedimentos que possam gravar sobre os dados suspeitos.

Para (Beebe, N. L., Clark, J. G., 2005), a prova deve ser preservada para maximizar a disponibilidade evidência e qualidade, e manter a integridade da evidência durante o processo

de investigação digital. Além disso, o princípio da preservação de evidência deve ser aplicado a todas as fases da investigação digital. Algumas regras para atender esse princípio, de acordo com a fase, são mostradas na Tabela 3-3.

Tabela 3-3: Regras para Preservação de Evidência nas Fases de Investigação de (Beebe & Clark, 2005)

Fases da Investigação	**Preservação da prova**
Preparação	Garantir a disponibilidade e qualidade das evidências digitais, quando necessário.
Resposta a Incidentes	Garantir a preservação da evidência durante a validação inicial e avaliação ("resposta ao vivo") das atividades.
Coleta de Dados	Certificar-se que os dados são coletados de forma forense. Algumas atividades incluem a criação de duplicatas forense, uso de tecnologias de proteção contra gravação, calcular de hashes, e ações de proteção do ambiente.
Análise de Dados	As cópias forenses são criadas conforme a necessidade. Além disso, o analista deve estar ciente de quais etapas e processos podem modificar as cópias de trabalho e executar as etapas metodicamente, partindo da abordagem menos invasiva para mais invasiva, ou continuamente retornar ao uso de cópias limpas.
Apresentação dos Resultados	Comunicar os achados de uma maneira que facilite a comprovação futura.
Encerramento do Incidente	Eliminação apropriada das cópias e retenção de informações relacionadas a todo o processo.

A prova deve ser preservada para maximizar a disponibilidade e qualidade da evidência, e manter a integridade da evidência durante o processo de investigação digital.

Um método mais detalhado é exposto em (Daniels D.J.; Hart S.V., 2004). O fluxograma da Figura 3-3 mostra os passos para aquisição de evidência digital desse método, publicado pelo NIJ (*National Institute of Justice*). O número indicado em cada passo do fluxograma indica o número de linha da Tabela 3-4, que contém uma descrição detalhada desses passos.

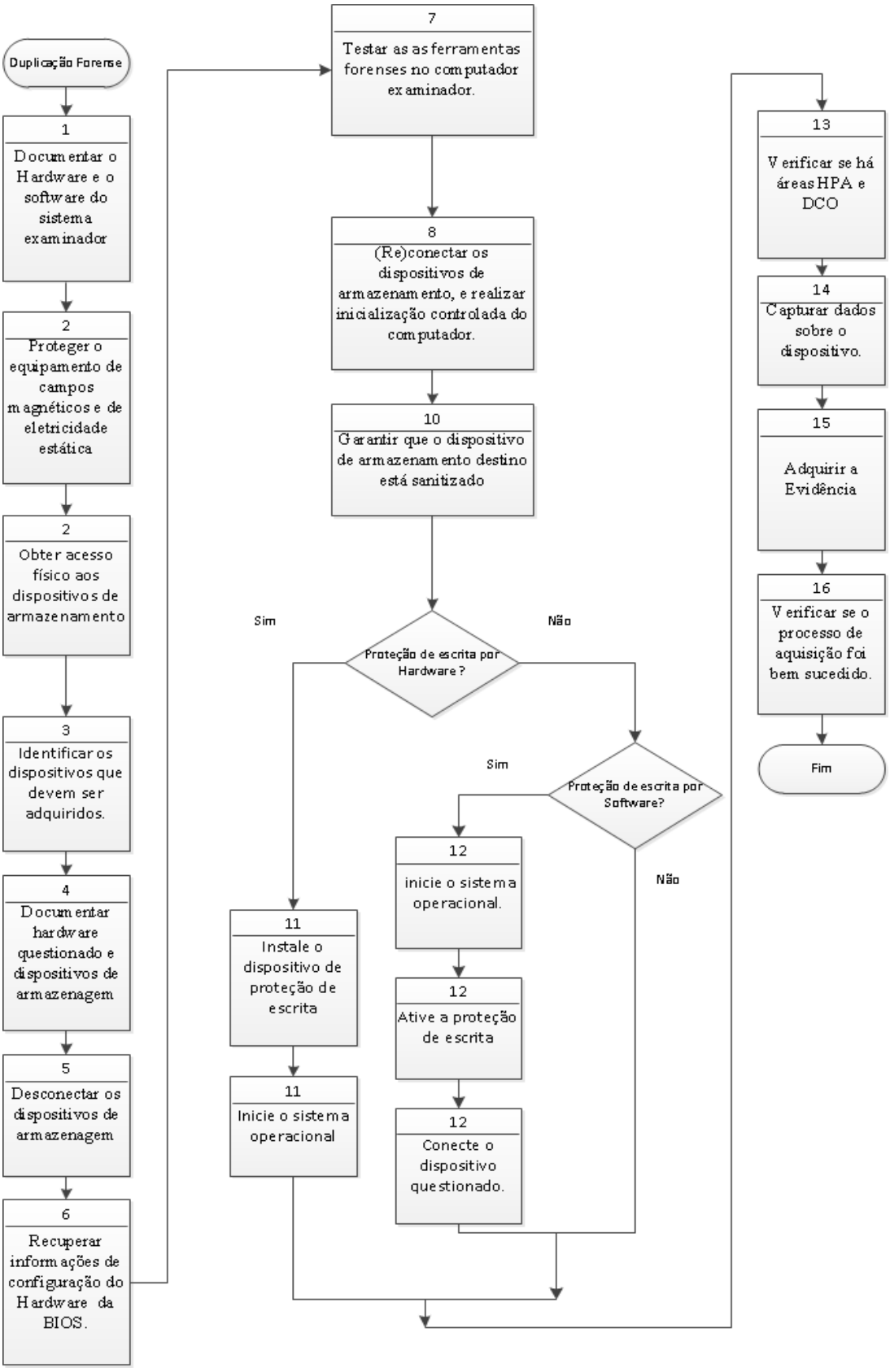

Figura 3-3: Fluxograma do Método do NIJ

Tabela 3-4: Método de Aquisição Digital por (Daniels D.J., Hart S.V., 2004)

#	Descrição
1	Documentar o Hardware e o software do sistema examinador;
2	Desmontar o gabinete do computador questionado para permitir acesso físico aos dispositivos de armazenamento. Tomar o cuidado de garantir que o equipamento está protegido de campos magnéticos e de eletricidade estática;
3	Identificar os dispositivos que devem ser adquiridos. Podem ser dispositivos internos, externos ou ambos;
4	Documentar a configuração do hardware questionado e dos dispositivos de armazenagem;
5	Desconectar os dispositivos de armazenagem para prevenir a destruição ou alteração dos dados;
6	Recuperar informações de configuração do equipamento questionado usando inicialização controlada. Isso inclui dados da CMOS/BIOS como sequência de boot, data e hora do sistema e se o sistema está protegido por senha;
7	Testar as funcionalidades do computador e do disco de boot onde estão as ferramentas forenses. Garantir que os dispositivos de armazenamento ainda estejam desconectados. Testar se o disco de inicialização com as ferramentas será devidamente inicializado;
8	Reconectar os dispositivos de armazenamento, e realizar inicialização controlada do computador, para verificar configurações dos drives;
9	Sempre que possível, realizar a aquisição da evidência usando o equipamento do examinador. Algumas circunstâncias podem resultar na decisão de não remover os dispositivos de aquisição, como por exemplo: discos em RAID, discos de netbooks e dependência de hardware legado;
10	Garantir que o dispositivo de armazenamento destino está sanitizado;
11	Se a proteção de escrita por hardware for utilizada, instale o dispositivo de proteção de escrita e inicie o sistema operacional;
12	Se a proteção de escrita por software for utilizada, inicie o sistema operacional, ative a proteção de escrita, e só depois conecte o dispositivo questionado;
13	Investigar a geometria dos dispositivos para garantir que todo o espaço declarado pelo fabricante está disponível no disco, incluindo áreas HPA e DCO;
14	Capturar o número serial eletrônico do dispositivo, entre outros dados;
15	Adquirir a evidência para o dispositivo destino usando hardware e software apropriados, como programas de duplicação, dispositivos de hardware dedicados, e programas de análise forense;
16	Verificar se o processo de aquisição foi bem sucedido comparando hashes do original e da cópia ou comparando-os bit-a-bit;

3.3. PASSOS ADICIONAIS AO MÉTODO

Tendo como ponto de partida os passos para aquisição de evidências, elaborado pelo NIJ, este capítulo mostra a aplicação do método nos experimentos realizados, e propõe passos adicionais a esse método. A Tabela 3-5 mostra o resultado da aplicação dos passos.

Tabela 3-5: Aplicação do Método de Duplicação Forense

#	Descrição
1	O Hardware utilizado é escrito na tabela A-1 do Anexo A. As ferramentas utilizadas são descritas na seção 2.5.3.
2	O ambiente de teste não usou disco rígido com evidências originais. O disco rígido de origem continha uma instalação do sistema operacional Windows XP, com dados diversos. A rede elétrica estava aterrada e estabilizada, e pulseiras anti-eletrostáticas foram utilizadas no manuseio dos equipamentos.
3	O dispositivo a ser adquirido está descrito no Anexo A, tabela A-2.
4	Os dispositivos de armazenagem estão descritos no Anexo A, tabela A-2.
5	Não se aplica, visto que os discos estavam previamente desconectados.

#	Descrição
6	A sequência de inicialização (boot) foi alterada para utilizar apenas boot pelo CD-ROM.
7	O CD-ROM contendo o sistema HELIX3 apresentou-se íntegro, através de teste executado automaticamente em sua inicialização.
8	Os dispositivos de armazenamento foram conectados, e a configuração da BIOS mostrou a descrição correta dos dispositivos.
9	A aquisição da evidência foi realizada usando o equipamento do examinador.
10	Foi realizada formatação lógica do dispositivo de armazenamento destino, para cada sistema de arquivo (FAT32, NTFS e EXT3), usando o utilitário Gparted. Depois de cada formatação, o disco foi sanitizado com a ferramenta DC3dd.
11	Não foi utilizada proteção de escrita por hardware.
12	Não foi utilizada proteção de escrita por software.
13	A geometria dos dispositivos descrita na BIOS foi comparada com as informações do fabricante (tabela A-2), não apresentando diferença.
14	Não foi possível capturar o número serial eletrônico do dispositivo.
15	Para cada experimento, a evidência foi adquirida para o dispositivo destino usando as ferramentas de duplicação indicadas na seção 2.5.2.
16	Em todos os experimentos, o processo de aquisição foi bem sucedido. A verificação foi realizada utilizando comparação bit-a-bit ou por comparação de hash, dependendo dos recursos oferecidos pelas ferramentas.

Após a realização dos experimentos, foi notada a necessidade de passos adicionais ao método estudado. A Tabela 3-6 mostra esses passos adicionais.

Tabela 3-6: Passos Adicionais ao Método de Duplicação Forense

Relativo à	Nro. Assertiva	Assertiva
Equipamento	1	A instalação elétrica do equipamento utilizado deve estar aterrada e estabilizada, com uma fonte de alimentação ininterrupta.
Ferramentas de Análise	2	O formato do arquivo de saída deve ser compatível com a ferramenta utilizada.
Equipamento suspeito	3	Em caso de uso da cópia para execução no equipamento suspeito ou em máquina virtual, os cuidados de isolamento devem ser observados.
Mídias	4	Em caso de uso de PC para cópia, com discos usando interfaces IDE/PATA, os discos origem e destino devem estar em portas IDE separadas.
	5	A velocidade máxima teórica da interface do disco destino deve ser maior ou igual a do disco origem.
	6	O resultado do cálculo de hashes por segmento deve ser armazenado em local apropriado.
	7	O sistema de arquivos de destino deve ser compatível com as ferramentas de análise usadas.
	8	O tamanho da partição destino deve ser maior que a capacidade total do disco de origem.
	9	O cálculo de hashes deve ser efetuado, e armazenado em local apropriado.

A partir dessas observações, foi desenvolvido o fluxograma mostrado na Figura 3-4 e na Figura 3-5.

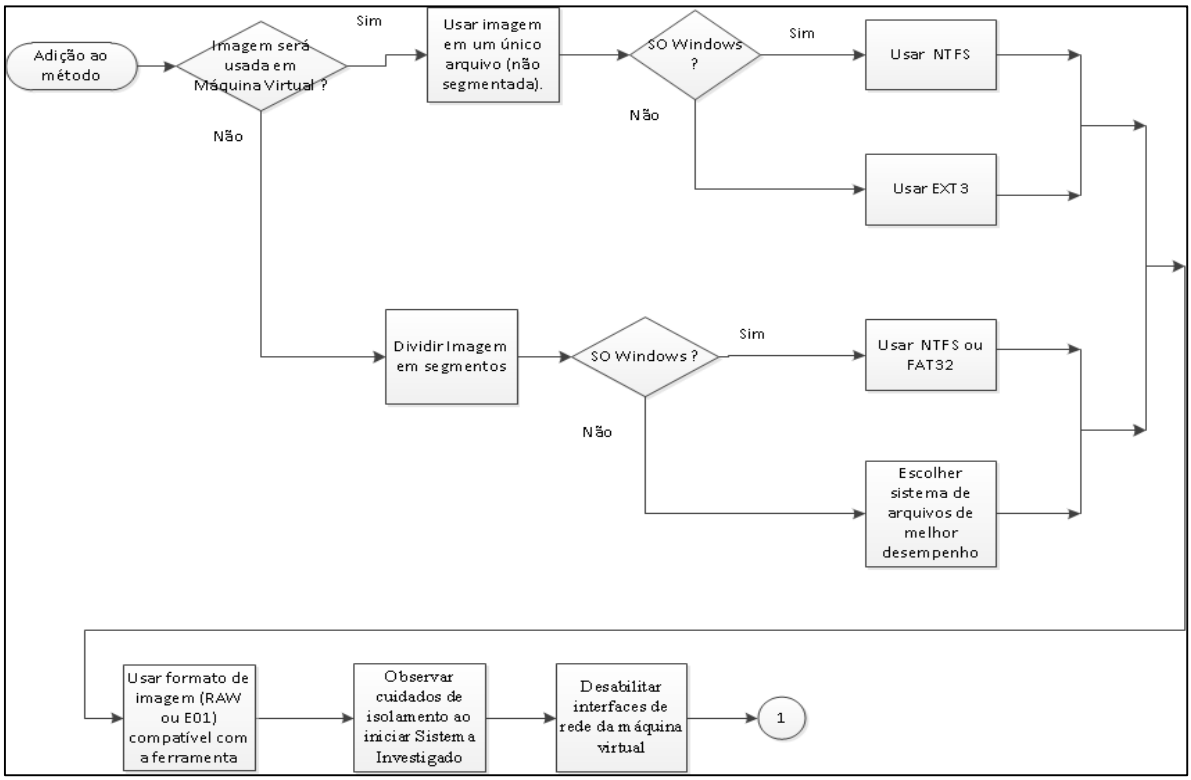

Figura 3-4: Fluxograma do Método Adicional Proposto (Parte 1)

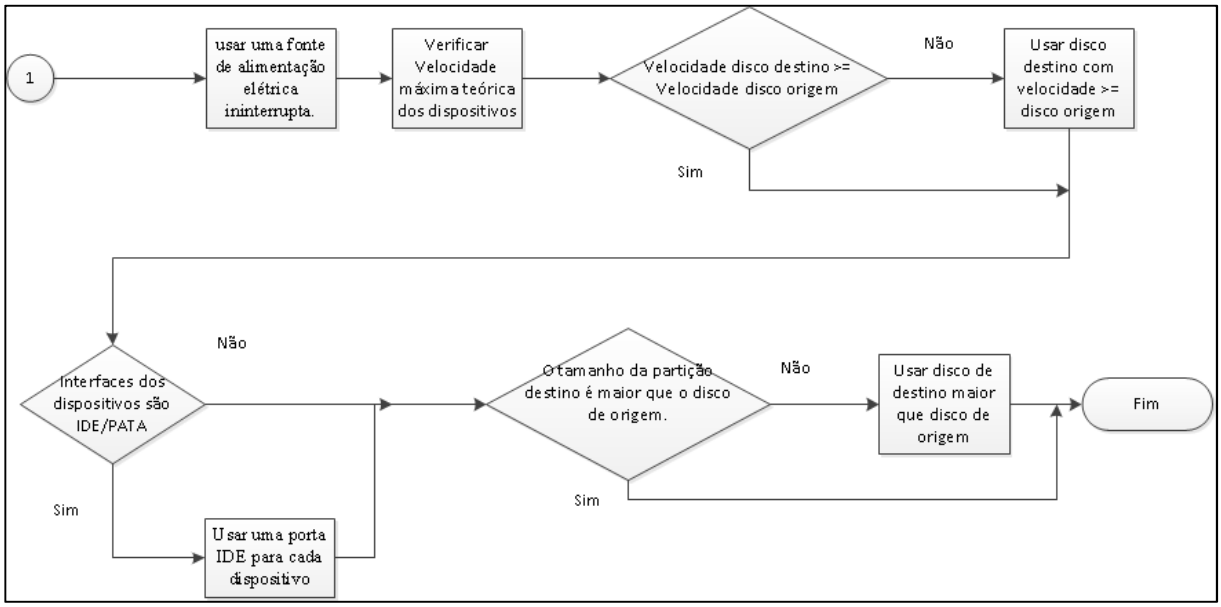

Figura 3-5: Fluxograma do Método Adicional Proposto (Parte 2)

Com o cenário descrito acima, foram realizados os experimentos descritos no capítulo 4.

3.4. FERRAMENTAS UTILIZADAS

Para alcançar o objetivo proposto por este estudo, de melhoria de desempenho no procedimento de duplicação forense, em conjunto com a proposta de método foram realizados experimentos de duplicação forense. Nestes experimentos, expostos no capítulo 4, foram usadas diferentes ferramentas. De acordo com (Costa, M.A.S.L., 2011), as ferramentas mais usadas na duplicação forense são o dd, o LinEn e o FTK Imager Lite. Entretanto, existem diversas outras ferramentas usadas para essa tarefa.

As ferramentas usadas nos testes foram escolhidas pela sua acessibilidade, flexibilidade e confiabilidade. Estas ferramentas foram encontradas em distribuições do sistema operacional Linux, específicas para análise forense, ou disponibilizadas gratuitamente pelos autores/fabricantes. Também disponibilizaram opções de parâmetros para configuração da duplicação forense. Além disso, a confiabilidade das ferramentas foi testada pelo NIST, e os resultados publicados em (*National Institute of Justice* [A], 2008) e (*National Institute of Justice* [B], 2008). Dentre outros recursos, as ferramentas precisavam mostrar os seguintes: não modificação do disco de origem; criação de imagem bit-a-bit do disco origem; aquisição eficaz de todos os setores do disco origem; e relatório de setores com erros.

As ferramentas foram reunidas em dois grupos, RAW e E01, com base no formato do arquivo de imagem gerado pela ferramenta. Para o grupo RAW foram usadas as ferramentas DCFLDD e o DC3DD. A ferramenta dd não foi considerada neste estudo, pois não realiza cálculo de hashes, função obrigatória para verificação de integridade da evidência de acordo com as metodologias estudadas. No grupo E01 foram usadas as ferramentas LinEn, FTKImager e EWFAcquire. Todos os experimentos foram executados por linha de comando. As seções abaixo apresentam as características de cada ferramenta e a sintaxe de seus comandos, e uma comparação entre elas é mostrada na Tabela 3-7.

Tabela 3-7: Comparação entre as Ferramentas de Aquisição

Ferramenta	Sistema Operacional	Suporte de Gravação em Sistemas de Arquivos	Cálculo de Hash	Compressão de Arquivo de Saída ?	Formato de Arquivo de Saída
DCFLDD	Linux Distribuição Helix	FAT32, EXT3,NTFS	Integral e Por Segmento	Não disponível	RAW integral ou com segmentação.
DC3DD	Linux	FAT32, EXT3,NTFS	Integral e por Segmento	Não disponível	RAW integral ou com segmentação.
Linen	Linux	FAT32, NTFS, EXT3	Integral	Sim	E01 integral ou com segmentação.
FTKImager	Linux	FAT32, NTFS, EXT3	Integral	Sim	E01 ou SMART, inteiro ou segmentado.
EWFAcquire	Linux	FAT32, NTFS, EXT3	Integral	Sim	E01 inteiro ou segmentado.

3.4.1. LinEn

A ferramenta LinEn é fornecido pela Guidance, e é uma versão Linux da ferramenta de aquisição para DOS do software **Encase**. Tem as mesmas funções básicas de espelhamento de dispositivos que a versão DOS, porém pode copiar para sistemas de arquivos Linux, FAT32 e NTFS, com uma maior velocidade de aquisição, pelo fato de o Linux ser um sistema de 64 bits, contra os 16 bits do DOS. O formato de saída é o formato proprietário Expert Witness, com extensão de arquivo "E0X", onde o X inicia em um, e é incrementado para cada segmento do arquivo destino.

O LinEn foi testado pelo programa CFTT (*Computer Forensic Testing Tools*), projeto conjunto do NIJ e NIST, e os resultados demonstram que está apto a ser usado para duplicação forense (*National Institute of Justice* [A], 2008).

O LinEn pode ser executado em uma interface gráfica, ou por linha de comando. Neste modo de execução o comando pode ser digitado inteiramente no prompt de comando, ou podem ser usados arquivos de configuração com os parâmetros desejados. Os parâmetros disponíveis no arquivo de configuração e a sintaxe do comando são mostrados na Tabela 3-8.

O parâmetro *blocksize* se refere ao tamanho do buffer usado para aquisição de dados e cálculo do CRC. A versão do Encase para DOS limita o *blocksize* em 64 setores. Já o Linen permite *blocksize* de até 512 setores. O Linen calcula um CRC para cada bloco. Quanto maior o

parâmetro blocksize, maior a velocidade de aquisição, pois um menor número de cálculos de CRC será efetuado.

Tabela 3-8: Parâmetros do Arquivo de Configuração do LinEn

Sintaxe de linha de comando do LinEn	
"linen –cl –j <caminho\nome do arquivo de configuração>"	
Parâmetro	**Descrição**
EvidencePath	Caminho e nome do arquivo de imagem a ser criado.
EvidenceName	Nome da evidência dentro do arquivo de imagem.
CaseNumber	Número de caso da evidência.
Examiner	Nome do Examinador.
EvidenceNumber	Número da evidência.
AlternatePath	Uma lista delimitada por ponto-e-vírgulas de caminhos alternativos para gravação do arquivo de imagem.
Notes	Notas.
MaxfileSize	Tamanho máximo de cada segmento do arquivo imagem.
Compress	Nível de compressão (0 = nenhum, 1 = rápido, 2 = melhor).
Granularity	granularidade de erro (em setores).
BlockSize	Setores por bloco para o arquivo de imagem.
Hash	Cálculo de hash MD5 Ligado (TRUE) ou desligado (FALSE).
SHA1	Cálculo de hash SHA1 Ligado (TRUE) ou desligado (FALSE).
Device	Dispositivo que será adquirido ou que terá seu hash calculado.
CommandLine	Sair de linha de comando se uma variável necessária não for preenchida (TRUE ou FALSE).
AcquireMode	Adquirir o dispositivo escolhido (TRUE ou FALSE).
HashMode	Calcular o hash do dispositivo escolhido (TRUE ou FALSE).

Em versões anteriores, se o Linen lesse um setor com erro, o bloco inteiro era desprezado, e um bloco zerado era colocado em seu lugar. Na versão usada, se um setor está com erro em um bloco de 64 setores, com granularidade 64, todos os setores do bloco são desprezados. Se a granularidade for Um, apenas o setor com erro é desprezado. Entretanto, o desempenho pode cair. Tipicamente, o valor de granularidade será 64, sendo usados valores menores apenas com discos rígidos com erros. O limite superior para esse parâmetro é o valor de *blocksize*.

Para detecção de áreas HPA e DCO em discos ATA-6 ou mais recentes, é necessário que a distribuição Linux onde esteja sendo executado o Linen use modo ATA direto, ou que um módulo específico do Encase para Windows chamado Fastblock SE seja adquirido.

