

COMPUTAÇÃO FORENSE: UMA APLICAÇÃO DE SOFTWARES LIVRES PARA RECUPERAÇÃO DE DADOS DIGITAIS

Bruno De Souza Eduardo
bruno_souza_eduardo@hotmail.com
UGB

Fabício Augusto Beijo Carvalho
fabicio.carvalho_@hotmail.com
UGB

Resumo:Resumo: No Brasil a Polícia Federal tem feito várias apreensões de documentos, principalmente computadores, na busca de provas que ajudariam a dar substâncias em suas investigações. Para isso, fazem um trabalho de extração de dados dos equipamentos eletrônicos. Este artigo estuda a aplicação de softwares livres capazes de recuperar arquivos que já foram deletados de um Hard Disk ou Flash Driver. Detalha o conceito da Computação Forense, juntamente com suas aplicações e soluções no mundo cibernético, e em quais cenários podem ser aplicados. A aplicação destes softwares livres tem grande importância para manter os dados em segurança, fazendo com que esses dados tenham backup, por exemplo, e também, mostrar como é possível recuperá-los após deletados.

Palavras Chave: computação - dados - forense - recuperação - software

INTRODUÇÃO

A popularização da Internet, que ocorreu nos anos 90, devido à criação do serviço de World Wide Web (WWW), por (LEE, 1989), permitiu que usuários espalhados pelo mundo pudessem trocar dados e informações em poucos milissegundos, permitindo maior velocidade e rapidez na comunicação entre máquinas e, conseqüentemente, entre as pessoas (ELEUTÉRIO e MACHADO, 2011).

Atualmente, apenas alguns minutos transcorrem entre conectar-se à Internet e ser atacado por outra máquina – e isso é apenas o ruído de fundo dos ataques sem um alvo específico. Houve uma época em que um computador poderia funcionar ano após ano sem sofrer ataques (FARMER e VENEMA, 2005).

Cada vez mais pessoas estão tendo acesso a qualquer meio digital, seja um notebook, celular, tablet ou qualquer outro dispositivo. Toda essa evolução traz benefícios para os usuários, que realizam atividades comuns do dia a dia como o acesso as redes sociais, quanto para as empresas, que aproveitam a praticidade das novas tecnologias para automatizar seus processos os deixando mais simples e eficiente. Mas com todo esse crescimento, paralelamente, surgem os crimes digitais, que é onde a perícia forense entra solucionar esses crimes.

Segundo (QUEIROZ e VARGAS, 2010), a forense computacional é um conjunto de procedimentos e metodologias com a função de investigar e armazenar evidências que possam responder se houve ou não um crime, tendo como base de análise equipamentos de processamento de dados (computadores pessoais, *laptops*, servidores, estações de trabalho ou outras mídias eletrônicas).

(SILVA e OLIVEIRA, 2014) apresentam um estudo sobre as ferramentas computacionais baseadas em software livre e as principais técnicas disponíveis para uma perícia forense computacional. Para isso foram utilizados as ferramentas Forense Digital ToolKit (FDTK-UbuntuBr) e Computer Aided Investigative Environment (CAINE), duas distribuições Linux que possuem um vasto conjunto de ferramentas que atendem aos diversos processos de investigação. Dentre algumas ferramentas apresentadas, foram utilizadas ferramentas para a recuperação de dados de ambas as plataformas (FDTK-UbuntuBr e CAINE), realizando ao final um comparativo entre as ferramentas.

Neste artigo, o objetivo é realizar a recuperação de arquivos já deletados da memória de dispositivos de armazenamento mais comuns, por meio dos principais softwares livres baseados em LINUX, assim sendo possível fazer com que provas apagadas de algum dispositivo, por exemplo, sejam novamente coletadas para a montagem de um dossiê de uma investigação criminal.

DISPOSITIVOS DE ARMAZENAMENTO

Com o avanço tecnológico hoje existem vários dispositivos de armazenamento além do computador, tais como: disquetes, discos ópticos (CD-ROM e DVD-ROOM), external Hard Disk, Pendrive, cartão de memória, dentre muitos outros. E todos esses dispositivos podem armazenar dados que possam ser evidências de um crime de informática.

A maioria dos