3.4.2. FTKImager

O Forensic Toolkit Imager (FTK Imager) é um pacote de software comercial para criação de imagens forenses distribuído pela AccessData. O FTK Imager suporta o armazenamento de imagens de disco em no formato E01, SMART , bem como em formato RAW (dd). Usa a tecnologia Isobuster para gerar imagens de CDs para arquivos em formato ISO/CUE.

Não oferece customização de tamanho de setores por bloco, usando sempre o valor de 512 setores por bloco. Entretanto, foi escolhida para os testes por ser usada em vários órgãos de perícia oficiais. A versão usada foi a "2.9 100407".

Pode ser usado para aquisição de dados em sistemas Windows em conjunto com bloqueadores de escrita. No Linux, é usado com linha de comando, com sintaxe apresentada nas Tabelas 3-9 e 3-10.

Tabela 3-9: Sintaxe de linha de comando do FTKIMAGER

SINTAXE DE LINHA DE COMANDO DO FTKIMAGER
ftkimager origem [arquivo_destino] [opções] Origem pode ser um dispositivo, um arquivo de imagem ou "–" para stdin. Se arquivo_destino for especificado, a extensão apropriada (de acordo com o format) será adicionada ao nome. Se arquivo_destino for "-" ou não especificado, os dados brutos serão escritos em stdout.

Tabela 3-10: Opções de linha de comando do FTKIMAGER

Opções	
help	Listar informações de ajuda.
list-drives	Mostrar os dispositivos físicos detectados.
verify	Calcular Hash e verificar a imagem destino, ou imagem de origem se nenhum destino for especificado
print-info	Mostra informações sobre um dispositivo ou imagem, e sai do programa.
quiet	Não mostra informação de progresso de criação/verificação.

Opções	
no-sha1	Não calcular hash SHA1 durante a aquisição ou verificação.
s01	Cria uma imagem comprimida no formato SMART.[2]
e01	Cria uma imagem no formato E01.[2]
frag x{K|M|G|T}	Cria imagem com fragmentos de no máximo x {Kbytes, Megabytes, GigaBytes ou Terabytes} de tamanho. Também aceita {kB| MB|GB|TB}, para potência de 10 (1kB=1000 bytes) em vez de 2 (1K=1024 bytes).[2]
compress C	Nível de compressão C (0=nenhuma, 1=rápida, ..., 9=máxima). [2]
E01/smart metadata	Metadados adicionais para formato E01/SMART. [2] (Usar aspas se o X contém espaços) case-number X evidence-number X description X examiner X notes X
inpass P	Decriptografa arquivo origem usando a senha P.[3]
incert C [P]	Decriptografa arquivo origem usando o certificado C com a senha P. [3]
outpass P	Criptografa arquivo destino usando a senha P.[3]
outcert C [P]	Criptografa arquivo destino usando o certificado C com a senha P.[3]

3.4.3. EWFAcquire

O EWFacquire é um utilitário para adquirir dados uma fonte digital e armazená-lo em formato EWF (Expert Witness compactado). O EWFacquire adquire dados no formato equivalente ao EnCase e FTK Imager, incluindo metadados. Também pode converter uma imagem em formato RAW para o formato EWF.

Tabela 3-11: Sintaxe de linha de Comando EWFAcquire

Sintaxe do Comando
ewfacquire [-A codepage] [-b amount_of_sectors] [-B amount_of_bytes] [-c compression_type] [-C case_number] [-d digest_type] [-D description] [-e examiner_name] [-E evidence_number] [-f format] [-g amount_of_sectors] [-l log_filename] [-m media_type] [-M media_flags] [-N notes] [-o offset] [-p process_buffer_size] [-P bytes_per_sector] [-r read_error_retries] [-S segment_file_size] [-t target] [-2 secondary_target] [-hqRsuvVw] source

O EWFacquire é parte do pacote libewf, uma biblioteca de funções que dá suporte ao formato EWF. A versão usada é a de número 20100119. A sintaxe de linha de comando da ferramenta mostrada na Tabela 3-11 e a descrição dos parâmetros é apresentada na Tabela 3-12.

[2] Opção válida apenas quando "arquivo_destino" é especificado.
[3] Opção para criação de imagens criptogradas AD.

A ferramenta lê arquivo ou dispositivo, até que encontra um erro de leitura. Nesse caso, irá repetir a leitura uma quantidade de vezes especificada. Se ainda for incapaz de ler e, se especificado, ele vai zerar (limpar) o restante da quantidade de setores especificado no parâmetro granularidade de erro. Se for especificado que deve realizar o mesmo procedimento do Encase, irá limpar todos os setores especificado em granularidade de erro.

A compressão de blocos vazios detecta blocos de setores exatamento com os mesmos dados e os comprime usando o nível de compressão padrão. Um dos parâmetros de formato utilizados, o "encase6", permite segmentos do arquivo de imagem maiores que 2 GigaBytes.

Tabela 3-12: Parâmetros do EWFAcquire

Parâmetro	Descrição
-A codepage	A página de código da seção de cabeçalho : ascii (default), windows-874, windows-1250, windows-1251, windows-1252, windows-1253, windows-1254, windows-1255, windows-1256, windows-1257, windows-1258
-b amount_of_sectors	Quantidade de setores a serem lidos de uma vez: 64 (default), 128, 256, 512, 1024, 2048, 4096, 8192, 16384 ou 32768
-B amount_of_bytes	Quantidade de bytes a ser adquirida.
-c compression_type	Tipo de compressão: none (default), empty-block, fast, best
-C case_number	Número do caso (metadado)
-d digest_type	Calcular hashes além do md5: sha1
-D description	Descrição (metadado)
-e examiner_name	Nome do examinador
-E evidence_number	Número da evidência
-f format	Formato do arquivo de saída a ser usado: ewf, smart, ftk, encase1, encase2, encase3, encase4, encase5, encase6 (default), linen5, linen6, ewfx.
-g amount_of_sectors	Número de setores para granularidade de erro: 64 (default), 128, 256, 512, 1024, 2048, 4096, 8192, 16384 or 32768
-h	Ajuda sobre os comandos.
-l log_filename	Grava erros de aquisição e hashes para o arquivo log_ filename.
-m media_type	Tipo de mídia: fixed (default), removable, optical, memory
-M media_flags	Flags de mídia: logical, physical (default)
-N notes	Notas do Caso
-o offset	Offset para início da aquisição (default é 0)
-p process_buffer_size	Tamanho do Buffer do processo (default é o tamanho do bloco)
-P bytes_per_sector	Quantidade de bytes por setor (default é 512) (usado para definir um tamanho de setor diferente da detecção automática)
-q	Não mostra informações de progressão da aquisição
-r read_error_retries	Número de tentativas de leitura quando ocorre um erro (default é 2)
-R	Reinicia a aquisição do ultimo ponto de parada.
-s	Troca pares de bytes dos dados (de AB para BA). Usado para converter de big endian para little endian e vice-versa.

Parâmetro	Descrição
-S segment_file_size	Tamanho de segment do arquivo em bytes (default é 1.4 GB). O mínimo é 1.0 MB, o máximo é 7.9 EB para o formato encase6 e 1.9 GB para outros formatos.
-t target	O arquivo destino a ser escrito (sem extensão).
-u	Modo não interativo.
-v	Envia informações da execução para stderr.
-V	Imprime a versão do programa.
-w	Grava zeros nos setores com erro de leitura (imita o tratamento de erros do EnCase).
-2 secondary_target	Arquivo alternativo para gravar.

3.4.4. DCFLDD

O DCFLDD é uma ferramenta com características úteis para análise forense e segurança. Foi criada pelo Laboratório de Computação Forense (Defense Computer Forensics Laboratory - DCFL), do Centro de Crimes Cibernéticos (DC3), que fornece apoio em forense digital ao Departamento de Defesa dos Estados Unidos. É baseado no programa dd encontrado no pacote GNU Coreutils, porém apresenta recursos voltados à área forense, como os mostrados na Tabela 3-13. A sintaxe do comando e suas opções são mostradas na Tabela 3-14.

Tabela 3-13: Recursos Forenses do DCFLDD

Característica	Descrição
Hashing durante a aquisição	Como o Linen, o dcfldd pode calcular o hash dos dados de entrada no momento que são copiados. Mas possibilita execução de cálculo de hashes de segmentos do arquivo de origem, em bytes, com tamanho independente do tamanho do segmento.
Saída de estado	Diferente do dd, o dcfldd pode atualizar o usuário do progresso da execução da cópia em termos da quantidade de dados transferidos e tempo estimado da operação.
Wipe de disco	Pode ser usado para limpar os discos de forma rápida e com um padrão escolhido.
Verificação de cópia	Verifica se a cópia bit-a-bit foi efetuada com sucesso.
Saídas múltiplas	Pode gravar em vários arquivos ou discos ao mesmo tempo.
Saída segmentada (Split)	Pode dividir a saída em vários arquivos.
Gravação de logs	Pode enviar os seus dados de log e de saída para outros comandos, através do uso de pipes, bem como para arquivos.

A versão usada foi a 1.3.4-1. O DCFLDD usa o formato de saída RAW, com os resultados de erros e de cálculos *hash* gravados em arquivos separados.

Tabela 3-14: Sintaxe do DCFLDD

Opções	Descrição
bs=BYTES	Força ibs = BYTES e obs = BYTES.

Opções	Descrição
cbs=BYTES	Converter BYTES bytes de cada vez.
conv=KEYWORDS	Converter o arquivo de acordo com a lista de palavras separadas por vírgulas.
count=BLOCKS	Copiar apenas BLOCKS blocos de entrada.
ibs=BYTES	Lê BYTES bytes de cada vez.
if=FILE	Lê do FILE em vez de stdin.
obs=BYTES	Escreve BYTES bytes de cada vez.
of=FILE	Escreve em FILE em vez de stdout. Pode ser usado várias vezes para escrever a saída para vários arquivos simultaneamente.
of=COMMAND	Executar e gravar a saída no processo COMMAND.
seek=BLOCKS	Saltar BLOCKS blocos de tamanho obs no início da produção.
skip=BLOCKS	Saltar BLOCKS blocos de tamanho ibs no início da entrada.
pattern=HEX	Usar o padrão binário especificado como entrada.
textpattern=TEXT	TEXT uso repetindo como entrada.
errlog=FILE	Enviar mensagens de erro para arquivo FILE, bem como para stderr.
hashwindow=BYTES	Executar um hash em cada quantidade de BYTES de dados.
hash=NAME	NAME pode ser md5, sha1, SHA256, SHA384 ou SHA512. O algoritmo padrão é md5. Para selecionar múltiplos algoritmos para executar simultaneamente, introduzir os nomes em uma lista separada por vírgulas.
hashlog=FILE	Enviar a saída do hash MD5 em FILE em vez de stderr. Se você estiver usando vários algoritmos de hash pode enviar cada um para um arquivo separado usando a convenção algorithmlog=FILE, por exemplo md5log = FILE1, sha1log = FILE2, etc.
hashlog=COMMAND	Executar e escrever hashlog no processo COMMAND.
hashconv=[before|after]	Executar o hashing antes ou após as conversões.
hashformat=FORMAT	Exibir cada hash de acordo com o formato de hash específico do DCFLDD.
totalhashformat=FORMAT	Exibir o valor total de hash acordo com o formato.
status=[on|off]	Exibir uma mensagem de status contínuo sobre a execução. O padrão é stderr "on".
statusinterval=N	Atualizar a mensagem de status a cada N blcos. O valor padrão é de 256 blocos.
sizeprobe=[if|of]	Determinar o tamanho do arquivo de entrada ou de saída para uso com mensagens de status. atenção: não use esta opção contra um dispositivo de fita.
split=BYTES	Escrever cada quantidade de dados para um novo arquivo. Essa operação se aplica a qualquer um file que se segue.
splitformat=TEXT	O formato de extensão de arquivo para operação de divisão. Pode ser usado qualquer número de 'a' ou 'n' em qualquer combinação. O formato padrão é "nnn". As opções split e splitformat só produz efeitos para os arquivos de saída especificados após, estas opções aparecem na linha de comando. Da mesma forma, você pode especificar essas opções diversas vezes para arquivos de saída diferentes dentro da mesma linha de comando.
vf=FILE	Verificar que o arquivo corresponde a entrada especificada.
verifylog=FILE	Enviar os resultados de verificação em FILE, em vez de stderr.
verifylog=COMMAND	Executar e escrever os resultados de verificação para o processo COMMAND.

3.4.5. DC3DD

A ferramenta DC3DD é uma atualização da ferramenta DCCIDD, com os recursos adicionais de divisão do arquivo de imagem em segmentos e cálculo de *hashes* MD5, SHA-1 e SHA-256 em paralelo. A versão utilizada foi a 7.1.614. As sintaxes dos comandos apresentam algumas diferenças em relação ao DCFLDD, mas suas funcionalidades são semelhantes. Entretanto, foi encontrada apenas a avaliação da funcionalidade de sanitização do disco (*wipe*) da ferramenta. Não foram encontradas avaliações da ferramenta no procedimento de duplicação forense, apenas do DCCIDD. A linha de comando do DC3DD tem a seguinte sintaxe:

dc3dd if=/dev/sdc bufsz=256K ssz=256K verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-512.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-512.log

Os principais parâmetros do comando são mostrados na Tabela 3-15.

Tabela 3-15: Sintaxe dos principais parâmetros do DC3DD

Comando	**Função**
if = dispositivo ou arquivo	Definir o dispositivo ou arquivo de entrada.
of = arquivo ou dispositivo	Escreve a saída para um arquivo ou dispositivo. Este opção pode ser usada mais de uma vez, para gerar saídas múltiplas.
ofs = BASE.FMT	Escreve a saída para um conjunto de arquivos com nome de base BASE e extensões de nome de arquivos sequenciais gerados a partir do especificador de formato FMT
ofsz = Bytes	Define o tamanho máximo de cada arquivo em imagens segmentadas.
hash = ALGORITMO	Cálculo de hash da entrada, onde ALGORITMO pode ser md5, sha1, sha256, ou SHA512. Esta opção pode ser usada uma vez para cada algoritmo suportado.
log = arquivo_de_log	Escreve estatísticas de E/S, erros de leitura e hashes no arquivo arquivo_de_log statistcs, diagnósticos e hashes totais de entrada e saída para FILE.
verb=on	Ativar relatório de execução.
rec=off	Por padrão, Zeros (0) são escritos para a saída em lugar de setores defeituosos, quando a entrada é um dispositivo. Use esta opção para fazer com que o programa pare a execução quando um setor defeituoso for encontrado.
wipe = DEVICE	Sanitiza o dispositivo escrevendo zeros (padrão) ou um padrão especificado.
bufsz = BYTES	Define o tamanho dos buffers para BYTES. Isto define o número máximo de bytes que pode ser lido de uma vez a partir da entrada. BYTES deve ser múltiplo do tamanho de setor. Utilize esta opção para ajustar o desempenho.
SSZ = BYTES	Define o número de BYTES para o tamanho do setor. Se SSZ = não for especificado, o tamanho do setor é determinado pela sondagem do dispositivo; Se a detecção falhar ou o alvo não é um dispositivo, um setor de 512 bytes é assumido.

O DC3DD grava imagem no formato RAW, com os resultados de erros e de cálculos *hash* gravados em um único arquivo de registro (log).

4. EXPERIMENTOS

Os experimentos realizados têm como objetivo avaliar o desempenho dos procedimentos de duplicação forense em diferentes cenários. Para isso, foram executadas duplicações forenses de um disco origem para um disco destino, usando as ferramentas escolhidas LinEn, FTKImager e EWFACQUIRE para arquivos no formato Expert Witness (E01), e DCFLDD e DC3DD, para formato RAW, variando o número de setores por bloco, para os sistemas de arquivo destino FAT32, NTFS e EXT3.

Com esses resultados espera-se determinar os cenários de melhor desempenho. Também se espera determinar a influência da natureza dos dados de origem sobre a velocidade do procedimento.

Como visto no item 3.1.2, a aquisição de todos os setores do disco na camada de abstração é a regra indicada por (Carrier, B., 2005), para que haja maiores possibilidades de encontrar as evidências procuradas. Assim, o uso de cópia lógica e de cópia de subconjuntos de arquivos da evidência foi desconsiderado neste estudo.

Entretanto, há formatos de arquivos de imagem do tipo incorporado que permitem a criação de arquivos de imagem comprimidos. Para esses cenários, onde há a possibilidade de variação de desempenho causado pela variação dos dados de origem, foram realizados testes com disco de origem preenchido com valor "1" (através de *wipe* de disco), garantindo homogeneidade dos dados, e também com o disco de origem criptografado após o preenchimento com "1".

Primeiramente, foi aplicado o método de duplicação forense proposto no capítulo 3. Os testes foram então definidos e executados.

Para organização dos testes, foi usado um subconjunto da metodologia para testes de ferramentas de duplicação forense, definida pelo CFTT/NIST em (CFTT/NIST, 2005). Essa metodologia define requerimentos para cada ferramenta, assertivas de teste e casos de teste. Os requerimentos definem comportamentos esperados de cada ferramenta, e as assertivas são usadas para verificar se esses comportamentos foram alcançados após a execução do caso de teste.

A medição do tempo do procedimento utilizou o comando *time* do Linux. A saída padrão do comando time mostra as informações *real time* (tempo real), *user time* (tempo de usuário) e *sys time* (tempo de sistema). O tempo real, obtido com o argumento do comando "%E",

mostra o tempo total entre a invocação do comando e seu término. O tempo de usuário, "%U", consiste no gasto de tempo de processos de usuário. Esse tempo aumenta nos testes onde há compressão do arquivo de destino. O tempo de sistema, "%S", consiste no tempo gasto por processos de sistema.

Nos testes em que as ferramentas informavam o tempo de execução, esse valor também foi anotado. Por último, foram analisados os resultados dessas execuções.

Foram usadas siglas na descrição dos elementos dos testes, descritas na Tabela 4-1. As ferramentas devem ser executadas em um ambiente de execução (XE – *execution environment*). A evidência original (DS – *Digital Source*) deve estar contida em um dispositivo de dados com interface de acesso (SRC-AI – *Source Access Interface*). A configuração do destino da cópia inclui os parâmetros Sistema de Arquivos (FS – file system), Tipo de Arquivo Destino (DFT – *Destination File Type*) e interface do dispositivo destino (DST-AI – *Destination Access Interface*). Para cada teste, o número de setores por bloco (BLK-SZ – *Block Size*) vai assumir os valores 64, 128, 256 e 512.

Tabela 4-1: Siglas Usadas nos Testes

Sigla	Significado	Descrição	Valores
XE	Execution Enviroment	Ambiente de execução usado para o teste.	"HELIX", "BACKTRACK"
TT	Tool	Ferramenta usada.	"DC3DD", "DCFLDD", "LinEn", "Ewfimager","FTKImager"
DS	Digital Source	Tipo do dispositivo de Origem	"HD" (Hard Disk)
SRC-AI	Source Access Interface	Interface de acesso do dispositivo de origem.	"PATA", "SATA"
FS	File System	Sistema de Arquivos destino.	"FAT32", "EXT3", "NTFS"
DST-AI	Destination Access Interface	Interface de acesso do dispositivo destino.	"PATA", "SATA"
BLK-SZ	Block Size	Tamanho do Bloco.	"64", "128", "256", "512"

As assertivas de teste são afirmações gerais sobre condições verificadas após a execução do teste. Neste estudo, elas foram usadas para descrever objetivamente o comportamento das ferramentas após cada teste. A lista integral de assertivas de teste propostas em (CFTT/NIST, 2005) é apresentada na tabela B-1 do Anexo B, enquanto que os casos de teste propostos no mesmo documento são mostrados na tabela C-1 do Anexo C. Nem todos os casos de teste

listados nesta tabela são necessários para este estudo. Assim, a seleção de casos de teste executados foi baseada nos recursos oferecidos pelas ferramentas e nos resultados buscados.

Tabela 4-2: Casos de teste Selecionados

Identificação do Teste	Descrição do Teste.
DA-06	Adquirir um dispositivo físico utilizando interface de acesso AI para um arquivo de imagem.
DA-10	Adquirir uma fonte digital para um arquivo de imagem em um formato alternativo.

Os casos de teste selecionados são mostrados na Tabela 4-2. Os testes DA-06 realizam uma duplicação forense para um arquivo de imagem, em um formato definido pela ferramenta usada. Assim, o DC3DD e o DCFLDD geram imagens no formato RAW, enquanto o LinEn, FTKImager e EWFAcquire usam o formato Expert Witness (E01). Os testes DA-10 são executados nos cenários em que as ferramentas disponibilizam gravação em formato alternativo, que nestes testes é o formato comprimido. A Tabela 4-3 mostra as seleções de testes para cada ferramenta.

Tabela 4-3: Casos de teste Selecionados para cada Ferramenta

Número do teste	**DC3DD**	**DCFLDD**	**LinEn**	**FTKImager**	**EWFAcquire**
DA-06	•	•	•	•	•
DA-10			•	•	•
DA-06-(...)-CRIPTO	•	•	•	•	•
DA-10-(...)-CRIPTO			•	•	•

Os resultados de execução dos testes são descritos no formato da Tabela 4-4. A identificação dos testes segue a nomenclatura “DA-XX-T-FS-TB”, onde DA é a sigla para *Disk Acquisition*, “XX” é o número do teste correspondente à tabela 4-2, “T” é a ferramenta utilizada, “FS” é o tipo do sistema de arquivos destino e “TB” é o tamanho do bloco (em setores). Para os testes DA-10, foi adicionado o sufixo “CRIPTO”, indicando que o dispositivo de origem estava criptografado.

Tabela 4-4: Formato de Descrição dos Testes Executados

Item	Descrição
Identificação do Teste	Identificador único para o teste.
Resumo do Teste	Breve descrição do teste
Comentário	Informações adicionais sobre o teste
Ferramentas necessárias	Lista de ferramentas necessárias para o teste
Dispositivos	Identificação dos dispositivos usados segundo tabela de dispositivos do Anexo A.
Configuração do Dispositivo de Origem	Informações sobre dispositivo de origem.
Configuração do Dispositivo de Origem	Informações sobre dispositivo de destino.
Informações de Registros	Dados relevantes dos registros (logs) de execução dos testes.
Preparação do Ambiente	Ações de preparação do ambiente de execução, anteriores à execução dos experimentos.
Comando Utilizado	Comando utilizado para executar o teste.
Variações	Para testes que são repetidos com variação de algum parâmetro, este item descreve as variações que são feitas.
Resultados Obtidos	Descrição dos resultados dos testes.

A Tabela 4-5 apresenta um resumo com a identificação dos testes e suas descrições.

Tabela 4-5: Identificação e Descrição dos Testes Executados

Identificação do Teste	Descrição
DA-06	Duplicação Forense usando DC3DD, DCFLDD, LINEN, FTKIMAGER e EWFIMAGER, com sistemas de arquivos destino FAT32, EXT3 E NTFS, com variação do número de setores por bloco (64, 128, 256 ou 512) e disco origem preenchido com valor “1”.
DA-06-(...)-CRIPTO	Duplicação Forense usando DC3DD, DCFLDD, LINEN, FTKIMAGER e EWFIMAGER, com sistemas de arquivos destino NTFS, 512 setores por bloco e disco origem criptografado.
DA-10	Duplicação Forense usando LINEN, FTKIMAGER e EWFIMAGER, com sistemas de arquivos destino NTFS, com 512 setores por bloco, disco origem preenchido com valor “1”, com imagem de destino comprimida.
DA-10-(...)-CRIPTO	Duplicação Forense usando LINEN, FTKIMAGER e EWFIMAGER, com sistemas de arquivos destino NTFS, com 512 setores por bloco, disco origem criptografado e imagem de destino comprimida.

A descrição detalhada dos testes executados, seus parâmetros e resultados são apresentados no Anexo D, enquanto que a Tabela 4-6 apresenta os parâmetros dos testes.

Tabela 4-6: Parâmetros e Resultados de Tempo dos Casos de Testes

Identificação do Teste	Tamanho do Disco Origem (GB)	Ferramenta	Sistema de Arquivos	Setores por Bloco	Tempo Total
DA-06-DC3DD-FAT32-64	20	DC3DD	FAT32	64	14m44.176s
DA-06-DC3DD-FAT32-128	20	DC3DD	FAT32	128	14m43.249s
DA-06-DC3DD-FAT32-256	20	DC3DD	FAT32	256	14m43.213
DA-06-DC3DD-FAT32-512	20	DC3DD	FAT32	512	14m43.254s
DA-06-DC3DD-EXT3-64	20	DC3DD	EXT3	64	14m48.367s
DA-06-DC3DD-EXT3-128	20	DC3DD	EXT3	128	14m44.258s
DA-06-DC3DD-EXT3-256	20	DC3DD	EXT3	256	14m43.564s
DA-06-DC3DD-EXT3-512	20	DC3DD	EXT3	512	14m43.263s
DA-06-DC3DD-NTFS-64	20	DC3DD	NTFS	64	15m0.380s
DA-06-DC3DD-NTFS-128	20	DC3DD	NTFS	128	14m53.461s
DA-06-DC3DD-NTFS-256	20	DC3DD	NTFS	256	14m48.993s
DA-06-DC3DD-NTFS-512	20	DC3DD	NTFS	512	14m43.219s
DA-06-DCFLDD-FAT32-64	20	DCFLDD	FAT32	64	14m43.973s
DA-06-DCFLDD-FAT32-128	20	DCFLDD	FAT32	128	14m44.476s
DA-06-DCFLDD-FAT32-256	20	DCFLDD	FAT32	256	14m44.305s
DA-06-DCFLDD-FAT32-512	20	DCFLDD	FAT32	512	14m44.122s
DA-06-DCFLDD-EXT3-64	20	DCFLDD	EXT3	64	14m50.445
DA-06-DCFLDD-EXT3-128	20	DCFLDD	EXT3	128	14m50.958
DA-06-DCFLDD-EXT3-256	20	DCFLDD	EXT3	256	14m50.473
DA-06-DCFLDD-EXT3-512	20	DCFLDD	EXT3	512	14m50.059
DA-06-DCFLDD-NTFS-64	20	DCFLDD	NTFS	64	14m58.092
DA-06-DCFLDD-NTFS-128	20	DCFLDD	NTFS	128	14m59.678
DA-06-DCFLDD-NTFS-256	20	DCFLDD	NTFS	256	14m59.250
DA-06-DCFLDD-NTFS-512	20	DCFLDD	NTFS	512	15m1.893
DA-06-LINEN-FAT32-64	20	LINEN	FAT32	64	20m38.704
DA-06-LINEN-FAT32-128	20	LINEN	FAT32	128	20m21.383
DA-06-LINEN-FAT32-256	20	LINEN	FAT32	256	19m37.951
DA-06-LINEN-FAT32-512	20	LINEN	FAT32	512	18m33.995
DA-06-LINEN-EXT3-64	20	LINEN	EXT3	64	21m42.437
DA-06-LINEN-EXT3-128	20	LINEN	EXT3	128	22m0.770
DA-06-LINEN-EXT3-256	20	LINEN	EXT3	256	21m46.045
DA-06-LINEN-EXT3-512	20	LINEN	EXT3	512	21m49.603
DA-06-LINEN-NTFS-64	20	LINEN	NTFS	64	37m55.866
DA-06-LINEN-NTFS-128	20	LINEN	NTFS	128	37m11.767
DA-06-LINEN-NTFS-256	20	LINEN	NTFS	256	36m37.453
DA-06-LINEN-NTFS-512	20	LINEN	NTFS	512	36m13.545
DA-06-EWFACQUIRE-FAT32-64	20	EWFACQUIRE	FAT32	64	15m5.090

Identificação do Teste	Tamanho do Disco Origem (GB)	Ferramenta	Sistema de Arquivos	Setores por Bloco	Tempo Total
DA-06-EWFACQUIRE-FAT32-128	20	EWFACQUIRE	FAT32	128	15m7.474
DA-06-EWFACQUIRE-FAT32-256	20	EWFACQUIRE	FAT32	256	15m5.987
DA-06-EWFACQUIRE-FAT32-512	20	EWFACQUIRE	FAT32	512	15m4.259
DA-06-EWFACQUIRE-EXT3-64	20	EWFACQUIRE	EXT3	64	15m8.337
DA-06-EWFACQUIRE-EXT3-128	20	EWFACQUIRE	EXT3	128	19m15.848
DA-06-EWFACQUIRE-EXT3-256	20	EWFACQUIRE	EXT3	256	15m14.505
DA-06-EWFACQUIRE-EXT3-512	20	EWFACQUIRE	EXT3	512	15m13.119
DA-06-EWFACQUIRE-NTFS-64	20	EWFACQUIRE	NTFS	64	15m18.803
DA-06-EWFACQUIRE-NTFS-128	20	EWFACQUIRE	NTFS	128	15m22.537
DA-06-EWFACQUIRE-NTFS-256	20	EWFACQUIRE	NTFS	256	15m14.483
DA-06-EWFACQUIRE-NTFS-512	20	EWFACQUIRE	NTFS	512	15m9.703
DA-06-FTKIMAGER-FAT32-512	20	FTKIMAGER	FAT32	512	19m6.939
DA-06-FTKIMAGER-EXT3-512	20	FTKIMAGER	EXT3	512	15m3.345s
DA-06-FTKIMAGER-NTFS-512	20	FTKIMAGER	NTFS	512	15m3.303s
DA-10-LINEN-NTFS-512	20	LINEN	NTFS	512	36m13.545s
DA-10-EWFACQUIRE-NTFS-512	20	EWFACQUIRE	NTFS	512	14m57.292s
DA-10-FTKIMAGER-NTFS-512	20	FTKIMAGER	NTFS	512	14m58.541s
DA-06-DC3DD-NTFS-512-CRIPTO	20	DC3DD	NTFS	512	14m43.171s
DA-06-DCFLDD-NTFS-512-CRIPTO	20	DCFLDD	NTFS	512	15m7.720s
DA-06-LINEN-NTFS-512-CRIPTO	20	LINEN	NTFS	512	36m11.694s
DA-10-LINEN-NTFS-512-CRIPTO	20	LINEN	NTFS	512	36m56.864s
DA-06-FTKIMAGER-NTFS-512-CRIPTO	20	FTKIMAGER	NTFS	512	15m3.620s
DA-10-FTKIMAGER-NTFS-512-CRIPTO	20	FTKIMAGER	NTFS	512	14m59.158s
DA-06-EWFACQUIRE-NTFS-512-CRIPTO	20	EWFACQUIRE	NTFS	512	15m24.272s
DA-10-EWFACQUIRE-NTFS-512-CRIPTO	20	EWFACQUIRE	NTFS	512	21m22.619s

O fluxograma com a sequência de execução dos testes é mostrado na Figura 4-1.

Levando-se em conta os tamanhos de segmentos utilizados por padrão nas ferramentas utilizadas, foram usados segmentos de arquivo de imagem com tamanho de 2000 MB.

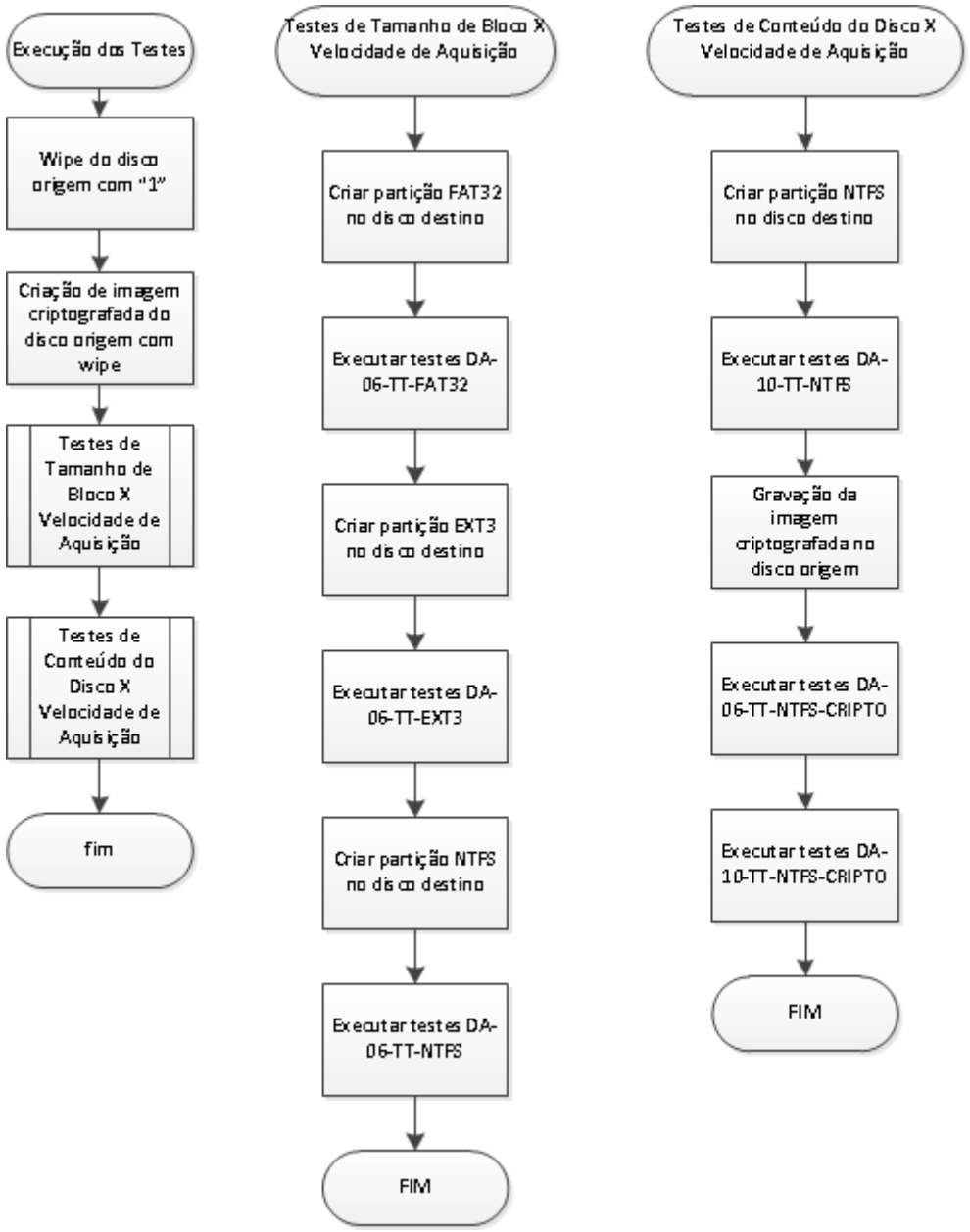

Figura 4-1: Sequência de Execução dos Testes

4.1. CAPACIDADE DOS DISCOS RÍGIDOS

Os casos de teste foram executados utilizando um disco rígido de origem com tamanho de 20 GB, identificado na tabela A-2 do Anexo A como 01-PATA. Esse tamanho é bem menor que a média de tamanho dos discos rígidos processados pelo Instituto-Geral de Perícias em 2011, de 198,12 GB, mostrado da Figura 1-1.

Porém, a realização dos testes planejados com um volume de dados de 200 GB, além de demandar mais tempo de execução, exigiria maiores recursos de armazenamento para a criação das imagens forenses. Devido a essa limitação de recursos, a utilização de um disco de dados de origem pequeno parecia ser adequada.

Entretanto, os resultados do estudo seriam limitantes caso fossem aplicáveis apenas a discos de 20 GB. Para que o resultado fosse aplicável a discos maiores, seria necessário que as proporções de variação de tempo, em relação ao tamanho de disco, formato de imagem, sistema de arquivos destino e ferramenta de aquisição fossem mantidas.

Para verificar se essa proporção se mantinha, foram executados testes preliminares, independentes dos casos de teste planejados, com as ferramentas DC3DD e EWFAcquire, com sistema de arquivos destino NTFS e com 512 setores por bloco, para diversos tamanhos de discos rígidos.

A Tabela 4-7 mostra a identificação do disco rígido na tabela A-2 do Anexo A, o tamanho do disco, a ferramenta utilizada e o tempo total dos testes executados.

Tabela 4-7: Testes com Variação de Tamanho do Disco de Origem

Número do Teste	Identificação do Disco	Tamanho do Disco	Ferramenta	Tempo Total
01	01-PATA	20	DC3DD	14m43.219s
02	01-PATA	20	EWFAcquire	15m9.703s
03	02-PATA	40	DC3DD	13m19.868s
04	02-PATA	40	EWFAcquire	15m35.545s
05	03-SATA	160	DC3DD	41m31.966s
06	03-SATA	160	EWFAcquire	56m11.260s
07	04-SATA	250	DC3DD	51m53.746s
08	04-SATA	250	EWFAcquire	73m55.786s
09	05-SATA	320	DC3DD	67m25.845s
10	05-SATA	320	EWFAcquire	118m22.324s

A Figura 4-2 mostra a comparação do tempo de aquisição com DC3DD para diferentes tamanhos de disco.

Tanto para DC3DD como para EWFAcquire (Figura 4-3), observa-se um aumento de tempo de aquisição proporcional ao tamanho do disco.

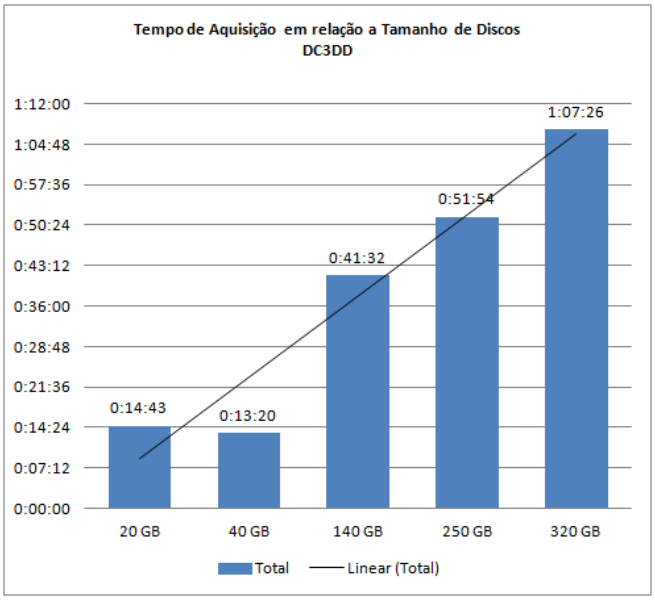

Figura 4-2: Tempos de Aquisição com DC3DD em relação ao tamanho de disco

Com esses resultados, pode-se afirmar que os tempos de aquisição deverão manter as proporções observando as variações de tempo em função do sistema de arquivo, do tamanho dos blocos e das ferramentas de cópia utilizadas.

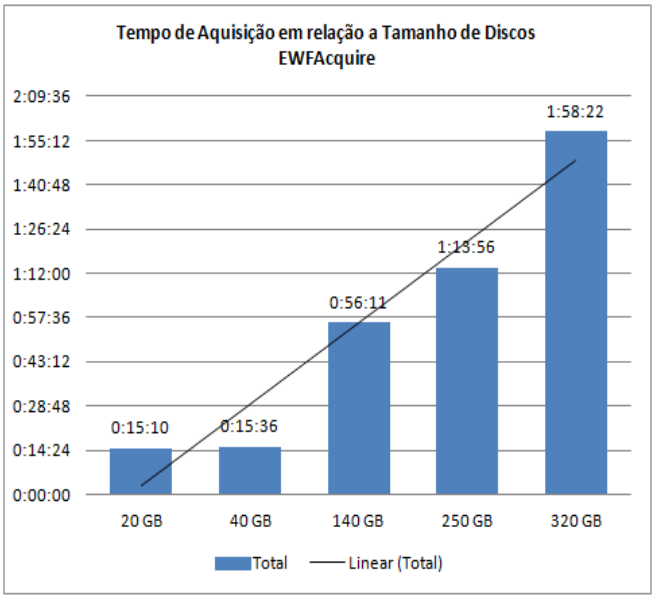

Figura 4-3: Tempos de Aquisição com EWFAcquire em relação ao tamanho de disco

Dessa forma, foi possível contornar os problemas de falta de recursos para a execução dos casos de teste, pois os resultados obtidos com o disco de 20 GB poderão ser usados como referência para outros tamanhos de discos.

4.2. ANÁLISE DOS RESULTADOS

Seguindo o fluxograma de testes da Figura 4-1, foram executados testes para verificar a relação entre a velocidade de aquisição e o tamanho do bloco/sistema de arquivos (testes DA-06), e também entre a velocidade de aquisição e a natureza dos dados de origem (testes DA-06, DA-10 e DA-(...)-CRIPTO), mostrados na Tabela 4-6. Outros dados de tempo de execução resultantes dos testes encontram-se organizados na tabela E-1 do Anexo E.

Esses dados podem ser analisados em relação à ferramenta utilizada, ao número de setores por bloco e ao tipo de sistema de arquivos de destino. Para todos os casos de teste, foi verificado comportamento esperado de acordo com as assertivas de teste da Tabela 4-8.

Tabela 4-8: Assertivas Verificadas nos Testes

Assertiva	Descrição
DA-AM-01	A ferramenta usa a interface de acesso SRC-AI para acessar a fonte digital.
DA-AM-02	A ferramenta adquire a fonte digital DS.
DA-AM-03	A ferramenta executa no ambiente de execução XE.
DA-AM-05	Se a criação do arquivo de imagem é especificada, a ferramenta cria um arquivo de imagem em um sistema de arquivos FS.
DA-AM-06	Todos os sectores visíveis são adquiridos a partir da fonte digital.
DA-AM-08	Todos os setores adquiridos a partir da fonte digital são adquiridos com precisão.
DA-AM-10	Se erros não resolvidos ocorrerem durante a leitura da fonte digital DS, a ferramenta grava um valor conhecido no objeto de destino, no lugar dos dados inacessíveis.
DA-AO-01	Se a ferramenta cria um arquivo de imagem os dados representados pelo arquivo é o mesmo que os dados adquiridos pela ferramenta.
DA-AO-02	Se um formato de arquivo de imagem é especificado, a ferramenta cria um arquivo de imagem no formato especificado.
DA-AO-03	Se houver um erro ao escrever o arquivo de imagem, a ferramenta notifica o usuário.
DA-AO-04	Se a ferramenta está criando um arquivo de imagem e não há espaço suficiente no dispositivo de destino imagem, a ferramenta deve notificar o usuário.
DA-AO-05	Se a ferramenta está criando uma imagem com vários arquivos de tamanho solicitado, então todos os arquivos individuais devem ser do tamanho solicitado, exceto que o último arquivo pode ser menor.
DA-AO-06	Se a ferramenta executa uma verificação de integridade de arquivos em arquivo de imagem que não foi alterado desde que o arquivo foi criado, a ferramenta notifica o usuário que o arquivo de imagem não foi alterado.

Como visto no item 2.4.2, o formato de imagem E01 necessita de maior tempo de processamento para ser criado em relação ao formato RAW, pois executa cálculo de código CRC para cada bloco de 64 setores. Dessa forma, a comparação entre tempos de aquisição entre ferramentas que usam diferentes formatos de imagem não pode ser considerada, demandando a análise em separado dessas duas configurações.

As seções abaixo apresentam a análise sobre os resultados de desempenho obtidos.

4.2.1. Resultados para Imagem do Tipo Raw

O gráfico da Figura 4-4 mostra os tempos de execução dos testes DA-06, para ferramentas que gravam o arquivo de imagem no formato RAW. O percentual de diferença entre os testes é mostrado na Tabela E-2, do Anexo E.

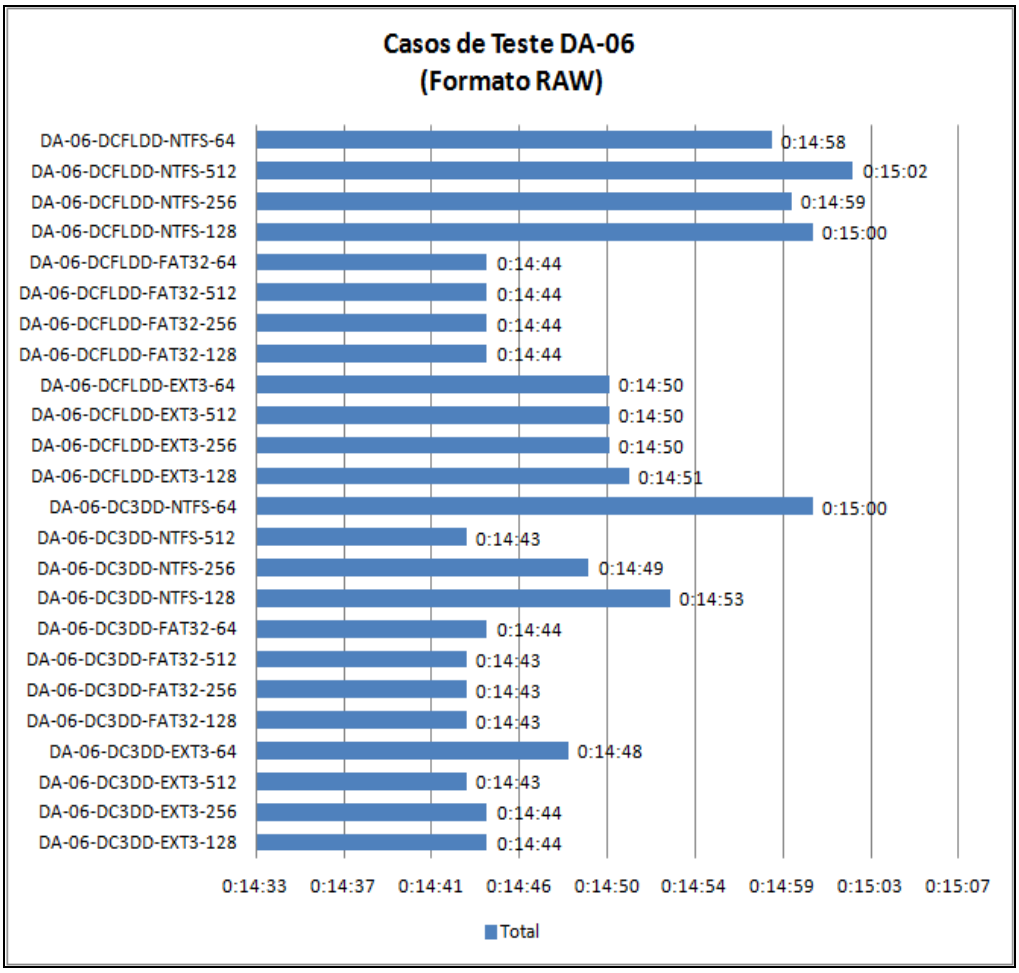

Figura 4-4: Tempos de Aquisição dos Testes DA-06 (Formato RAW)

A comparação da Figura 4-5 indica que o DC3DD foi, em média, 0,92% mais rápido que o DCFLDD. A maior diferença de tempo ocorreu no teste DA-06-(...)-NTFS-512, onde o DC3DD foi 2%, ou 18 segundos, mais rápido que o DCFLDD.

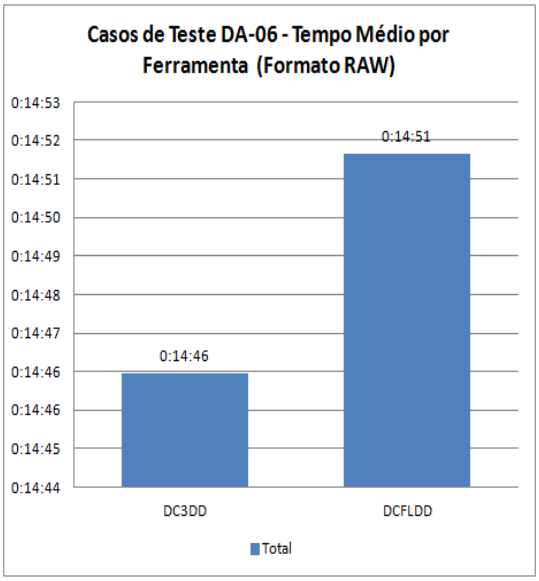

Figura 4-5: Tempo Médio por Ferramenta (Formato RAW)

A Figura 4-6 mostra o tempo médio por tamanho de bloco. Para DC3DD, o bloco com 512 setores apresentou um desempenho 0,89% que o bloco com 64.

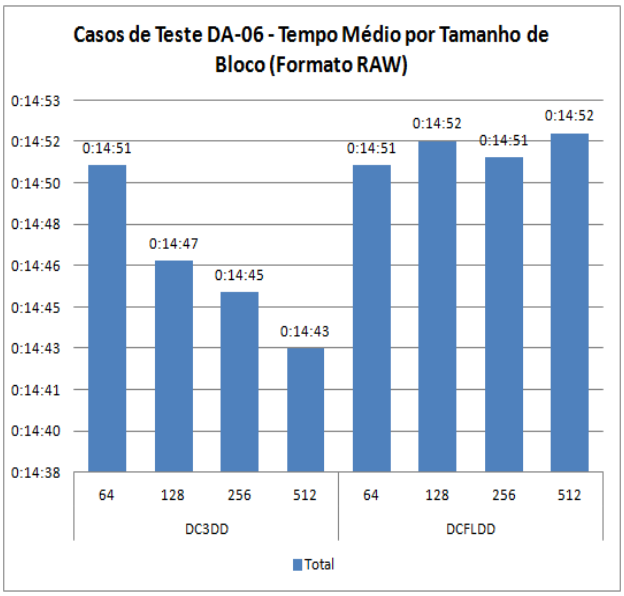

Figura 4-6: Tempo Médio por Tamanho de Bloco (Formato RAW)

As variações de número de setores por bloco do DCFLDD apresentaram pouca diferença entre seus resultados.

A comparação entre sistemas de arquivo mostra que o FAT32 apresentou um desempenho 0,9% mais rápido em relação ao NTFS, e 0,44% em relação ao EXT3 (Figura 4-7).

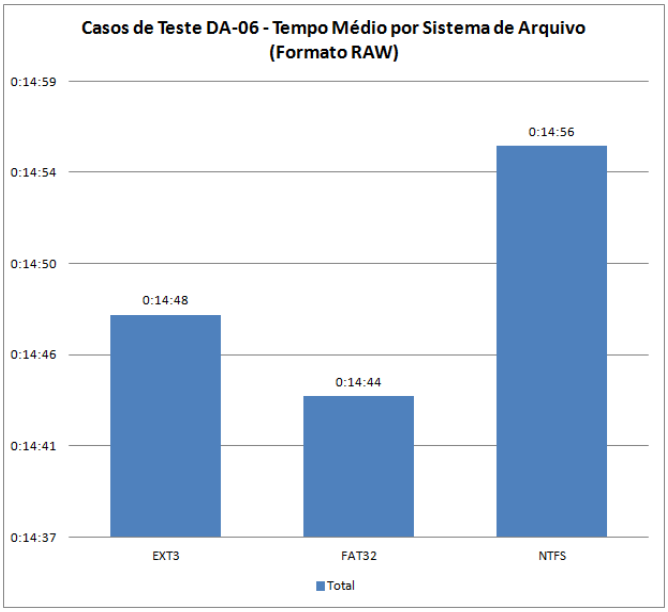

Figura 4-7: Tempo Médio por Sistema de Arquivos (Formato RAW)

4.2.2. Resultados para Imagem do Tipo E01

A Tabela E-3 do Anexo E mostra dados estatísticos dos testes DA-06 para o formato E01. A Figura 4-9 mostra os tempos obtidos para estes casos de teste.

Como visto anteriormente, o FTKimager não permite a escolha do tamanho do bloco, usando o tamanho padrão de 512 setores por bloco.

Considerando apenas a comparação entre estes casos de teste com 512 setores por bloco, o EWFAcquire teve o melhor desempenho médio, sendo em média 40,69% mais rápido que o LinEn e 21,77 % mais rápido que o FTKImager (Figura 4-8).

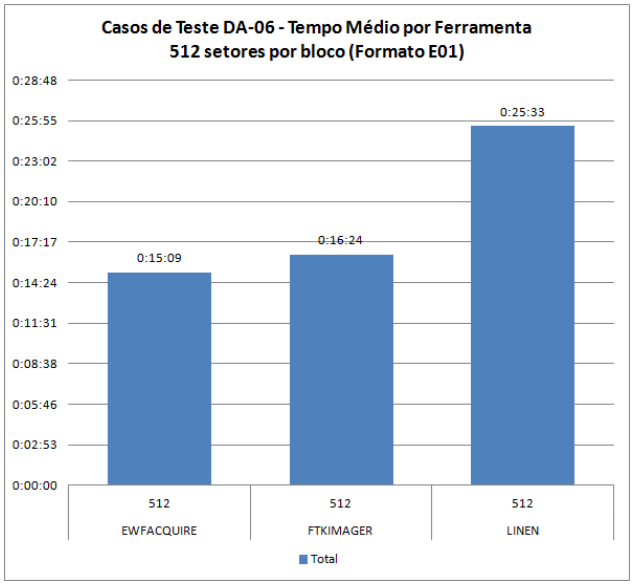

Figura 4-8: Tempo Médio por Ferramenta para 512 setores por Bloco (Formato E01)

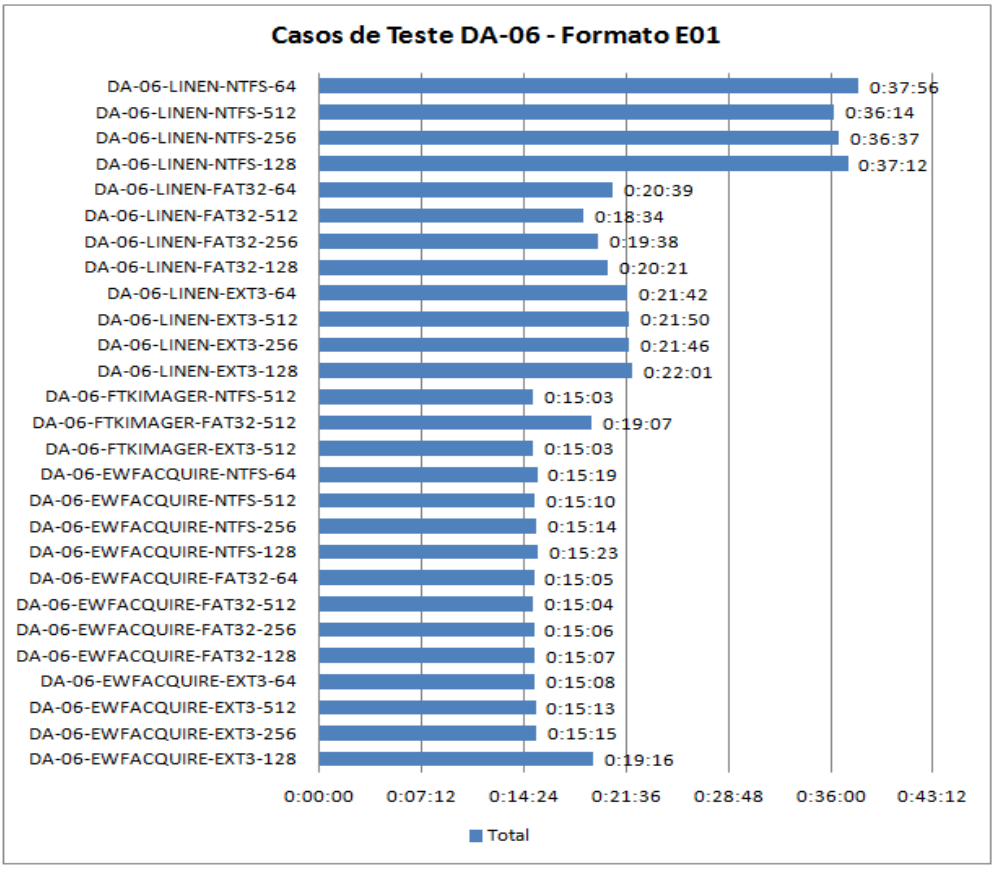

Figura 4-9: Tempos de Aquisição (Formato E01)

Na comparação entre o tamanho de bloco usado, os menores tempos de aquisição foram obtidos usando 512 setores por bloco (Figura 4-10), sendo em média 4,74% mais rápido que 64 setores por bloco, 7,67% mais rápido que 128 setores por bloco e 2,9 % mais rápido que 256 setores por bloco.

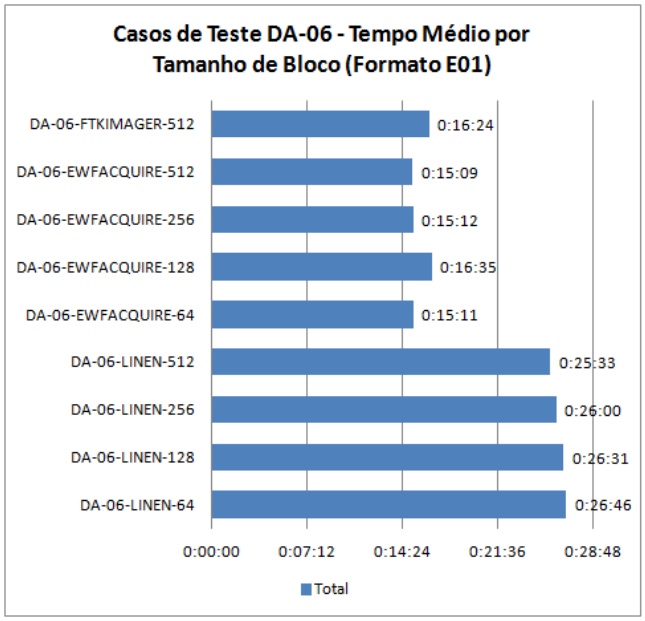

Figura 4-10: Tempo Médio de Aquisição por Tamanho de Bloco (em setores), para formato E01

Entre os sistemas de arquivos usados, o FAT32 apresentou o melhor desempenho médio, sendo 30,65% mais rápido em relação ao tempo médio do NTFS, e 7,45% mais rápido em relação ao tempo médio do EXT3 (Figura 4-11).

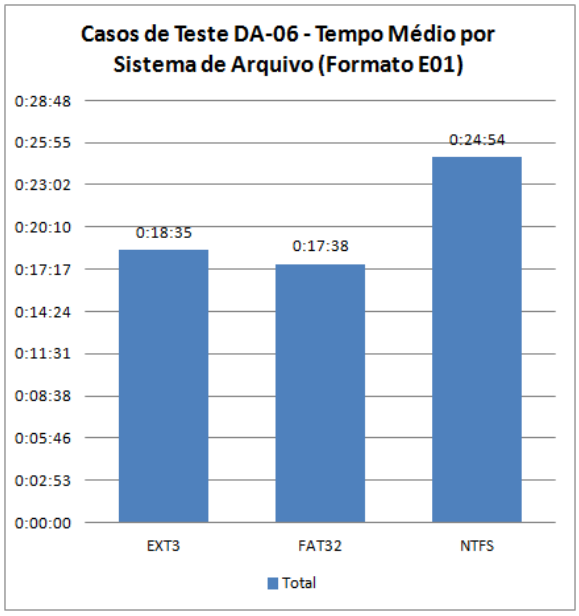

Figura 4-11: Tempo Médio por Sistema de Arquivos (Formato E01)

4.2.3. Comparação Entre Natureza Dos Dados E Velocidade De Aquisição

Conforme exposto anteriormente, os testes DA criam imagens de um conjunto de dados homogêneo, enquanto os testes DA-(...)-CRIPTO criam imagens de um conjunto de dados criptografado, ou seja, dados altamente heterogêneos. Para facilitar essa comparação, foram realizados testes DA-(...)-CRIPTO apenas para o sistema de arquivos NTFS, com bloco de 512 setores. A Figura 4-12 mostra as diferenças de tempo entre DA-06 e DA-06-(...)-CRIPTO. Essa variação é pequena, em média de 0,33%, mas em todos os testes DA-06-CRIPTO foi verificado aumento semelhante para tempo de usuário, indicando mais tempo de processamento para cálculos de *hashes*.

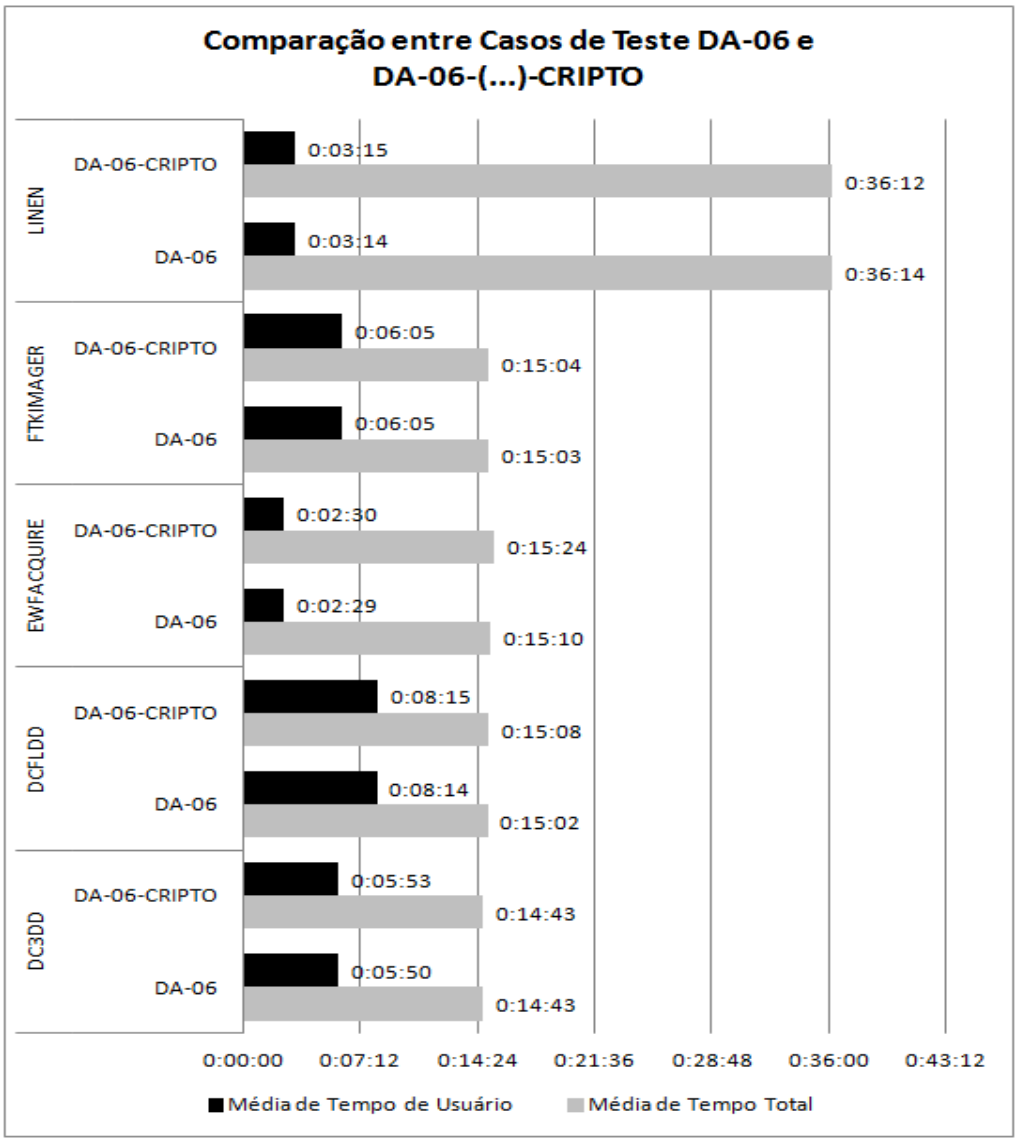

Figura 4-12: Comparação entre testes DA-06 e DA-06-(...)-CRIPTO

Da mesma forma, a influência da natureza dos dados mostra-se mais acentuada na análise dos experimentos DA-10, que usam a funcionalidade de compressão de dados do formato E01(Figura 4-13). Para dados homogêneos, a média de tempo total dos experimentos DA-10 (imagem no formato E01 comprimido) foi 14,63% maior em relação à média dos testes DA-06. Já a média de tempo de usuário foi 124,01 % em relação aos testes DA-06, indicando o tempo adicional necessário para execução da compressão.

A média de tempo total dos experimentos DA-10-(...)-CRIPTO (imagem de dados cripotgrafados no formato E01 comprimido) foi 27,02% maior em relação à média dos testes DA-06. Já o tempo de usuário dos testes DA-10-CRIPTO mostra que é gasto muito mais tempo de processamento para comprimir um conjunto de dados muito heterogêneo. Essa diferença é de 309,58% em relação aos testes DA-06.

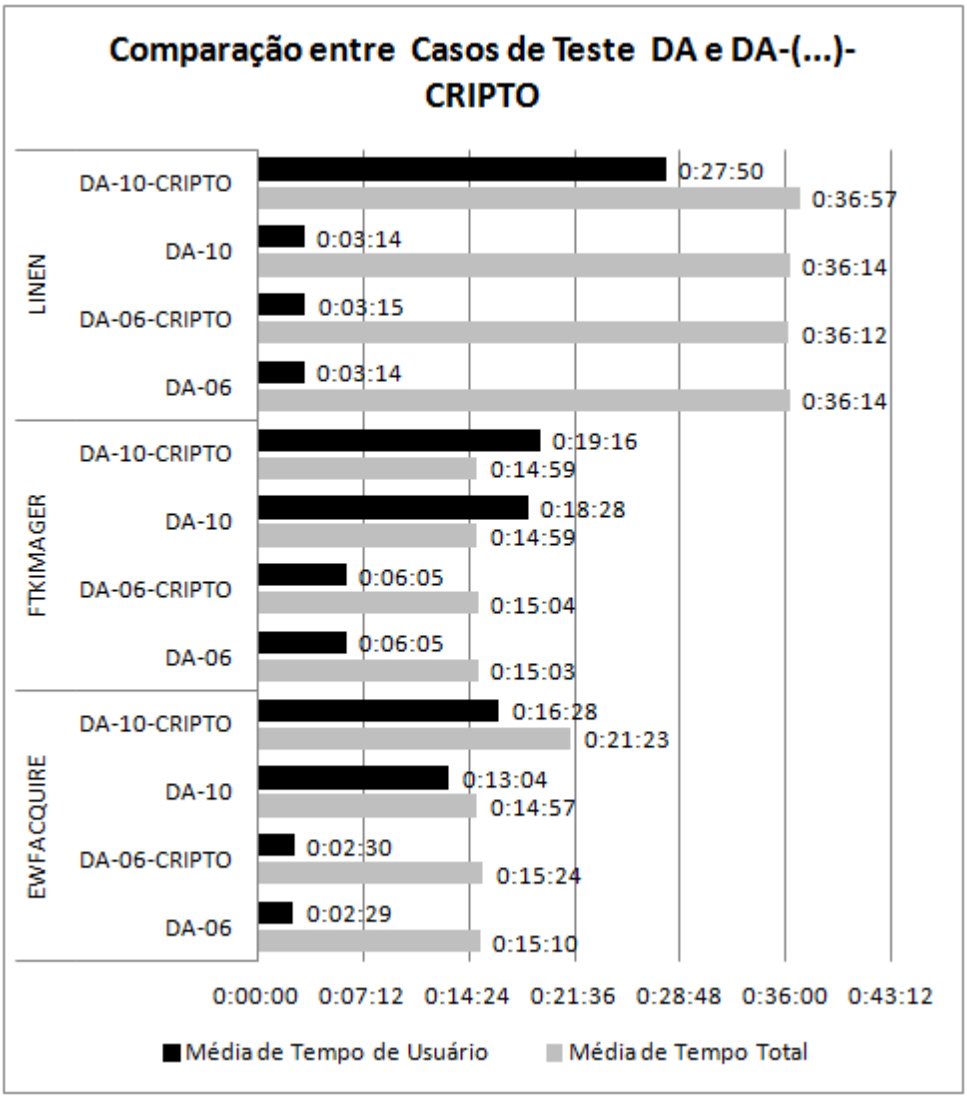

Figura 4-13: Comparação entre Testes DA e DA-(...)-CRIPTO

5. CONCLUSÃO

Este trabalho apresentou um método adicional para o processo de duplicação forense, com base nos resultados de desempenho de ferramentas escolhidas, em diferentes cenários. Foram mostradas as bases jurídicas da exigência da duplicação forense no processo de análise forense de evidências digitais. Em seguida, foram estudados aspectos dos discos rígidos, e as metodologias usadas atualmente para o processo de duplicação forense deste tipo de mídia.

A velocidade de acesso aos dados de um disco rígido é variável, dependendo da posição do dado no disco. Como o disco é lido de fora para dentro, e as trilhas mais externas são maiores e contém mais dados, a velocidade do início da cópia é mais lenta. Ao copiar as trilhas mais internas ao disco, a velocidade é maior. Os testes efetuados com EWFAcquire geraram dados que mostraram esse comportamento, com a velocidade de acesso variando de 25 MB/s no início do disco à 21 MB/s no seu final, com velocidade média de 21 MB/s.

Entretanto, as metodologias estudadas mostram que, quanto mais baixa a camada de abstração de dados usada, maior a quantidade de informações/evidências que pode ser recuperada. Assim, a imagem do disco rígido deve ser copiada integralmente, através de uma duplicação bit-a-bit. Como os dados são copiados sequencialmente, a variação de velocidade em relação à posição do dado no disco pode ser desprezada.

A busca por melhorias no desempenho da duplicação forense incluiu a realização testes que indicassem quais parâmetros influenciam no tempo gasto nesse procedimento. Os cenários testados incluiram a variação do sistema de arquivo destino, do tamanho de setores por bloco e da natureza dos dados de origem. Foram utilizadas as ferramentas LinEn, FTKImager e EWFAcquire, para gravação de imagens no formato E01, e as ferramentas DC3DD e DCFLDD, para o formato RAW.

A capacidade do disco rígido de origem usado nos casos de teste é de 20 GB. Para verificar se as proporções de variação de tempo em relação aos parâmetros testados se mantinham para discos com maior capacidade de armazenamento, foram feitos testes extras com discos de maior capacidade. Esses testes indicaram que o aumento de tempo em relação à capacidade dos foi similar a um aumento linear. Dessa forma, os resultados dos casos de teste com o disco de 20 GB podem ser usados como referência para discos com maior capacidade.

O tipo de dados de origem influencia a velocidade de aquisição. Os casos de teste DA-06 e DA-10 utilizaram dados homogêneos, com o disco de origem sanitizado (*wiped*), com o valor

Um gravado na parte de dados dos seus setores, enquanto que os testes DA-06-(...)-CRIPTO e DA-10-(...)-CRIPTO utilizaram o mesmo disco de origem, porém contendo uma imagem criptografada do disco rígido sanitizado. Estes dados de origem muito heterogêneos, como discos rígidos criptografados, compactados ou com arquivos contendo fotografias ou vídeos, exigem maior processamento no caso de criação de imagens do tipo EWF/E01 comprimidos, devido à análise dos dados para compressão e aos cálculos de *hashes*. Por outro lado, a diferença de tempo de processamento para cálculo de *hashes* não apresentou diferenças significativas entre dados de origem homogêneos ou heterogêneos. O tempo de aquisição para imagens do tipo RAW também não apresentou variação significativa quanto à natureza dos dados de origem.

A quantidade de tempo para criação de imagens forenses varia em relação ao formato do arquivo destino. O formato de imagem EWF/E01 exige maior processamento para sua criação, devido ao uso de estruturas de encapsulamento dos dados de origem e dos cálculos para controle de erro e integridade da cópia, armazenados nessas estruturas. O formato RAW, por não criar estruturas de dados, mostrou-se mais rápido.

O sistema de arquivos destino também influencia na velocidade do procedimento. O desempenho do NTFS foi inferior, talvez pelo fato de não ser acessado nativamente pelo sistema operacional Linux. O sistema de arquivos FAT32 apresentou um melhor desempenho, com a vantagem de poder ser acessado nativamente tanto pelo Linux como pelos sistemas Windows, facilitando o acesso ao arquivo de imagem para a fase de análise de evidências.

A quantidade de setores por bloco de leitura influenciou principalmente as imagens com formato E01. Quanto maior o número de setores por bloco, menor deve ser o tempo de aquisição para esse formato, pois um número menor de cálculos CRC será efetuado. O limite do LinEn para o número de setores por bloco é de 512, enquanto que o limite do EWFAcquire é 32768 setores, o que deve prover uma velocidade ainda maior. Entretanto, o parâmetro de granularidade de erro deve ser cuidadosamente escolhido para que, nos casos de erros de leitura do dispositivo, a maior quantidade de informações úteis seja extraída. Os casos de teste com imagem em formato RAW não apresentaram diferenças significativas em relação à variação do número de setores por bloco.

Os resultados mostraram que dados de origem muito heterogêneos e o número de setores por bloco não influenciam no desempenho da duplicação forense na criação de imagens em formato RAW. Para o formato E01, o uso do sistema de arquivos FAT32 e de 512 setores por

bloco garantiu um melhor desempenho. Entretanto, quando o formato da imagem for o E01 comprimido, dados heterogêneos influenciam negativamente o desempenho.

A partir dos resultados dos casos de teste, a aplicação dos passos adicionais propostos apontou para a melhor configuração a ser usada, com o objetivo de obter um melhor desempenho na duplicação forense. Dessa forma, conclui-se que a aplicação do método proposto proporciona uma melhora no desempenho do procedimento de duplicação forense, apontando as melhores configurações em relação aos cenários apresentados.

5.1. TRABALHOS FUTUROS

Há assuntos que podem ser trabalhados em profundidade a partir deste estudo. A avaliação de desempenho de outras soluções de duplicação forense, como duplicadores dedicados e bloqueadores de escrita por *hardware* pode fornecer melhores parâmetros para avaliação de aquisição dessas soluções.

A manipulação de parâmetros e estruturas dos sistemas de arquivos, com o estudo de acesso ao NTFS pelos sistemas operacionais LINUX, também pode ser estudado, devido ao pobre desempenho apresentado na duplicação forense.

O desempenho de acesso ao arquivo de imagens, em relação ao tipo de imagem (RAW ou E01) e ao uso de compressão, importante para a fase de análise das evidências adquiridas, pode complementar o presente estudo, indicando os melhores parâmetros de desempenho para essa fase.

REFERÊNCIAS BIBLIOGRÁFICAS

Bartol, N.; Bates, B.; Goertzel, K. M.; Winograd, T. (2009). Measuring Cybersecurity and Information Assurance. State-of-Art Report. IATAC – Information Assurance Technology Analysis Center.

Beebe, N. L., Clark, J. G. (2005). A hierarchical, objectives-based framework for the digital investigations process. Digital Investigation.

Botchek, R., (2008). Benchmarking Hard Disk Duplication Performance in Forensic Applications. Tableau LLC. EUA.

Carrier, B. (2005). File System Forensic Analisys. EUA. Addison Wesley Professional.

Carrier, B. (2006). Risks of live digital forensic analysis. Communications of the ACM,

49:56–61.

Carvalho, A. C. A. P.; Sousa, F. B.; Neto, J. F.; Neves, P. H. C.; Fragoso, R.; Mazzonetto, R.P. (2008). Crimes da Informática no Código Penal Brasileiro, 97-103. ICOFCS, 2008.

CFTT/NIST. 2005. Digital Data Acquisition Tool Test Assertions and Test Plan. Version 1.0. Acessado em 02/06/2011 em http://www.cftt.nist.gov/DA-ATP-pc-01.pdf.

CFTT/NIST. 2012. Computer Forensics Tool Testing Handbook. Data de Revisão 02/01/2012. Acessado em 02/03/2012 em http://www.cftt.nist.gov/CFTT-Booklet-Revised-02012012.pdf.

Código Penal. (1940). República Federativa do Brasil. Acessado em 27/06/2012 em http://www.planalto.gov.br/ccivil_03/Decreto-Lei/Del2848.htm.

Código de Processo Penal. (1941). República Federativa do Brasil. Acessado em 27/06/2012 em http://www.planalto.gov.br/ccivil_03/Decreto-Lei/Del3689.htm.

Costa, M.A.S.L. (2011). Computação Forense. 3ª Edição. Millenium Editora. São Paulo.

Daniels D.J.; Hart S.V. (2004). Forensic Examination of Digital Evidence : A Guide for Law Enforcement. Office, 44(2), 634-111. National Institute of Justice. EUA.

E-fense Helix3. (2009). Versão R20091, disponível em https://www.e-fense.com/store/index.php?_a=viewProd&productId=11, acessada em 20/09/2011.

Garfunkel, S., Malan, D., Dubec, S., Stevens, C. e Pham, C. 2006. Advanced Forensic Format: An Open, Extensible Format for Disk Imaging. Advances in Digital Forensics II. http://www.springerlink.com/content/guq2425227t8/front-matter.pdf, acessado em 11/07/2012.

Hankins, R.; Uehara, T.; Liu, J. (2009). A Comparative Study of Forensic Science and Computer Forensics. 2009 Third IEEE International Conference on Secure Software Integration and Reliability Improvement.

Hoelz, B. W. P. (2009). MADIK Uma Abordagem Multiagente Para O Exame Pericial de Sistemas Computacionais. Dissertação de Mestrado, Universidade de Brasília, Brasil.

Holder E.H., Robinson L. O., 2008. Special Report Test Results for Forensic Media Preparation Tool : *National Institute of Justice Special Report. EUA*.

IOCE. (2002). Guidelines for best practice in the forensic examination of digital technology. Acessado de http://www.ioce.org/fileadmin/user_upload/2002/ioce_bp_exam_digit_tech.html

Kruse, W. G.; Heiser, J. G. (2002). Computer forensics: incident response essentials. Ed. Addison-Wesley. pp. 392. ISBN 0201707195.

Marques, J. F. (2002). Tratado de Direito Penal-Volume II. Ed. Millennium.

Moore, R. (2005). Cybercrime: Investigating High-Technology Computer Crime. Anderson Publishing, EUA.

Mukasey M.B., Sedgwick J.L., Hagy D.W., 2008. Test Results for Digital Data Acquisition Tool : EnCase LinEn 6.01. *National Institute of Justice Special Report*. EUA.

National Academy of Sciences (2009). Strengthening Forensic Science in the United States : A Path Forward. The National Academies Press, EUA.

National Institute of Justice [A] (2008). Test Results for Digital Data Acquisition Tool: Encase LinEn 6.01. in NIJ Special Reports. www.ojp.usdoj.gov/nij, acessado em 02/06/2012.

National Institute of Justice [B] (2008). Test Results for Forensic Media Preparation Tool: DC3dd. in NIJ Special Reports. www.ojp.usdoj.gov/nij, acessado em 02/06/2012.

Piropo, B. (2007). Computadores XXXV: Métodos de acesso.

http://blogs.forumpcs.com.br/bpiropo/2007/07/07/computadores-xxxv-metodos-de-acesso/, acessado em 10/03/2012.

Ruback, M. C. (2011). Mineração de Dados Aplicada à Construção de Bases de *Hash* em Computação Forense. Dissertação de Mestrado. Universidade de Brasília.

Schneier, B. (1995). One-way *hash* functions. In: Applied cryptography. 2nd Edition. John Wiley & Sons. EUA.

SWGDE. (1999). Digital Evidence : Standards and Principles - Proposed Standards for the Exchange of Digital Evidence.

Silberschatz, A., Galin, P., Gagne G., (2000). Sistemas Operacionais: Conceitos e Aplicações. Rio de Janeiro: Editora Campus.

Stallings, W. (2008). Criptografia e Segurança de Redes. 4ª Edição. Pearson Prentice Hall, 2008.

Tessmann, C. (2012). Priorização da Análise Pericial pela Amostragem Estatística de Dados de Discos Rígidos Suspeitos. Dissertação de Mestrado, Universidade de Brasília, Brasil.

WikiPedia[A], 2012. List of Digital Forensics Tools. http://en.wikipedia.org/wiki/List_of_digital_forensics_tools, acessado em 11/07/2012.

Wikipedia[B], 2012. Sistemas de Arquivos. http://pt.wikipedia.org/wiki/Sistema_de_ficheiros, acessado em 12/07/2012.

Zelenovsky, R., Mendonça, A., (2006). PC: Um Guia Prático de Hardware e Interfaceamento. Rio de Janeiro: MZ Editora.

ANEXOS

A. EQUIPAMENTOS UTILIZADOS NOS EXPERIMENTOS

O equipamento utilizado para realização dos experimentos deste trabalho foi uma das estações periciais da Seção de Informática Forense do Departamento de Criminalística do Instituto-Geral de Perícias / SSP / RS, e é descrito na tabela A-1. A tabela A-2 apresenta a descrição detalhada dos discos rígidos utilizados.

Tabela A-1: Descrição do Equipamento Utilizado

Tipo: Computador Pessoal.
Placa de Sistema: ASUS P5Q-E.
CPU: Intel Core 2 Quad Q9550 à 2,83GHz.
Memória RAM: Tipo DDR2, capacidade de 4GB, configurada em duplo canal à 400MHz com latência 4-4-4-12.
Sistema Operacional: Linux Helix 10.04 LTS 64 bits.

Tabela A-2: Características dos Discos Rígidos Utilizados

Nome do Disco	**Marca**	**Modelo**	**Interface**	**LBA**	**RPM**	**Tamanho do Setor (bytes)**	**Capacidade Total (GB)**
01-PATA	Samsung	SV0221N	IDE	39.179.952	5.400	512	20
02-PATA	Western Digital	WD400	IDE	78.156.288	5.400	512	40
03-SATA	Samsung	HD161HJ	SATA	312.581.808	7.200	512	160
04-SATA	Samsung	HD250HJ	SATA	488.397.168	7.200	512	250
05-SATA	Samsung	HD322HJ	SATA	625.142.448	7.200	512	320
06-SATA	Samsung	HD502HI	SATA 3.0GB/s	976.773.168	5.400	512	500

B. ASSERTIVAS DE TESTE DO CFTT/NIST

Tabela B-1: Assertivas de Teste do CFTT/NIST (CFTT/NIST, 2005)

Assertiva	Descrição
DA-AM-01	A ferramenta usa a interface de acesso SRC-AI para acessar a fonte digital.
DA-AM-02	A ferramenta adquire a fonte digital DS.
DA-AM-03	A ferramenta executa no ambiente de execução XE.
DA-AM-05	Se a criação do arquivo de imagem é especificada, a ferramenta cria um arquivo de imagem em um sistema de arquivos FS.
DA-AM-06	Todos os sectores visíveis são adquiridos a partir da fonte digital.
DA-AM-07	Todos os setores ocultos são adquiridos a partir da fonte digital.
DA-AM-08	Todos os setores adquiridos a partir da fonte digital são adquiridos com precisão.
DA-AM-09	Se erros não resolvidos ocorrerem durante a leitura da fonte digital DS, a ferramenta notifica o usuário sobre o tipo de erro e a localização dentro da fonte digital.
DA-AM-10	Se erros não resolvidos ocorrerem durante a leitura da fonte digital DS, a ferramenta grava um valor conhecido no objeto de destino, no lugar dos dados inacessíveis.
DA-AO-01	Se a ferramenta cria um arquivo de imagem os dados representados pelo arquivo é o mesmo que os dados adquiridos pela ferramenta.
DA-AO-02	Se um formato de arquivo de imagem é especificado, a ferramenta cria um arquivo de imagem no formato especificado.
DA-AO-03	Se houver um erro ao escrever o arquivo de imagem, a ferramenta notifica o usuário.
DA-AO-04	Se a ferramenta está criando um arquivo de imagem e não há espaço suficiente no dispositivo de destino imagem, a ferramenta deve notificar o usuário.
DA-AO-05	Se a ferramenta está criando uma imagem com vários arquivos de tamanho solicitado, então todos os arquivos individuais devem ser do tamanho solicitado, exceto que o último arquivo pode ser menor.
DA-AO-06	Se a ferramenta executa uma verificação de integridade de arquivos em um arquivo de imagem que não foi alterado desde que o arquivo foi criado, a ferramenta notifica o usuário que o arquivo de imagem não foi alterado.

Assertiva	Descrição
DA-AO-07	Se a ferramenta executa uma verificação de integridade de arquivos em um arquivo de imagem que foi alterado desde que o arquivo foi criado, a ferramenta notifica o usuário que o arquivo de imagem foi alterado.
DA-AO-08	Se a ferramenta executa uma verificação de integridade de arquivos em um arquivo de imagem que foi alterado desde que o arquivo foi criado, a ferramenta notifica o usuário sobre os locais afetados.
DA-AO-09	Se a ferramenta converte um arquivo de imagem de um formato para um arquivo de imagem destino em outro formato, os dados adquiridos, representados no arquivo de imagem destino, são os mesmos que os dados adquiridos no arquivo de imagem de origem.
DA-AO-10	Se não houver espaço suficiente para conter todos os arquivos de uma imagem com vários arquivos, e a mudança de dispositivo de destino é suportada, a imagem continua em outro dispositivo.
DA-AO-11	Se requerido, um clone é criado durante a aquisição de uma fonte digital DS.
DA-AO-12	Se solicitado, um clone é criado a partir de um arquivo de imagem.
DA-AO-13	Um clone é criado usando a interface de acesso DST-AI para escrever para o dispositivo destino.
DA-AO-14	Se um clone alinhado é criado, cada setor gravado no clone é gravado com precisão para o mesmo endereço de disco ocupado na fonte digital DS.
DA-AO-15	Se um clone alinhado é criado, cada setor contido em um intervalo contíguo de setores da origem é gravado com precisão para o endereço de disco no dispositivo destino relativo ao início do intervalo contíguo do setor do dispositivo origem. (Um intervalo de setores pode ser uma partição montável ou uma sequência contígua de sectores como, por exemplo, setores não-alocados. As partições estendidas, que podem conter tanto partições montáveis e setores não alocados, não são partições montáveis.)
DA-AO-16	Se um subconjunto de uma imagem ou de aquisição é especificado, todo o subconjunto é clonado.
DA-AO-17	Se for solicitado, todos os setores em excesso em um dispositivo destino não são modificados.
DA-AO-18	Se solicitado, um preenchimento benigno é gravado para setores em excesso do dispositivo destino.
DA-AO-19	Se não há espaço suficiente para criar uma cópia completa, uma cópia truncada é criada usando todos os sectores disponíveis do dispositivo

Assertiva	Descrição
	destino.
DA-AO-20	Se uma cópia truncada é criada, a ferramenta notifica o usuário.
DA-AO-21	Se houver um erro de escrita durante a criação da cópia, a ferramenta notifica o usuário.
DA-AO-22	Se solicitado, a ferramenta calcula hashes de bloco para um tamanho de bloco especificado durante uma aquisição para cada bloco adquirido a partir da fonte digital.
DA-AO-23	Se a ferramenta registra qualquer informação de log significativa, a informação é devidamente registrada no arquivo de log.
DA-AO-24	Se a ferramenta executa em um ambiente de execução forense seguro, a fonte digital é inalterada pelo processo de aquisição.

C. CASOS DE TESTE DO CFTT/NIST

Tabela C-1: Casos de teste do CFTT/NIST (CFTT/NIST, 2005)

Identificação do Teste	Descrição do Teste
DA-01	Adquirir um dispositivo físico utilizando interface de acesso AI para uma cópia não alinhada.
DA-02	Adquirir uma fonte digital do tipo DS para uma cópia não alinhada.
DA-03	Adquirir um dispositivo físico para uma cópia alinhada por cilindro.
DA-04	Adquirir um dispositivo físico para uma cópia truncada.
DA-05	Responder a um erro de gravação no dispositivo destino durante uma aquisição de uma cópia.
DA-06	Adquirir um dispositivo físico utilizando interface de acesso AI para um arquivo de imagem.
DA-07	Adquirir uma fonte digital do tipo DS para um arquivo de imagem.
DA-08	Adquirir uma unidade física com setores ocultos para um arquivo de imagem.
DA-09	Adquirir uma fonte digital que tem pelo menos um setor de dados defeituoso.
DA-10	Adquirir uma fonte digital para um arquivo de imagem em um formato alternativo.
DA-11	Responder a um erro de disco ao escrever um arquivo de imagem.
DA-12	Tentar criar um arquivo de imagem onde não há espaço suficiente.
DA-13	Criar um arquivo de imagem, onde não há espaço suficiente em um único volume, e utilizar mudança de dispositivo destino para continuar em outro volume.
DA-14	Criar uma cópia não alinhada de um arquivo de imagem.
DA-15	Criar uma cópia alinhada por cilindro de um arquivo de imagem.
DA-16	Criar uma cópia de um subconjunto de um arquivo de imagem.
DA-17	Criar uma cópia truncada de um arquivo de imagem
DA-18	Responder a um erro de gravação no dispositivo destino durante a criação de uma cópia a partir de uma imagem.
DA-19	Adquirir um dispositivo físico para uma cópia desalinhada, preenchendo setores em excesso.
DA-20	Adquirir um dispositivo lógico para uma cópia desalinhada, preenchendo

Identificação do Teste	Descrição do Teste
	setores restantes.
DA-21	Adquirir um dispositivo físico para uma cópia alinha por cilindro, preenchendo setores restantes.
DA-22	Criar uma cópia não alinhada de um arquivo de imagem, preenchendo setores restantes.
DA-23	Criar uma cópia alinhada por cilindro a partir de um arquivo de imagem, preenchendo setores restantes.
DA-24	Verificar se uma imagem é válida.
DA-25	Detectar uma imagem corrompida.
DA-26	Converter uma imagem para um formato alternativo de arquivo de imagem.

D. DESCRIÇÃO DOS TESTES EXECUTADOS

Tabela D-1: DA-PREPARAÇÃO-01

Item	Descrição
Identificação do Teste	DA-PREPARAÇÃO-01
Resumo do Teste	*Wipe* do disco de origem.
Comentário	*Wipe* do disco de origem com a ferramenta disk*wipe*, com valor de preenchimento "1".
Ferramentas necessárias	Disk*wipe*[4], sha1sum, md5sum
Dispositivos	01-PATA
Informações de Registros	SAMSUNG SV0221N S01NJ10WC00253 39179952 on drive /dev/sdc *Wipe* Drive /dev/sdc *Wipe*out from 0 up to 39179952
Comando Utilizado	./disk*wipe wipe*-disk01 host01 daniel /dev/sdc 1
Resultados Obtidos	Disco preenchido com "1": 39179952 sectors *wipe*d with 1 elapsed time 0:39:31 Normal exit SHA1SUM: 1c3fe3d2d2c717bc74635e89a3ff75048281e883 MD5SUM: 180ad904a9b257344d9c3f39488a6bea Disk /dev/sdc: 20.1 GB, 20060135424 bytes 64 heads, 63 sectors/track, 9717 cylinders Units = cylinders of 4032 * 512 = 2064384 bytes Sector size (logical/physical): 512 bytes / 512 bytes Disk identifier: 0x1d14c1e4

Tabela D-2: DA-PREPARAÇÃO-02

Item	Descrição
Identificação do Teste	DA-PREPARAÇÃO-02
Resumo do Teste	Criação de partição FAT32 do disco de destino.
Comentário	Criação de partição FAT32 do disco de destino com a ferramenta GPARTED.
Ferramentas necessárias	Gparted
Dispositivos	02-SATA
Resultados Obtidos	Device Boot Start End Blocks Id System /dev/sdb1 1 60801 488384001 b W95 FAT32

[4] Ferramenta da Suite Forensic Software Testing Support Tools (FS-TST) do NIST, disponível em http://www.nist.gov/itl/ssd/cs/cftt/cftt-disk-imaging.cfm.

Tabela D-3: DA-PREPARAÇÃO-03

Item	Descrição
Identificação do Teste	DA-PREPARAÇÃO-03
Resumo do Teste	Criação de partição EXT3 do disco de destino.
Comentário	Criação de partição EXT3 do disco de destino com a ferramenta GPARTED.
Ferramentas necessárias	Gparted
Dispositivos	02-SATA
Resultados Obtidos	Device Boot Start End Blocks Id System /dev/sdb1 1 60801 488384001 83 Linux

Tabela D-4: DA-PREPARAÇÃO-04

Item	Descrição
Identificação do Teste	DA-PREPARAÇÃO-04
Resumo do Teste	Criação de partição NTFS do disco de destino.
Comentário	Criação de partição NTFS do disco de destino com a ferramenta GPARTED.
Ferramentas necessárias	Gparted
Dispositivos	02-SATA
Resultados Obtidos	Device Boot Start End Blocks Id System /dev/sdb1 1 60801 488384001 7 HPFS/NTFS

Tabela D-5: DA-PREPARAÇÃO-05

Item	Descrição
Identificação do Teste	DA-PREPARAÇÃO-05
Resumo do Teste	Gravação de imagem criptografada no disco de origem.
Comentário	Criação de imagem criptografada do disco origem (preenchido com “1”) em um arquivo, e gravação do arquivo criptografado no disco de origem.
Ferramentas necessárias	Gparted, gpg
Dispositivos	01-PATA, 02-SATA

Tabela D-6: DA-06-DC3DD-FAT32

Item	Descrição
Identificação do Teste	DA-06-DC3DD-FAT32
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando DC3DD com sistema de arquivos destino FAT32, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, DC3DD.
Dispositivos	Origem: 01-PATA Destino:02-SATA
Variações	O teste será repetido com variação do parâmetro bufsz e ssz, com os valores 32K, 64K, 128K e 256K, equivalente em bytes a 64, 128, 256 e 512 setores, respectivamente.
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-02 sudo mount -o rw –t vfat /dev/sdb1 /media/sdb1

Variação: bufsz e ssz=32K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=32K ssz=32k verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-64.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-64.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=32k ssz=32k verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-64.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-64.log device size: 39179952 sectors (probed) sector size: 32768 bytes (set) 20060135424 bytes (19 G) copied (100%), 884.137 s, 22 M/s input results for device `/dev/sdc': 612186 sectors + 24576 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-fat32-64.111': 612186 sectors + 24576 bytes out 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.001' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.002' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.003' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.004' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.005' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.006' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.007' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.008' 65536 sectors out to `/media/sdb1/da-06-dc3dd-fat32-64.009' 22362 sectors + 24576 bytes out to `/media/sdb1/da-06-dc3dd-fat32-64.010'

	dc3dd completed at 2012-08-04 21:59:53 -0300 real 14m44.176s user 6m7.555s sys 1m30.870s	

Variação: bufsz e ssz =64K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=64K ssz=64K verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-128.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-128.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=64k ssz=64k verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-128.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-128.log device size: 39179952 sectors (probed) sector size: 65536 bytes (set) 20060135424 bytes (19 G) copied (100%), 883.206 s, 22 M/s input results for device `/dev/sdc': 306093 sectors + 24576 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-fat32-128.111': 306093 sectors + 24576 bytes out 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.001' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.002' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.003' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.004' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.005' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.006' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.007' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.008' 32768 sectors out to `/media/sdb1/da-06-dc3dd-fat32-128.009' 11181 sectors + 24576 bytes out to `/media/sdb1/da-06-dc3dd-fat32-128.010' dc3dd completed at 2012-08-04 22:14:37 -0300 real 14m43.249s user 5m52.770s sys 1m5.660s

Variação: bufsz e ssz=128K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=128K ssz=128 verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-256.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-256.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=128k ssz=128k verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-256.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-256.log device size: 39179952 sectors (probed) sector size: 131072 bytes (set) 20060135424 bytes (19 G) copied (100%), 883.396 s, 22 M/s input results for device `/dev/sdc': 153046 sectors + 90112 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-fat32-256.111': 153046 sectors + 90112 bytes out 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.001' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.002' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.003' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.004' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.005' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.006' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.007' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.008' 16384 sectors out to `/media/sdb1/da-06-dc3dd-fat32-256.009' 5590 sectors + 90112 bytes out to `/media/sdb1/da-06-dc3dd-fat32-256.010' dc3dd completed at 2012-08-04 22:29:20 -0300 real 14m43.440s user 5m46.930s sys 0m53.111s

Variação: bufsz e ssz =256K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=256K ssz=256K verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-64.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-512.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=256k ssz=256k verb=on ofs=/media/sdb1/da-06-dc3dd-fat32-512.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-fat32-512.log device size: 39179952 sectors (probed) sector size: 262144 bytes (set) 20060135424 bytes (19 G) copied (100%), 883.188 s, 22 M/s input results for device `/dev/sdc': 76523 sectors + 90112 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-fat32-512.111': 76523 sectors + 90112 bytes out 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.001' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.002' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.003' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.004' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.005' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.006' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.007' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.008' 8192 sectors out to `/media/sdb1/da-06-dc3dd-fat32-512.009' 2795 sectors + 90112 bytes out to `/media/sdb1/da-06-dc3dd-fat32-512.010' dc3dd completed at 2012-08-04 22:44:03 -0300 real 14m43.254s user 5m49.594s sys 0m50.347s

Tabela D-7: DA-06-DC3DD-EXT3

Item	Descrição
Identificação do Teste	DA-06-DC3DD-EXT3
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando DC3DD com sistema de arquivos destino EXT3, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, DC3DD.
Dispositivos	Origem: 01-PATA Destino:02-SATA
Variações	O teste será repetido com variação do parâmetro bufsz e ssz, com os valores 32K, 64K, 128K e 256K, equivalente em bytes a 64, 128, 256 e 512 setores, respectivamente.
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-03 mount -o rw –t ext3 /dev/sdb1 /media/sdb1

Variação: bufsz e ssz =32K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=32K ssz =32K verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-64.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-64.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=32k ssz=32k verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-64.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-64.log device size: 39179952 sectors (probed) sector size: 32768 bytes (set) 20060135424 bytes (19 G) copied (100%), 888.343 s, 22 M/s input results for device `/dev/sdc': 612186 sectors + 24576 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ext3-64.111': 612186 sectors + 24576 bytes out 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.001' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.002' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.003' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.004' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.005' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.006' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.007' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.008'

	65536 sectors out to `/media/sdb1/da-06-dc3dd-ext3-64.009' 22362 sectors + 24576 bytes out to `/media/sdb1/da-06-dc3dd-ext3-64.010' dc3dd completed at 2012-08-02 21:17:53 -0300 real 14m48.367s user 6m4.727s sys 1m43.446s

Variação: bufsz e ssz =64K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=64K ssz=64k verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-128.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-128.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=64k ssz=64k verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-128.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-128.log device size: 39179952 sectors (probed) sector size: 65536 bytes (set) 20060135424 bytes (19 G) copied (100%), 884.2 s, 22 M/s input results for device `/dev/sdc': 306093 sectors + 24576 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ext3-128.111': 306093 sectors + 24576 bytes out 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.001' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.002' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.003' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.004' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.005' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.006' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.007' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.008' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ext3-128.009' 11181 sectors + 24576 bytes out to `/media/sdb1/da-06-dc3dd-ext3-128.010' dc3dd completed at 2012-08-02 21:32:37 -0300 real 14m44.258s user 5m38.985s sys 1m15.957s

Variação: bufsz e ssz=128K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=128K ssz=128k verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-256.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-256.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=128k ssz=128k verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-256.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-256.log device size: 39179952 sectors (probed) sector size: 131072 bytes (set) 20060135424 bytes (19 G) copied (100%), 883.197 s, 22 M/s input results for device `/dev/sdc': 153046 sectors + 90112 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ext3-256.111': 153046 sectors + 90112 bytes out 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.001' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.002' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.003' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.004' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.005' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.006' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.007' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.008' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ext3-256.009' 5590 sectors + 90112 bytes out to `/media/sdb1/da-06-dc3dd-ext3-256.010' dc3dd completed at 2012-08-02 21:47:21 -0300 real 14m43.564s user 5m48.478s sys 1m3.396s

Variação: bufsz e ssz=256K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=256K ssz=256k verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-512.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-512.log**

Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=256k ssz=256k verb=on ofs=/media/sdb1/da-06-dc3dd-ext3-512.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ext3-512.log device size: 39179952 sectors (probed) sector size: 262144 bytes (set) 20060135424 bytes (19 G) copied (100%), 883.194 s, 22 M/s input results for device `/dev/sdc': 76523 sectors + 90112 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ext3-512.111': 76523 sectors + 90112 bytes out 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.001' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.002' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.003' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.004' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.005' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.006' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.007' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.008' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ext3-512.009' 2795 sectors + 90112 bytes out to `/media/sdb1/da-06-dc3dd-ext3-512.010' dc3dd completed at 2012-08-02 22:02:04 -0300 real 14m43.263s user 5m43.121s sys 0m58.260s

Tabela D-8: DA-06-DC3DD-NTFS

Item	Descrição
Identificação do Teste	DA-06-DC3DD-NTFS
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando DC3DD com sistema de arquivos destino NTFS, com variação de número de setores por bloco.
Ferramentas necessárias	Fdisk, Mount, DC3DD.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Variações	O teste será repetido com variação do parâmetro bufsz e ssz, com os valores 32K, 64K, 128K e 256K, equivalente em bytes a 64, 128, 256 e 512 setores, respectivamente.
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Variação: bufsz e ssz=32K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=32K ssz=32k verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-64.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-64.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=32k ssz=32k verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-64.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-64.log device size: 39179952 sectors (probed) sector size: 32768 bytes (set) 20060135424 bytes (19 G) copied (100%), 900.353 s, 21 M/s input results for device `/dev/sdc': 612186 sectors + 24576 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ntfs-64.111': 612186 sectors + 24576 bytes out 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.001' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.002' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.003' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.004' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.005' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.006' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.007' 65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.008'

	65536 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-64.009' 22362 sectors + 24576 bytes out to `/media/sdb1/da-06-dc3dd-ntfs-64.010' dc3dd completed at 2012-08-02 15:07:12 -0300 real 15m0.380s user 5m51.786s sys 1m29.430s	

Variação: bufsz e ssz =64K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=64K ssz=64K verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-128.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-128.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=64k ssz=64k verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-128.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-128.log device size: 39179952 sectors (probed) sector size: 65536 bytes (set) 20060135424 bytes (19 G) copied (100%), 893.422 s, 21 M/s input results for device `/dev/sdc': 306093 sectors + 24576 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ntfs-128.111': 306093 sectors + 24576 bytes out 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.001' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.002' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.003' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.004' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.005' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.006' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.007' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.008' 32768 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-128.009' 11181 sectors + 24576 bytes out to `/media/sdb1/da-06-dc3dd-ntfs-128.010' dc3dd completed at 2012-08-02 15:22:06 -0300 real 14m53.461s user 5m57.490s sys 1m4.956s

Variação: bufsz e ssz =128K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=128K ssz=128K verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-256.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-256.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=128k ssz=128k verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-256.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-256.log device size: 39179952 sectors (probed) sector size: 131072 bytes (set) 20060135424 bytes (19 G) copied (100%), 888.987 s, 22 M/s input results for device `/dev/sdc': 153046 sectors + 90112 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ntfs-256.111': 153046 sectors + 90112 bytes out 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.001' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.002' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.003' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.004' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.005' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.006' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.007' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.008' 16384 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-256.009' 5590 sectors + 90112 bytes out to `/media/sdb1/da-06-dc3dd-ntfs-256.010' dc3dd completed at 2012-08-02 15:36:55 -0300 real 14m48.993s user 5m54.970s sys 0m51.907s

Variação: bufsz e ssz =256K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=256K ssz=256k verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-512.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-512.log**

Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=256k ssz=256k verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-512.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-512.log device size: 39179952 sectors (probed) sector size: 262144 bytes (set) 20060135424 bytes (19 G) copied (100%), 883.182 s, 22 M/s input results for device `/dev/sdc': 76523 sectors + 90112 bytes in 0 bad sectors replaced by zeros ef40e6dbf01a843a689cdc83d3aaec8f (md5) f4d3c999e3df3c5c0a1937958fd016f4518bdf57 (sha1) 868db4e25b89e876ccee764ec71b5eacccbb28ace52494bee99e6eff2b2357be (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ntfs-512.111': 76523 sectors + 90112 bytes out 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.001' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.002' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.003' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.004' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.005' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.006' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.007' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.008' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512.009' 2795 sectors + 90112 bytes out to `/media/sdb1/da-06-dc3dd-ntfs-512.010' dc3dd completed at 2012-08-02 15:51:38 -0300 real 14m43.219s user 5m50.442s sys 0m43.347s

Tabela D-9: DA-06-DCFLDD-FAT32

Item	Descrição
Identificação do Teste	DA-06- DCFLDD-FAT32
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando DCFLDD com sistema de arquivos destino FAT32, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, DCFLDD.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Variações	O teste será repetido com variação do parâmetro bs, com os valores 32K, 64K, 128K e 256K, equivalente em bytes a 64, 128, 256 e 512 setores, respectivamente.
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-02 sudo mount -o rw –t vfat /dev/sdb1 /media/sdb1

Variação: bs=32K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=32K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-fat32-64.log md5log=/media/sdb1/da-06-dcfldd-fat32-64-md5.log sha1log=/media/sdb1/da-06-dcfldd-fat32-64-sha1-sha1.log sha256log=/media/sdb1/da-06-dcfldd-fat32-64-sha256.log of=/media/sdb1/da-06-dcfldd-fat32-64**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m43.973s user 7m46.889s sys 0m47.039s

Variação: bs=64K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=64K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-fat32-128.log md5log=/media/sdb1/da-06-dcfldd-fat32-128-md5.log sha1log=/media/sdb1/da-06-dcfldd-fat32-128-sha1.log sha256log=/media/sdb1/da-06-dcfldd-fat32-128-sha256.log of=/media/sdb1/da-06-dcfldd-fat32-128**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m44.476s user 7m48.269s sys 0m46.635s

Variação: bs=128K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=128K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-fat32-256.log md5log=/media/sdb1/da-06-dcfldd-fat32-256-md5.log sha1log=/media/sdb1/da-06-dcfldd-fat32-256-sha1.log sha256log=/media/sdb1/da-06-dcfldd-fat32-256-sha256.log of=/media/sdb1/da-06-dcfldd-fat32-256**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m44.305s user 7m44.929s sys 0m46.555s

Variação: bs=256K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=256K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-fat32-512.log md5log=/media/sdb1/da-06-dcfldd-fat32-512-md5.log sha1log=/media/sdb1/da-06-dcfldd-fat32-512-sha1.log sha256log=/media/sdb1/da-06-dcfldd-fat32-512-sha256.log of=/media/sdb1/da-06-dcfldd-fat32-512**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m44.122s user 7m40.789s sys 0m46.731s

Tabela D-10: DA-06-DCFLDD-EXT3

Item	Descrição
Identificação do Teste	DA-06-DCFLDD-EXT3
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando DCFLDD com sistema de arquivos destino EXT3, com variação de número de setores por bloco.
Ferramentas necessárias	Fdisk, Mount, DCFLDD.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Variações	O teste será repetido com variação do parâmetro bs, com os valores 32K, 64K, 128K e 256K, equivalente em bytes a 64, 128, 256 e 512 setores, respectivamente.
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-02 sudo mount -o rw –t ext3 /dev/sdb1 /media/sdb1

Variação: bs=32K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=32K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ext3-64.log md5log=/media/sdb1/da-06-dcfldd-ext3-64-md5.log sha1log=/media/sdb1/da-06-dcfldd-ext3-64-sha1-sha1.log sha256log=/media/sdb1/da-06-dcfldd-ext3-64-sha256.log of=/media/sdb1/da-06-dcfldd-ext3-64**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m50.445s user 7m40.141s sys 0m56.936s

Variação: bs=64K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=64K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ext3-128.log md5log=/media/sdb1/da-06-dcfldd-ext3-128-md5.log sha1log=/media/sdb1/da-06-dcfldd-ext3-128-sha1.log sha256log=/media/sdb1/da-06-dcfldd-ext3-128-sha256.log of=/media/sdb1/da-06-dcfldd-ext3-128**

Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m50.958s user 7m40.257s sys 0m57.088s

Variação: bs=128K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=128K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ext3-256.log md5log=/media/sdb1/da-06-dcfldd-ext3-256-md5.log sha1log=/media/sdb1/da-06-dcfldd-ext3-256-sha1.log sha256log=/media/sdb1/da-06-dcfldd-ext3-256-sha256.log of=/media/sdb1/da-06-dcfldd-ext3-256**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m50.473s user 7m35.732s sys 0m56.420s

Variação: bs=256K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=256K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ext3-512.log md5log=/media/sdb1/da-06-dcfldd-ext3-512-md5.log sha1log=/media/sdb1/da-06-dcfldd-ext3-512-sha1.log sha256log=/media/sdb1/da-06-dcfldd-ext3-512-sha256.log of=/media/sdb1/da-06-dcfldd-ext3-512**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m50.059s user 7m44.789s sys 0m58.180s

Tabela D-11: DA-06-DCFLDD-NTFS

Item	Descrição
Identificação do Teste	DA-06-DCFLDD-NTFS
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando DCFLDD com sistema de arquivos destino NTFS, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, DCFLDD.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Variações	O teste será repetido com variação do parâmetro bs, com os valores 32K, 64K, 128K e 256K, equivalente em bytes a 64, 128, 256 e 512 setores, respectivamente.

Variação: bs=32K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror bs=32K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ntfs-64.log md5log=/media/sdb1/da-06-dcfldd-ntfs-64.log sha1log=/media/sdb1/da-06-dcfldd-ntfs-64.log sha256log=/media/sdb1/da-06-dcfldd-ntfs-64.log of=/media/sdb1/da-06-dcfldd-ntfs-64**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m58.092s user 8m14.159s sys 0m52.615s

Variação: bs=64K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=64K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ntfs-128.log md5log=/media/sdb1/da-06-dcfldd-ntfs-128.log sha1log=/media/sdb1/da-06-dcfldd-ntfs-128.log sha256log=/media/sdb1/da-06-dcfldd-ntfs-128.log of=/media/sdb1/da-06-dcfldd-ntfs-128**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m59.678s user 8m14.635s sys 0m52.547s

Variação: bs=128K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=128K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ntfs-256.log md5log=/media/sdb1/da-06-dcfldd-ntfs-256.log sha1log=/media/sdb1/da-06-dcfldd-ntfs-256.log sha256log=/media/sdb1/da-06-dcfldd-ntfs-256.log of=/media/sdb1/da-06-dcfldd-ntfs-256**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 14m59.250s user 8m12.143s sys 0m53.139s

Variação: bs=256K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=256K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ntfs-512.log md5log=/media/sdb1/da-06-dcfldd-ntfs-512.log sha1log=/media/sdb1/da-06-dcfldd-ntfs-512.log sha256log=/media/sdb1/da-06-dcfldd-ntfs-512.log of=/media/sdb1/da-06-dcfldd-ntfs-512**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 15m1.893s user 8m14.435s sys 0m53.403s

Tabela D-12: DA-06-LINEN-FAT32

Item	Descrição
Identificação do Teste	DA-06-LINEN-FAT32
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando LINEN com sistema de arquivos destino FAT32, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, LINEN.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-02 sudo mount -o rw –t vfat /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros -g e -b, com os valores 64, 128, 256 e 512.

Variação: Tamanho de Bloco e Granularidade = 64.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-fat32-64.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-fat32-64.cfg	Device=/dev/sdc EvidencePath=/media/sdb1/da-06-linen-fat32-64.e01 EvidenceName= da-06-linen-fat32-64 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste Compress=0 Granularity=64 BlockSize=64 Hash=TRUE Sha1=TRUE CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-fat32-64.e01" Elapsed time: 0:20:38 MD5 Value: EF40E6DBF01A843A689CDC83D3AAEC8F SHA1 Value: F4D3C999E3DF3C5C0A1937958FD016F4518BDF57 real 20m38.704s user 3m28.341s sys 1m31.230s

Variação: Tamanho de Bloco e Granularidade = 128.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-fat32-128.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-fat32-128.cfg	Device=/dev/sdc EvidencePath=/media/sdb1/da-06-linen-fat32-128.e01 EvidenceName= da-06-linen-fat32-128 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste Compress=0 Granularity=128 BlockSize=128 Hash=TRUE Sha1=TRUE CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-fat32-128.e01" Elapsed time: 0:20:21 MD5 Value: EF40E6DBF01A843A689CDC83D3AAEC8F SHA1 Value: F4D3C999E3DF3C5C0A1937958FD016F4518BDF57 da-06-linen-fat32-256 real 20m21.383s user 3m22.037s sys 1m25.961s

Variação: Tamanho de Bloco e Granularidade = 256.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-fat32-256.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-fat32-256.cfg	Device=/dev/sdc EvidencePath=/media/sdb1/da-06-linen-fat32-256.e01 EvidenceName= da-06-linen-fat32-256 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste Compress=0 Granularity=256 BlockSize=256 Hash=TRUE Sha1=TRUE

	CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-fat32-256.e01" Elapsed time: 0:19:38 MD5 Value: EF40E6DBF01A843A689CDC83D3AAEC8F SHA1 Value: F4D3C999E3DF3C5C0A1937958FD016F4518BDF57 da-06-linen-fat32-512 real 19m37.951s user 3m1.339s sys 1m27.833s

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-fat32-512.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-fat32-512.cfg	Device=/dev/sdc EvidencePath=/media/sdb1/da-06-linen-fat32-512.e01 EvidenceName= da-06-linen-fat32-512 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste Compress=0 Granularity=512 BlockSize=512 Hash=TRUE Sha1=TRUE CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-fat32-512.e01" Elapsed time: 0:18:34 MD5 Value: EF40E6DBF01A843A689CDC83D3AAEC8F SHA1 Value: F4D3C999E3DF3C5C0A1937958FD016F4518BDF57 real 18m33.995s user 3m5.040s sys 1m27.477s

Tabela D-13: DA-06-LINEN-EXT3

Item	Descrição
Identificação do Teste	DA-06-LINEN-EXT3
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando LINEN com sistema de arquivos destino EXT3, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, LINEN.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-03 sudo mount -o rw –t ext3 /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros -g e -b, com os valores 64, 128, 256 e 512.

Variação: Tamanho de Bloco e Granularidade = 64.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ext3-64.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ext3-64.cfg	EvidencePath=/media/sdb1/da-06-linen-ext3-64.e01 EvidenceName=da-06-linen-ext3-64 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=64 BlockSize=64 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ext3-64.e01" Elapsed time: 0:21:42 MD5 Value: 00315C752900A17AE684763159C44323 SHA1 Value: 762F1531ABEBB314A871548FFDB8F76D9681ADC9 real 21m42.437s user 3m35.525s sys 1m29.610s

Variação: Tamanho de Bloco e Granularidade = 128.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ext3-128.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ext3-128.cfg	EvidencePath=/media/sdb1/da-06-linen-ext3-128.e01 EvidenceName=da-06-linen-ext3-128 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=128 BlockSize=128 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ext3-128.e01" Elapsed time: 0:22:01 MD5 Value: 00315C752900A17AE684763159C44323 SHA1 Value: 762F1531ABEBB314A871548FFDB8F76D9681ADC9 real 22m0.770s user 3m30.957s sys 1m22.869s

Variação: Tamanho de Bloco e Granularidade = 256.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ext3-256.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ext3-256.cfg	EvidencePath=/media/sdb1/da-06-linen-ext3-256.e01 EvidenceName=da-06-linen-ext3-256 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=256 BlockSize=256 Hash=TRUE Sha1=TRUE

	Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ext3-256.e01" Elapsed time: 0:21:46 MD5 Value: 00315C752900A17AE684763159C44323 SHA1 Value: 762F1531ABEBB314A871548FFDB8F76D9681ADC9 real 21m46.045s user 3m8.108s sys 1m27.861s

Variação : Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ext3-512.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ext3-512.cfg	EvidencePath=/media/sdb1/da-06-linen-ext3-512.e01 EvidenceName=da-06-linen-ext3-512 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=512 BlockSize=512 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ext3-512.e01" Elapsed time: 0:21:49 MD5 Value: 00315C752900A17AE684763159C44323 SHA1 Value: 762F1531ABEBB314A871548FFDB8F76D9681ADC9 real 21m49.603s user 3m15.344s sys 1m26.733s

Tabela D-14: DA-06-LINEN-NTFS

Item	Descrição
Identificação do Teste	DA-06-LINEN-NTFS
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando LINEN com sistema de arquivos destino NTFS, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, LINEN.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros –b (tamanho de bloco) e –g (granularidade), com os valores 64, 128, 256 e 512.

Variação: Tamanho de Bloco e Granularidade = 64.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ntfs-64.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ntfs-64.cfg	EvidencePath=/media/sdb1/da-06-linen-ntfs-64.e01 EvidenceName=da-06-linen-ntfs-64 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=64 BlockSize=64 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALS
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ntfs-64.e01" Elapsed time: 0:37:56 MD5 Value: 180AD904A9B257344D9C3F39488A6BEA SHA1 Value: 1C3FE3D2D2C717BC74635E89A3FF75048281E883 real 37m55.866s user 3m36.022s sys 2m7.228s

Variação: Tamanho de Bloco e Granularidade = 128.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ntfs-128.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ntfs-128.cfg	Device=/dev/sdc EvidencePath=/media/sdb1/da-06-linen-ntfs-128.e01 EvidenceName= da-06-linen-ntfs-128 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste Compress=0 Granularity=128 BlockSize=128 Hash=TRUE Sha1=TRUE CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ntfs-128.e01" Elapsed time: 0:37:11 MD5 Value: 180AD904A9B257344D9C3F39488A6BEA SHA1 Value: 1C3FE3D2D2C717BC74635E89A3FF75048281E883 real 37m11.767s user 3m30.125s sys 1m57.387s

Variação: Tamanho de Bloco e Granularidade = 256.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ntfs-256.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ntfs-256.cfg	EvidencePath=/media/sdb1/da-06-linen-ntfs-256.e01 EvidenceName=da-06-linen-ntfs-256 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=256 BlockSize=256 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE

	AcquireMode=TRUE HashMode=FALS
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ntfs-256.e01" Elapsed time: 0:36:37 MD5 Value: 180AD904A9B257344D9C3F39488A6BEA SHA1 Value: 1C3FE3D2D2C717BC74635E89A3FF75048281E883 real 36m37.453s user 3m15.924s sys 2m2.188s

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ntfs-512.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ntfs-512.cfg	EvidencePath=/media/sdb1/da-06-linen-ntfs-512.e01 EvidenceName=da-06-linen-ntfs-512 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=512 BlockSize=512 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALS
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ntfs-512.e01" Elapsed time: 0:36:14 MD5 Value: 180AD904A9B257344D9C3F39488A6BEA SHA1 Value: 1C3FE3D2D2C717BC74635E89A3FF75048281E883 real 36m13.545s user 3m14.044s sys 1m59.831s

Tabela D-15: DA-06-EWFACQUIRE-FAT32

Item	Descrição
Identificação do Teste	DA-06-EWFACQUIRE-FAT32
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando EWFACQUIRE com sistema de arquivos destino FAT32, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, EWFACQUIRE.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-02 sudo mount -o rw –t vfat /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros -g e -b, com os valores 64, 128, 256 e 512.

Variação:Tamanho de Bloco e Granularidade = 64.

Comando Utilizado	**time ewfacquire -b 64 -c none -C 1 -d sha1 -D "fat32 nao comprimido 64 setores" -e daniel -E 1 -f encase6 -g 64 -l /media/sdb1/da-06-ewfacquire-fat32-64 -m fixed -M physical -N "fat32 nao comprimido 64 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-fat32-64 -uvw /dev/sdc**
Resultados	da-06-ewfacquire-fat32-64 ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. acquired 32 KiB (32768 bytes) of total 18 GiB (20060135424 bytes). (linhas suprimidas) Acquiry completed at: Sun Jul 22 19:59:14 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 5 second(s) with 21 MiB/s (22165895 bytes/second). MD5 hash calculated over data: ef40e6dbf01a843a689cdc83d3aaec8f SHA1 hash calculated over data:

	f4d3c999e3df3c5c0a1937958fd016f4518bdf57 real 15m5.090s user 2m5.024s sys 0m47.223s	

Variação:Tamanho de Bloco e Granularidade = 128.

Comando Utilizado	**time ewfacquire -b 128 -c none -C 1 -d sha1 -D "fat32 nao comprimido 128 setores" -e daniel -E 1 -f encase6 -g 128 -l /media/sdb1/da-06-ewfacquire-fat32-128 -m fixed -M physical -N "fat32 nao comprimido 128 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-fat32-128 -uvw /dev/sdc**	
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. acquired 64 KiB (65536 bytes) of total 18 GiB (20060135424 bytes). Acquiry completed at: Sun Jul 22 20:14:22 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 8 second(s) with 21 MiB/s (22092660 bytes/second). MD5 hash calculated over data: ef40e6dbf01a843a689cdc83d3aaec8f SHA1 hash calculated over data: f4d3c999e3df3c5c0a1937958fd016f4518bdf57 real 15m7.474s user 1m59.767s sys 0m45.351s	

Variação: Tamanho de Bloco e Granularidade = 256.

Comando Utilizado	**time ewfacquire -b 256 -c none -C 1 -d sha1 -D "fat32 nao comprimido 256 setores" -e daniel -E 1 -f encase6 -g 256 -l /media/sdb1/da-06-ewfacquire-fat32-256 -m fixed -M physical -N "fat32 nao comprimido 256 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-fat32-256 -uvw /dev/sdc**
Resultados	da-06-ewfacquire-fat32-256 ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. (linhas suprimidas) Acquiry completed at: Sun Jul 22 20:29:28 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 6 second(s) with 21 MiB/s (22141430 bytes/second). MD5 hash calculated over data: ef40e6dbf01a843a689cdc83d3aaec8f SHA1 hash calculated over data: f4d3c999e3df3c5c0a1937958fd016f4518bdf57 real 15m5.987s user 1m59.479s sys 0m46.763s

Variação:Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time ewfacquire -b 512 -c none -C 1 -d sha1 -D "fat32 nao comprimido 512 setores" -e daniel -E 1 -f encase6 -g 512 -l /media/sdb1/da-06-ewfacquire-fat32-512 -m fixed -M physical -N "fat32 nao comprimido 512 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-fat32-512 -uvw /dev/sdc**
Resultados	da-06-ewfacquire-fat32-512 ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid)

Media information:
Device type: Direct access
Bus type: ATA/ATAPI
Vendor: ATA
Model: SAMSUNG SV0221N
Serial: S01NJ10WC00253
Media size: 20 GB (20060135424 bytes)

Status: at 0%.

(linhas suprimidas)

Acquiry completed at: Sun Jul 22 20:44:32 2012

Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 4 second(s) with 21 MiB/s (22190415 bytes/second).
MD5 hash calculated over data: ef40e6dbf01a843a689cdc83d3aaec8f
SHA1 hash calculated over data: f4d3c999e3df3c5c0a1937958fd016f4518bdf57

real 15m4.259s
user 2m1.996s
sys 0m48.355s

Tabela D-16: DA-06-EWFACQUIRE-EXT3

Item	Descrição
Identificação do Teste	DA-06-EWFACQUIRE-EXT3
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando EWFACQUIRE com sistema de arquivos destino EXT3, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, EWFACQUIRE.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-03 sudo mount -o rw –t ext3 /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros -g e -b, com os valores 64, 128, 256 e 512.

Variação: Tamanho de Bloco e Granularidade = 64.

Comando Utilizado	**time ewfacquire -b 64 -c none -C 1 -d sha1 -D "ext3 nao comprimido 64 setores" -e daniel -E 1 -f encase6 -g 64 -l /media/sdb1/da-06-ewfacquire-ext3-64 -m fixed -M physical -N "ext3 nao comprimido 64 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ext3-64 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. Acquiry completed at: Mon Jul 23 01:18:01 2012 (linhas suprimidas) Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 8 second(s) with 21 MiB/s (22092660 bytes/second). MD5 hash calculated over data: 00315c752900a17ae684763159c44323 SHA1 hash calculated over data: 762f1531abebb314a871548ffdb8f76d9681adc9

	real 15m8.337s user 2m6.972s sys 0m59.564s

Variação: Tamanho de Bloco e Granularidade = 128.

Comando Utilizado	**time ewfacquire -b 128 -c none -C 1 -d sha1 -D "ext3 nao comprimido 128 setores" -e daniel -E 1 -f encase6 -g 128 -l /media/sdb1/da-06-ewfacquire-ext3-128 -m fixed -M physical -N "ext3 nao comprimido 128 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ext3-128 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. (linhas suprimidas) Acquiry completed at: Mon Jul 23 01:33:12 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 10 second(s) with 21 MiB/s (22044105 bytes/second). MD5 hash calculated over data: 00315c752900a17ae684763159c44323 SHA1 hash calculated over data: 762f1531abebb314a871548ffdb8f76d9681adc9 real 15m10.422s user 2m8.464s sys 0m57.664s

Variação: Tamanho de Bloco e Granularidade = 256.

Comando Utilizado	**time ewfacquire -b 256 -c none -C 1 -d sha1 -D "ext3 nao comprimido 256 setores" -e daniel -E 1 -f encase6 -g 256 -l /media/sdb1/da-06-ewfacquire-ext3-256 -m fixed -M physical -N "ext3 nao comprimido 256 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ext3-256 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114,

	zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. (linhas suprimidas) Acquiry completed at: Mon Jul 23 01:48:26 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 14 second(s) with 20 MiB/s (21947632 bytes/second). MD5 hash calculated over data: 00315c752900a17ae684763159c44323 SHA1 hash calculated over data: 762f1531abebb314a871548ffdb8f76d9681adc9 real 15m14.505s user 2m5.580s sys 0m58.040s

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time ewfacquire -b 512 -c none -C 1 -d sha1 -D "ext3 nao comprimido 512 setores" -e daniel -E 1 -f encase6 -g 512 -l /media/sdb1/da-06-ewfacquire-ext3-512 -m fixed -M physical -N "ext3 nao comprimido 512 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ext3-512 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes)

	Status: at 0%. (linhas suprimidas) Acquiry completed at: Mon Jul 23 02:03:39 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 12 second(s) with 20 MiB/s (21995762 bytes/second). MD5 hash calculated over data: 00315c752900a17ae684763159c44323 SHA1 hash calculated over data: 762f1531abebb314a871548ffdb8f76d9681adc9 real 15m13.119s user 2m3.948s sys 0m56.544s	

Tabela D-17: DA-06-EWFACQUIRE-NTFS

Item	Descrição
Identificação do Teste	DA-06-EWFACQUIRE-NTFS
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando EWFACQUIRE com sistema de arquivos destino NTFS, com variação de número de setores por bloco.
Ferramentas necessárias	Mount, EWFACQUIRE.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros –b (tamanho de bloco) e –g (granularidade), com os valores 64, 128, 256 e 512.

Variação: Tamanho de Bloco e Granularidade = 64.

Comando Utilizado	**time ewfacquire -b 64 -c none -C 1 -d sha1 -D "ntfs nao comprimido 64 setores" -e daniel -E 1 -f encase6 -g 64 -l /media/sdb1/da-06-ewfacquire-ntfs-64 -m fixed -M physical -N "ntfs nao comprimido 64 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ntfs-64 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) (linhas suprimidas) Acquiry completed at: Sat Jul 21 00:24:49 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 19 second(s) with 20 MiB/s (21828221 bytes/second). MD5 hash calculated over data: 180ad904a9b257344d9c3f39488a6bea SHA1 hash calculated over data: 1c3fe3d2d2c717bc74635e89a3ff75048281e883 real 15m18.803s user 2m34.906s sys 0m58.224s

Variação: Tamanho de Bloco e Granularidade = 128.

Comando Utilizado	**time ewfacquire -b 128 -c none -C 1 -d sha1 -D "ntfs nao comprimido 128**

	setores" -e daniel -E 1 -f encase6 -g 128 -l /media/sdb1/da-06-ewfacquire-ntfs-128 -m fixed -M physical -N "ntfs nao comprimido 128 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ntfs-128 -uvw /dev/sdc
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) (linhas suprimidas) Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 22 second(s) with 20 MiB/s (21757197 bytes/second). MD5 hash calculated over data: 180ad904a9b257344d9c3f39488a6bea SHA1 hash calculated over data: 1c3fe3d2d2c717bc74635e89a3ff75048281e883 real 15m22.537s user 2m31.973s sys 0m53.243s

Variação: Tamanho de Bloco e Granularidade = 256.

Comando Utilizado	**time ewfacquire -b 256 -c none -C 1 -d sha1 -D "ntfs nao comprimido 256 setores" -e daniel -E 1 -f encase6 -g 256 -l /media/sdb1/da-06-ewfacquire-ntfs-256 -m fixed -M physical -N "ntfs nao comprimido 256 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ntfs-256 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) (linhas suprimidas) Acquiry completed at: Sat Jul 21 00:55:26 2012

	Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 14 second(s) with 20 MiB/s (21947632 bytes/second). MD5 hash calculated over data: 180ad904a9b257344d9c3f39488a6bea SHA1 hash calculated over data: 1c3fe3d2d2c717bc74635e89a3ff75048281e883 real 15m14.483s user 2m28.441s sys 0m53.947s	

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time ewfacquire -b 512 -c none -C 1 -d sha1 -D "ntfs nao comprimido 512 setores" -e daniel -E 1 -f encase6 -g 512 -l /media/sdb1/da-06-ewfacquire-ntfs-512 -m fixed -M physical -N "ntfs nao comprimido 512 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ntfs-512 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Acquiry completed at: Sat Jul 21 01:10:36 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 10 second(s) with 21 MiB/s (22044105 bytes/second). MD5 hash calculated over data: 180ad904a9b257344d9c3f39488a6bea SHA1 hash calculated over data: 1c3fe3d2d2c717bc74635e89a3ff75048281e883 real 15m9.703s user 2m28.893s sys 0m52.883s

Tabela D-18: DA-06-FTKIMAGER-FAT32-512

Item	Descrição
Identificação do Teste	DA-06-FTKIMAGER-FAT32-512
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando FTKIMAGER com sistema de arquivos destino FAT32.
Ferramentas necessárias	Mount, FTKIMAGER.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-02 sudo mount -o rw –t vfat /dev/sdb1 /media/sdb1
Variações	A execução do FTKImager no ambiente Linux por linha de comando não oferece opção de variação de tamanho de setores por bloco. O valor usado pelo programa é de 512 setores por bloco.

Variação:Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time sudo /root/experimentos/ftkimager /dev/sdc /media/sdb1/da-06-ftkimager-fat32-512 --e01 --frag 2G --compress 0 --case-number 4 --evidence-number 2 --description "da-06-ftkimager-fat32-512 ftkimager split sem compressao" --examiner Daniel --notes "mais notes"**
Resultados	Created By AccessDataÂ® FTKÂ® Imager 2.9 100407 Case Information: Case Number: 4 Evidence Number: 2 Unique description: da-06-ftkimager-fat32-512 ftkimager split sem compressao Examiner: Daniel Notes: mais notes --- Information for /media/sdb1/da-06-ftkimager-fat32-512: Physical Evidentiary Item (Source) Information: [Drive Geometry] Cylinders: 2438 Heads: 255 Sectors per Track: 63 Bytes per Sector: 512 Sector Count: 39179952 [Physical Drive Information] Drive Model: ATA SAMSUNG SV0221N

	Drive Interface Type: SCSI Source data size: 19130 MB Sector count: 39179952 [Computed Hashes] MD5 checksum: ef40e6dbf01a843a689cdc83d3aaec8f SHA1 checksum: f4d3c999e3df3c5c0a1937958fd016f4518bdf57 Image Information: Acquisition started: Sun Jul 22 20:44:34 2012 Acquisition finished: Sun Jul 22 20:59:32 2012 Segment list: /media/sdb1/da-06-ftkimager-fat32-512.E01 /media/sdb1/da-06-ftkimager-fat32-512.E02 /media/sdb1/da-06-ftkimager-fat32-512.E03 /media/sdb1/da-06-ftkimager-fat32-512.E04 /media/sdb1/da-06-ftkimager-fat32-512.E05 /media/sdb1/da-06-ftkimager-fat32-512.E06 /media/sdb1/da-06-ftkimager-fat32-512.E07 /media/sdb1/da-06-ftkimager-fat32-512.E08 /media/sdb1/da-06-ftkimager-fat32-512.E09 /media/sdb1/da-06-ftkimager-fat32-512.E10 real 19m6.939s user 10m57.129s sys 1m34.638s	

Tabela D-19: DA-06-FTKIMAGER-EXT3-512

Item	Descrição
Identificação do Teste	DA-06-FTKIMAGER-EXT3-512
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando FTKIMAGER com sistema de arquivos destino EXT3.
Ferramentas necessárias	Mount, FTKIMAGER.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-03 sudo mount -o rw –t ext3 /dev/sdb1 /media/sdb1
Variações	A execução do FTKImager no ambiente Linux por linha de comando não oferece opção de variação de tamanho de setores por bloco. O valor usado pelo programa é de 512 setores por bloco.

Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time sudo /root/experimentos/ftkimager /dev/sdc /media/sdb1/da-06-ftkimager-ext3-512 --e01 --frag 2G --compress 0 --case-number 4 --evidence-number 2 --description "da-06-ftkimager-ext3-512 ftkimager split sem compressao" --examiner Daniel --notes "mais notes"**
Resultados	Created By AccessDataÂ® FTKÂ® Imager 2.9 100407 Case Information: Case Number: 4 Evidence Number: 2 Unique description: da-06-ftkimager-ext3-512 ftkimager split sem compressao Examiner: Daniel Notes: mais notes --- Information for /media/sdb1/da-06-ftkimager-ext3-512: Physical Evidentiary Item (Source) Information: [Drive Geometry] Cylinders: 2438 Heads: 255 Sectors per Track: 63 Bytes per Sector: 512 Sector Count: 39179952 [Physical Drive Information] Drive Model: ATA SAMSUNG SV0221N Drive Interface Type: SCSI

	Source data size: 19130 MB Sector count: 39179952 [Computed Hashes] MD5 checksum: 00315c752900a17ae684763159c44323 SHA1 checksum: 762f1531abebb314a871548ffdb8f76d9681adc9 Image Information: Acquisition started: Mon Jul 23 02:03:42 2012 Acquisition finished: Mon Jul 23 02:18:51 2012 Segment list: /media/sdb1/da-06-ftkimager-ext3-512.E01 /media/sdb1/da-06-ftkimager-ext3-512.E02 /media/sdb1/da-06-ftkimager-ext3-512.E03 /media/sdb1/da-06-ftkimager-ext3-512.E04 /media/sdb1/da-06-ftkimager-ext3-512.E05 /media/sdb1/da-06-ftkimager-ext3-512.E06 /media/sdb1/da-06-ftkimager-ext3-512.E07 /media/sdb1/da-06-ftkimager-ext3-512.E08 /media/sdb1/da-06-ftkimager-ext3-512.E09 /media/sdb1/da-06-ftkimager-ext3-512.E10 real 15m3.345s user 5m53.706s sys 1m26.625s	

Tabela D-20: DA-06-FTKIMAGER-NTFS-512

Item	Descrição
Identificação do Teste	DA-06-FTKIMAGER-NTFS-512
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando FTKIMAGER com sistema de arquivos destino NTFS.
Ferramentas necessárias	Mount, FTKIMAGER.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1
Variações	A execução do FTKImager no ambiente Linux por linha de comando não oferece opção de variação de tamanho de setores por bloco. O valor usado pelo programa é de 512 setores por bloco.

Tamanho de Bloco : 512.

Comando Utilizado	**time sudo /root/experimentos/ftkimager /dev/sdc /media/sdb1/da-06-ftkimager-ntfs-512 --e01 --frag 2G --compress 0 --case-number 4 --evidence-number 2 --description "da-06-ftkimager-ntfs-512 ftkimager split sem compressao" --examiner Daniel --notes "mais notes"**
Resultados	Created By AccessDataÂ® FTKÂ® Imager 2.9 100407 Case Information: Case Number: 4 Evidence Number: 2 Unique description: da-06-ftkimager-ntfs-512 ftkimager split sem compressao Examiner: Daniel Notes: mais notes --- Information for /media/sdb1/da-06-ftkimager-ntfs-512: Physical Evidentiary Item (Source) Information: [Drive Geometry] Cylinders: 2438 Heads: 255 Sectors per Track: 63 Bytes per Sector: 512 Sector Count: 39179952 [Physical Drive Information] Drive Model: ATA SAMSUNG SV0221N Drive Interface Type: SCSI Source data size: 19130 MB Sector count: 39179952 [Computed Hashes]

	MD5 checksum: 180ad904a9b257344d9c3f39488a6bea SHA1 checksum: 1c3fe3d2d2c717bc74635e89a3ff75048281e883 Image Information: Acquisition started: Sat Jul 21 15:58:01 2012 Acquisition finished: Sat Jul 21 16:13:05 2012 Segment list: /media/sdb1/da-06-ftkimager-ntfs-512.E01 /media/sdb1/da-06-ftkimager-ntfs-512.E02 /media/sdb1/da-06-ftkimager-ntfs-512.E03 /media/sdb1/da-06-ftkimager-ntfs-512.E04 /media/sdb1/da-06-ftkimager-ntfs-512.E05 /media/sdb1/da-06-ftkimager-ntfs-512.E06 /media/sdb1/da-06-ftkimager-ntfs-512.E07 /media/sdb1/da-06-ftkimager-ntfs-512.E08 /media/sdb1/da-06-ftkimager-ntfs-512.E09 /media/sdb1/da-06-ftkimager-ntfs-512.E10 AccessData FTK Imager v2.9 CLI (May 12 2010) Copyright 2006-2010 AccessData Corp., 384 South 400 West, Lindon, UT 84042 All rights reserved. real 14m58.541s user 18m27.613s sys 1m42.262s	

Tabela D-21: DA-10-LINEN-NTFS-512

Item	Descrição
Identificação do Teste	DA-10-LINEN-NTFS-512
Resumo do Teste	Adquirir uma fonte digital para um arquivo de imagem em um formato alternativo.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando LINEN com sistema de arquivos destino NTFS, com 512 setores por bloco e arquivo de imagem comprimido.
Ferramentas necessárias	Mount, LINEN.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros –b (tamanho de bloco) e –b (granularidade), com os valores 64, 128, 256 e 512.

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ntfs-512.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ntfs-512.cfg	EvidencePath=/media/sdb1/da-06-linen-ntfs-512.e01 EvidenceName=da-06-linen-ntfs-512 CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=512 BlockSize=512 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALS
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ntfs-512.e01" Elapsed time: 0:36:14 MD5 Value: 180AD904A9B257344D9C3F39488A6BEA SHA1 Value: 1C3FE3D2D2C717BC74635E89A3FF75048281E883 real 36m13.545s user 3m14.044s sys 1m59.831s

Tabela D-22: DA-10-EWFACQUIRE-NTFS-512

Item	Descrição
Identificação do Teste	DA-10-EWFACQUIRE-NTFS-512.
Resumo do Teste	Adquirir uma fonte digital para um arquivo de imagem em um formato alternativo.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando EWFACQUIRE com sistema de arquivos destino NTFS, com 512 setores por bloco e arquivo de imagem comprimido.
Ferramentas necessárias	Mount, EWFACQUIRE.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1
Variações	O teste será repetido com variação dos parâmetros –b (tamanho de bloco) e –g (granularidade), com os valores 64, 128, 256 e 512.

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time ewfacquire -b 512 -c best -C 1 -d sha1 -D "ntfs comprimido 512 setores" -e daniel -E 1 -f encase6 -g 512 -l /media/sdb1/da-10-ewfacquire-ntfs-512 -m fixed -M physical -N "ntfs comprimido 512 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-10-ewfacquire-ntfs-512 -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. (linhas suprimidas) Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 16 second(s) with 20 MiB/s (21899711 bytes/second). MD5 hash calculated over data: ef40e6dbf01a843a689cdc83d3aaec8f SHA1 hash calculated over data: f4d3c999e3df3c5c0a1937958fd016f4518bdf57 real 14m57.292s user 13m3.745s sys 0m22.817s

Tabela D-23: DA-10-FTKIMAGER-NTFS-512

Item	Descrição
Identificação do Teste	DA-10-FTKIMAGER-NTFS-512
Resumo do Teste	Adquirir uma fonte digital para um arquivo de imagem em um formato alternativo.
Comentário	Duplicação Forense do disco rígido de origem para disco rígido de destino, usando FTKIMAGER com sistema de arquivos destino NTFS, com 512 setores por bloco e arquivo de imagem comprimido..
Ferramentas necessárias	Mount, FTKIMAGER.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1
Variações	A execução do FTKImager no ambiente Linux por linha de comando não oferece opção de variação de tamanho de setores por bloco. O valor usado pelo programa é de 512 setores por bloco.

Tamanho de Bloco: 512.

Comando Utilizado	**time sudo /root/experimentos/ftkimager /dev/sdc /media/sdb1/da-10-ftkimager-ntfs-512 --e01 --frag 2G --compress 9 --case-number 4 --evidence-number 4 --description "da-10-ftkimager-ntfs-512 ftkimager split com compressao" --examiner Daniel --notes "mais notes"**
Resultados	Created By AccessDataÂ® FTKÂ® Imager 2.9 100407 Case Information: Case Number: 4 Evidence Number: 4 Unique description: da-10-ftkimager-ntfs-512 ftkimager split com compressao Examiner: Daniel Notes: mais notes --- Information for /media/sdb1/da-10-ftkimager-ntfs-512: Physical Evidentiary Item (Source) Information: [Drive Geometry] Cylinders: 2438 Heads: 255 Sectors per Track: 63 Bytes per Sector: 512 Sector Count: 39179952 [Physical Drive Information] Drive Model: ATA SAMSUNG SV0221N Drive Interface Type: SCSI

	Source data size: 19130 MB Sector count: 39179952 [Computed Hashes] MD5 checksum: 180ad904a9b257344d9c3f39488a6bea SHA1 checksum: 1c3fe3d2d2c717bc74635e89a3ff75048281e883 Image Information: Acquisition started: Sat Jul 21 17:02:20 2012 Acquisition finished: Sat Jul 21 17:17:04 2012 Segment list: /media/sdb1/da-10-ftkimager-ntfs-512.E01 real 16m26.879s user 19m0.051s sys 1m2.940s	

Tabela D-24: DA-06-DC3DD-NTFS-512-CRIPTO

Item	Descrição
Identificação do Teste	DA-06-DC3DD-NTFS-512-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem cripotgrafado para disco rígido de destino, usando DC3DD com sistema de arquivos destino NTFS, com 512 setores por bloco.
Ferramentas necessárias	Fdisk, Mount, DC3DD.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Variação: bufsz e ssz=256K

Comando Utilizado	**time dc3dd if=/dev/sdc bufsz=256K ssz=256K verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-512-cripto.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-512-cripto.log**
Resultados Obtidos	command line: dc3dd if=/dev/sdc bufsz=256K ssz=256K verb=on ofs=/media/sdb1/da-06-dc3dd-ntfs-512-cripto.111 ofsz=2G hash=md5 hash=sha1 hash=sha256 log=/media/sdb1/da-06-dc3dd-ntfs-512-cripto.log device size: 39179952 sectors (probed) sector size: 262144 bytes (set) 20060135424 bytes (19 G) copied (100%), 883.148 s, 22 M/s input results for device `/dev/sdc': 76523 sectors + 90112 bytes in 0 bad sectors replaced by zeros c038293d76e6e1375d7c64ad7b59c59d (md5) ad9a619758497b3073b2efad10abf646e1409c4e (sha1) e48a5dd8759287d94843d5becfefd51deb4f55a4ec394a29de1f25f189a66172 (sha256)  output results for files `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.111': 76523 sectors + 90112 bytes out 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.001' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.002' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.003' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.004' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.005' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.006' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.007' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.008' 8192 sectors out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.009'

	2795 sectors + 90112 bytes out to `/media/sdb1/da-06-dc3dd-ntfs-512-cripto.010' dc3dd completed at 2012-08-06 18:27:36 -0300 real 14m43.171s user 5m52.606s sys 0m41.727s	

Tabela D-25: DA-06-DCFLDD-NTFS-512-CRIPTO

Item	Descrição
Identificação do Teste	DA-06-DCFLDD-NTFS-512-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem criptografado para disco rígido de destino, usando DCFLDD com sistema de arquivos destino NTFS, com 512 setores por bloco.
Ferramentas necessárias	Mount, DCFLDD.
Dispositivos	Origem: 01-PATA Destino: 02-SATA

Variação: bs=256K

Comando Utilizado	**time dcfldd if=/dev/sdc conv=sync,noerror split=2G bs=256K hash=md5,sha1,sha256 errlog=/media/sdb1/da-06-dcfldd-ntfs-512-cripto.log md5log=/media/sdb1/da-06-dcfldd-ntfs-512-cripto-md5.log sha1log=/media/sdb1/da-06-dcfldd-ntfs-512-cripto-sha1.log sha256log=/media/sdb1/da-06-dcfldd-ntfs-512-cripto-sha256.log of=/media/sdb1/da-06-dcfldd-ntfs-cripto-512**
Resultados Obtidos	612096 blocks (19128Mb) written. 612186+1 records in 612187+0 records out real 15m7.720s user 8m14.703s sys 0m51.955s

Tabela D-26: DA-06-LINEN-NTFS-512-CRIPTO

Item	Descrição
Identificação do Teste	DA-06-LINEN-NTFS-512-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem criptografado para disco rígido de destino, usando LINEN com sistema de arquivos destino NTFS, com 512 setores por bloco.
Ferramentas necessárias	Mount, LINEN.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-06-linen/da-06-linen-ntfs-512-cripto.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ntfs-512.cfg	EvidencePath=/media/sdb1/da-06-linen-ntfs-512-cripto.e01 EvidenceName=da-06-linen-ntfs-512-cripto CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=0 Granularity=512 BlockSize=512 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-06-linen-ntfs-512-cripto.e01" Elapsed time: 0:36:12 MD5 Value: 4F9178EDA6C1B39D504F4BFBAA37EFD9 SHA1 Value: 5DFB7557923FA889C277CD22BFEBD1DBA1B40CEE real 36m11.694s user 3m14.864s sys 1m58.143s

Tabela D-27: DA-10-LINEN-NTFS-512-CRIPTO

Item	Descrição
Identificação do Teste	DA-10-LINEN-NTFS-512-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem criptografado para disco rígido de destino, usando LINEN com sistema de arquivos destino NTFS, com 512 setores por bloco.
Ferramentas necessárias	Mount, LINEN.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time /root/experimentos/linen -j /root/testes_de_caso/da-10-linen/da-01-linen-ntfs-512-cripto.cfg**
Conteúdo do Arquivo de Configuração da-06-linen-ntfs-512.cfg	EvidencePath=/media/sdb1/da-10-linen-ntfs-512-cripto.e01 EvidenceName=da-10-linen-ntfs-512-cripto CaseNumber=1 Examiner=Daniel EvidenceNumber=1 alternatepath= Notes=teste MaxfileSize=2000 Compress=2 Granularity=512 BlockSize=512 Hash=TRUE Sha1=TRUE Device=/dev/sdc CommandLine=TRUE AcquireMode=TRUE HashMode=FALSE
Resultados	"/dev/sdc" acquired to "/media/sdb1/da-10-linen-ntfs-512-cripto.e01" Elapsed time: 0:36:57 MD5 Value: 4F9178EDA6C1B39D504F4BFBAA37EFD9 SHA1 Value: 5DFB7557923FA889C277CD22BFEBD1DBA1B40CEE real 36m56.864s user 27m50.008s sys 1m55.023s

Tabela D-28: DA-06-EWFACQUIRE-NTFS-CRIPTO

Item	Descrição
Identificação do Teste	DA-06-EWFACQUIRE-NTFS-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem criptografado para disco rígido de destino, usando EWFACQUIRE com sistema de arquivos destino NTFS, com 512 setores por bloco.
Ferramentas necessárias	Mount, EWFACQUIRE.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Variação: Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time ewfacquire -b 512 -c none -C 1 -d sha1 -D "ntfs origem cripto nao comprimido 512 setores" -e daniel -E 1 -f encase6 -g 512 -l /media/sdb1/da-06-ewfacquire-ntfs-512-cripto -m fixed -M physical -N "ntfs nao comprimido 512 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-06-ewfacquire-ntfs-512-cripto -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. (linhas suprimidas) Acquiry completed at: Sun Jul 22 14:06:37 2012 Written: 18 GiB (20060135768 bytes) in 15 minute(s) and 23 second(s) with 20 MiB/s (21733624 bytes/second). MD5 hash calculated over data: 4f9178eda6c1b39d504f4bfbaa37efd9 SHA1 hash calculated over data: 5dfb7557923fa889c277cd22bfebd1dba1b40cee real 15m24.272s user 2m29.509s sys 0m51.251s

Tabela D-29: DA-10-EWFACQUIRE-NTFS-CRIPTO

Item	Descrição
Identificação do Teste	DA-06-EWFACQUIRE-NTFS-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem criptografado para disco rígido de destino, usando EWFACQUIRE com sistema de arquivos destino NTFS, com 512 setores por bloco, com compressão do arquivo de imagem.
Ferramentas necessárias	Mount, EWFACQUIRE.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Tamanho de Bloco e Granularidade = 512.

Comando Utilizado	**time ewfacquire -b 512 -c best -C 1 -d sha1 -D "ntfs origem cripto comprimido 512 setores" -e daniel -E 1 -f encase6 -g 512 -l /media/sdb1/da-10-ewfacquire-ntfs-512-cripto -m fixed -M physical -N "ntfs comprimido 512 setores" -o 0 -r 2 -S 2G -t /media/sdb1/da-10-ewfacquire-ntfs-512-cripto -uvw /dev/sdc**
Resultados	ewfacquire 20100119 (libewf 20100119, libuna 20091031, libbfio 20091114, zlib 1.2.3.3, libcrypto 0.9.8, libuuid) Media information: Device type: Direct access Bus type: ATA/ATAPI Vendor: ATA Model: SAMSUNG SV0221N Serial: S01NJ10WC00253 Media size: 20 GB (20060135424 bytes) Status: at 0%. (linhas suprimidas) Acquiry completed at: Sun Jul 22 14:28:00 2012 Written: 18 GiB (20060135768 bytes) in 21 minute(s) and 22 second(s) with 14 MiB/s (15647531 bytes/second). MD5 hash calculated over data: 4f9178eda6c1b39d504f4bfbaa37efd9 SHA1 hash calculated over data: 5dfb7557923fa889c277cd22bfebd1dba1b40cee real 21m22.619s user 16m27.530s sys 0m57.468s

Tabela D-30: DA-06-FTKIMAGER-NTFS-512-CRIPTO

Item	Descrição
Identificação do Teste	DA-06-FTKIMAGER-NTFS-512-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem criptografado para disco rígido de destino, usando FTKIMAGER com sistema de arquivos destino NTFS, com 512 setores por bloco.
Ferramentas necessárias	Mount, FTKIMAGER.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Tamanho de Bloco: 512.

Comando Utilizado	**time sudo /root/experimentos/ftkimager /dev/sdc /media/sdb1/da-06-ftkimager-ntfs-512-cripto --e01 --frag 2G --compress 0 --case-number 4 --evidence-number 4 --description "da-10-ftkimager-ntfs-512 ftkimager split cripto sem compressao" --examiner Daniel --notes "mais notes"**
Resultados	Created By AccessDataÂ® FTKÂ® Imager 2.9 100407 Case Information: Case Number: 4 Evidence Number: 4 Unique description: da-06-ftkimager-ntfs-512 ftkimager split cripto sem compressao Examiner: Daniel Notes: mais notes --- Information for /media/sdb1/da-06-ftkimager-ntfs-512-cripto: Physical Evidentiary Item (Source) Information: [Drive Geometry] Cylinders: 2438 Heads: 255 Sectors per Track: 63 Bytes per Sector: 512 Sector Count: 39179952 [Physical Drive Information] Drive Model: ATA SAMSUNG SV0221N Drive Interface Type: SCSI Source data size: 19130 MB Sector count: 39179952

	[Computed Hashes] MD5 checksum: 4f9178eda6c1b39d504f4bfbaa37efd9 SHA1 checksum: 5dfb7557923fa889c277cd22bfebd1dba1b40cee Image Information: Acquisition started: Sun Jul 22 14:28:02 2012 Acquisition finished: Sun Jul 22 14:43:27 2012 Segment list: /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E01 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E02 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E03 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E04 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E05 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E06 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E07 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E08 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E09 /media/sdb1/da-06-ftkimager-ntfs-512-cripto.E10 real 15m3.620s user 6m5.103s sys 1m25.173s	

Tabela D-31: DA-10-FTKIMAGER-NTFS-512-CRIPTO

Item	Descrição
Identificação do Teste	DA-10-FTKIMAGER-NTFS-512-CRIPTO
Resumo do Teste	Adquirir um dispositivo físico utilizando interface de acesso PATA para um arquivo de imagem.
Comentário	Duplicação Forense do disco rígido de origem criptografado para disco rígido de destino, usando FTKIMAGER com sistema de arquivos destino NTFS, com 512 setores por bloco e compressão do arquivo de imagem.
Ferramentas necessárias	Mount, FTKIMAGER.
Dispositivos	Origem: 01-PATA Destino: 02-SATA
Preparação do Ambiente	DA-PREPARAÇÃO-01 DA-PREPARAÇÃO-04 sudo mount -o rw –t ntfs-3g /dev/sdb1 /media/sdb1

Tamanho de Bloco : 512.

Comando Utilizado	**time sudo /root/experimentos/ftkimager /dev/sdc /media/sdb1/da-10-ftkimager-ntfs-512-cripto --e01 --frag 2G --compress 9 --case-number 4 --evidence-number 4 --description "da-10-ftkimager-ntfs-512 ftkimager cripto split com compressao" --examiner Daniel --notes "mais notes"**
Resultados	Created By AccessDataÂ® FTKÂ® Imager 2.9 100407 Case Information: Case Number: 4 Evidence Number: 4 Unique description: da-10-ftkimager-ntfs-512 ftkimager cripto split com compressao Examiner: Daniel Notes: mais notes --- Information for /media/sdb1/da-10-ftkimager-ntfs-512-cripto: Physical Evidentiary Item (Source) Information: [Drive Geometry] Cylinders: 2438 Heads: 255 Sectors per Track: 63 Bytes per Sector: 512 Sector Count: 39179952 [Physical Drive Information] Drive Model: ATA SAMSUNG SV0221N Drive Interface Type: SCSI Source data size: 19130 MB

	Sector count: 39179952 [Computed Hashes] MD5 checksum: 4f9178eda6c1b39d504f4bfbaa37efd9 SHA1 checksum: 5dfb7557923fa889c277cd22bfebd1dba1b40cee Image Information: Acquisition started: Sun Jul 22 14:48:58 2012 Acquisition finished: Sun Jul 22 15:04:06 2012 Segment list: /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E01 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E02 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E03 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E04 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E05 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E06 /media/sdb1/da-10-ftkimager-ntfs-512-cripto E07 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E08 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E09 /media/sdb1/da-10-ftkimager-ntfs-512-cripto.E10 real 14m59.158s user 19m16.488s sys 1m42.554s	

E. CONSOLIDAÇÃO DAS MEDIÇÕES DE TEMPO DOS EXPERIMENTOS

A tabela E-1 apresenta a consolidação das medições de tempo, resultantes dos testes listados nas tabelas do anexo D. A Tabela E-2 mostra estatísticas dos testes DA-06 do formato RAW, e a Tabela E-3 as estatísticas dos testes DA-06 para formato E01.

Tabela E-1: Medições de Tempo dos Casos de teste

Identificação do Teste	Tempo Total	Tempo de Usuário	Tempo de Sistema	Velocidade Média de Aquisição [5]	Tempo de Aquisição [6]
DA-06-DC3DD-FAT32-64	14m44.176s	6m7.555s	1m30.870s	22 M/s	887.829s
DA-06-DC3DD-FAT32-128	14m43.249s	5m52.770s	1m5.660s	22 M/s	883.277s
DA-06-DC3DD-FAT32-256	14m43.213	5m49.386	0m54.863	22 M/s	883.198s
DA-06-DC3DD-FAT32-512	14m43.254s	5m49.594s	0m50.347s	22 M/s	883.38 s
DA-06-DC3DD-EXT3-64	14m48.367s	6m4.727s	1m43.446s	22 M/s	886.131 s
DA-06-DC3DD-EXT3-128	14m44.258s	5m38.985s	1m15.957s	22 M/s	883.383 s
DA-06-DC3DD-EXT3-256	14m43.564s	5m48.478s	1m3.396s	22 M/s	883.18 s
DA-06-DC3DD-EXT3-512	14m43.263s	5m43.121s	0m58.260s	22 M/s	883.416 s
DA-06-DC3DD-NTFS-64	15m0.380s	5m51.786s	1m29.430s	21 M/s	892.397 s
DA-06-DC3DD-NTFS-128	14m53.461s	5m57.490s	1m4.956s	21 M/s	922.141s
DA-06-DC3DD-NTFS-256	14m48.993s	5m54.970s	0m51.907s	22 M/s	883.886s
DA-06-DC3DD-NTFS-512	14m43.219s	5m50.442s	0m43.347s	22 M/s	883.271s
DA-06-DCFLDD-FAT32-64	14m43.973s	7m46.889s	0m47.039s	-	-
DA-06-DCFLDD-FAT32-128	14m44.476s	7m48.269s	0m46.635s	-	-
DA-06-DCFLDD-FAT32-256	14m44.305s	7m44.929s	0m46.555s	-	-
DA-06-DCFLDD-FAT32-512	14m44.122s	7m40.789s	0m46.731s	-	-
DA-06-DCFLDD-EXT3-64	14m50.445	7m40.141	0m56.936	-	-
DA-06-DCFLDD-EXT3-128	14m50.958	7m40.257	0m57.088	-	-
DA-06-DCFLDD-EXT3-256	14m50.473	7m35.732	0m56.420	-	-
DA-06-DCFLDD-EXT3-512	14m50.059	7m44.789	0m58.180	-	-
DA-06-DCFLDD-NTFS-64	14m58.092	8m14.159	0m52.615	-	-
DA-06-DCFLDD-NTFS-128	14m59.678	8m14.635	0m52.547	-	-
DA-06-DCFLDD-NTFS-256	14m59.250	8m12.143	0m53.139	-	-
DA-06-DCFLDD-NTFS-512	15m1.893	8m14.435	0m53.403	-	-
DA-06-LINEN-FAT32-64	20m38.704	3m28.341	1m31.230	-	00:20:38

[5] Apenas quando constar nos registros de execução da aplicação.
[6] Apenas quando constar nos registros de execução da aplicação.

Identificação do Teste	Tempo Total	Tempo de Usuário	Tempo de Sistema	Velocidade Média de Aquisição [5]	Tempo de Aquisição [6]
DA-06-LINEN-FAT32-128	20m21.383	3m22.037	1m25.961	-	00:20:21
DA-06-LINEN-FAT32-256	19m37.951	3m1.339	1m27.833	-	00:19:38
DA-06-LINEN-FAT32-512	18m33.995	3m5.040	1m27.477	-	00:18:34
DA-06-LINEN-EXT3-64	21m42.437	3m35.525	1m29.610	-	00:21:42
DA-06-LINEN-EXT3-128	22m0.770	3m30.957	1m22.869	-	00:22:01
DA-06-LINEN-EXT3-256	21m46.045	3m8.108	1m27.861	-	00:21:46
DA-06-LINEN-EXT3-512	21m49.603	3m15.344	1m26.733	-	00:21:49
DA-06-LINEN-NTFS-64	37m55.866	3m36.022	2m7.228	-	00:37:56
DA-06-LINEN-NTFS-128	37m11.767	3m30.125	1m57.387	-	00:37:11
DA-06-LINEN-NTFS-256	36m37.453	3m15.924	2m2.188	-	00:36:37
DA-06-LINEN-NTFS-512	36m13.545	3m14.044	1m59.831	-	00:36:14
DA-06-EWFACQUIRE-FAT32-64	15m5.090	2m5.024	0m47.223	21 MiB/s	00:15:05
DA-06-EWFACQUIRE-FAT32-128	15m7.474	1m59.767	0m45.351	21 MiB/s	00:15:08
DA-06-EWFACQUIRE-FAT32-256	15m5.987	1m59.479	0m46.763	21 MiB/s	00:15:06
DA-06-EWFACQUIRE-FAT32-512	15m4.259	2m1.996	0m48.355	21 MiB/s	00:15:04
DA-06-EWFACQUIRE-EXT3-64	15m8.337	2m6.972	0m59.564	21 MiB/s	00:15:08
DA-06-EWFACQUIRE-EXT3-128	19m15.848	10m48.293	1m43.226	21 MiB/s	00:15:10
DA-06-EWFACQUIRE-EXT3-256	15m14.505	2m5.580	0m58.040	20 MiB/s	00:15:14
DA-06-EWFACQUIRE-EXT3-512	15m13.119	2m3.948	0m56.544	20 MiB/s	00:15:12
DA-06-EWFACQUIRE-NTFS-64	15m18.803	2m34.906	0m58.224	20 MiB/s	00:15:19
DA-06-EWFACQUIRE-NTFS-128	15m22.537	2m31.973	0m53.243	20 MiB/s	00:15:22
DA-06-EWFACQUIRE-NTFS-256	15m14.483	2m28.441	0m53.947	20 MiB/s	00:15:14
DA-06-EWFACQUIRE-NTFS-512	15m9.703	2m28.893	0m52.883	21 MiB/s	00:15:10
DA-06-FTKIMAGER-FAT32-512	19m6.939	10m57.129	1m34.638	-	-
DA-06-FTKIMAGER-EXT3-512	15m3.345s	5m53.706s	1m26.625s	-	-
DA-06-FTKIMAGER-NTFS-512	15m3.303s	6m4.739s	1m26.205s	-	-
DA-10-LINEN-NTFS-512	36m13.545s	3m14.044s	1m59.831s	-	00:36:14
DA-10-EWFACQUIRE-NTFS-512	14m57.292s	13m3.745s	0m22.817s		
DA-10-FTKIMAGER-NTFS-512	14m58.541s	18m27.613s	1m42.262s	-	-
DA-06-DC3DD-NTFS-512-CRIPTO	14m43.171s	5m52.606s	0m41.727s		
DA-06-DCFLDD-NTFS-512-CRIPTO	15m7.720s	8m14.703s	0m51.955s		
DA-06-LINEN-NTFS-512-CRIPTO	36m11.694	3m14.864	1m58.143	-	00:36:12
DA-10-LINEN-NTFS-512-CRIPTO	36m56.864	27m50.008	1m55.023	-	00:36:57
DA-06-FTKIMAGER-NTFS-512-CRIPTO	15m3.620s	6m5.103s	1m25.173s	-	
DA-10-FTKIMAGER-NTFS-512-CRIPTO	14m59.158s	19m16.488s	1m42.554s	-	
DA-06-EWFACQUIRE-NTFS-512-CRIPTO	15m24.272	2m29.509	0m51.251	20 MiB/s	00:15:23
DA-10-EWFACQUIRE-NTFS-512-CRIPTO	21m22.619	16m27.530	0m57.468	14 MiB/s	00:21:22

Tabela E-2:Estatísticas dos testes DA-06 para Formato RAW

Rótulos de Linha	Média de Tempo Total	%Dif Ext3	%Dif NTFS	%Dif Fat32	%Dif 64	%Dif 128	%Dif 256	%Dif 512	%Dif DC3DD	%Dif DCFLDD
DC3DD	00:14:46,4									-0,55%
EXT3	00:14:44,8		-0,73%	0,17%						-0,62%
64	00:14:48,0		-1,33%	0,45%		0,45%	0,45%	0,57%		-0,22%
128	00:14:44,0		-1,01%	0,11%	-0,45%		0,00%	0,11%		-0,79%
256	00:14:44,0		-0,56%	0,11%	-0,45%	0,00%		0,11%		-0,67%
512	00:14:43,0		0,00%	0,00%	-0,56%	-0,11%	-0,11%			-0,79%
FAT32	00:14:43,3	-0,17%	-0,90%							-0,08%
64	00:14:44,0	-0,45%	-1,78%			0,11%	0,11%	0,11%		0,00%
128	00:14:43,0	-0,11%	-1,12%		-0,11%		0,00%	0,00%		-0,11%
256	00:14:43,0	-0,11%	-0,67%		-0,11%	0,00%		0,00%		-0,11%
512	00:14:43,0	0,00%	0,00%		-0,11%	0,00%	0,00%			-0,11%
NTFS	00:14:51,3	0,73%		0,91%						-0,94%
64	00:15:00,0	1,35%		1,81%		0,78%	1,24%	1,93%		0,22%
128	00:14:53,0	1,02%		1,13%	-0,78%		0,45%	1,13%		-0,78%
256	00:14:49,0	0,57%		0,68%	-1,22%	-0,45%		0,68%		-1,11%
512	00:14:43,0	0,00%		0,00%	-1,89%	-1,12%	-0,67%			-2,11%
DCFLDD	00:14:51,3								0,55%	
EXT3	00:14:50,3		-1,06%	0,71%					0,62%	
64	00:14:50,0		-0,89%	0,68%		-0,11%	0,00%	0,00%	0,23%	
128	00:14:51,0		-1,00%	0,79%	0,11%		0,11%	0,11%	0,79%	
256	00:14:50,0		-1,00%	0,68%	0,00%	-0,11%		0,00%	0,68%	
512	00:14:50,0		-1,33%	0,68%	0,00%	-0,11%	0,00%		0,79%	
FAT32	00:14:44,0	-0,70%	-1,75%						0,08%	
64	00:14:44,0	-0,67%	-1,56%			0,00%	0,00%	0,00%	0,00%	
128	00:14:44,0	-0,79%	-1,78%		0,00%		0,00%	0,00%	0,11%	
256	00:14:44,0	-0,67%	-1,67%		0,00%	0,00%		0,00%	0,11%	
512	00:14:44,0	-0,67%	-2,00%		0,00%	0,00%	0,00%		0,11%	
NTFS	00:14:59,7	1,07%		1,78%					0,95%	
64	00:14:58,0	0,90%		1,58%		-0,22%	-0,11%	-0,44%	-0,22%	
128	00:15:00,0	1,01%		1,81%	0,22%		0,11%	-0,22%	0,78%	
256	00:14:59,0	1,01%		1,70%	0,11%	-0,11%		-0,33%	1,12%	
512	00:15:02,0	1,35%		2,04%	0,45%	0,22%	0,33%		2,15%	

Tabela E-3: Estatísticas dos testes DA-06 para Formato E01

Rótulos de Linha	Média de Tempo Total	%Dif Ext3	%Dif NTFS	%Dif Fat32	%Di f 64	%Dif 128	%Dif 256	%Dif 512	%Dif EWF	%Dif FTK	%Dif Linen
DA-06	0:20:22										
EWFACQUIR	0:15:32									-	-
EXT3	0:16:13		6,16%	7,45%						7,75%	-
64	0:15:08		-1,20%	0,33%		-	-	-			-
128	0:19:16		25,24	27,45	27,3		26,34	26,62			-
256	0:15:15		0,11%	0,99%	0,77	-		0,22			-
512	0:15:13		0,33%	1,00%	0,55	-	-			1,11%	-
FAT32	0:15:06	-	-1,20%							-	-
64	0:15:05	-	-1,52%			-	-	0,11			-
128	0:15:07	-	-1,73%		0,22		0,11	0,33			-
256	0:15:06	-	-0,88%		0,11	-		0,22			-
512	0:15:04	-	-0,66%		-	-	-			-	-
NTFS	0:15:16	-		1,21%						1,50%	-
64	0:15:19	1,21%		1,55%		-	0,55	0,99			-
128	0:15:23	-		1,76%	0,44		0,98	1,43			-
256	0:15:14	-		0,88%	-	-		0,44			-
512	0:15:10	-		0,66%	-	-	-			0,78%	-
FTKIMAGER	0:16:24								5,65%		-
EXT3	0:15:03		0,00%	-					-		-
512	0:15:03		0,00%	-					-		-
FAT32	0:19:07	27,02	27,02						26,67		-
512	0:19:07	27,02	27,02						26,88		2,96
NTFS	0:15:03	0,00%		-					-		-
512	0:15:03	0,00%		-					-		-
LINEN	0:26:12								68,78	59,75	
EXT3	0:21:50		-	10,25					34,61	45,04	
64	0:21:42		-	5,08%		-	-	-	43,39		
128	0:22:01		-	8,19%	1,46		1,15	0,84	14,27		
256	0:21:46		-	10,87	0,31	-		-	42,73		
512	0:21:50		-	17,59	0,61	-	0,31		43,48	45,07	
FAT32	0:19:48	-	-						31,20	3,57%	
64	0:20:39	-	-			1,47	5,18	11,22	36,91		
128	0:20:21	-	-		-		3,65	9,61	34,62		
256	0:19:38	-	-		-	-		5,75	30,02		
512	0:18:34	-	-		-	-	-		23,23	-	
NTFS	0:37:00	69,48		86,85					142,2	145,8	
64	0:37:56	74,81		83,70		1,97	3,60	4,69	147,6		
128	0:37:12	68,96		82,80	-		1,59	2,67	141,8		
256	0:36:37	68,22		86,50	-	-		1,06	140,3		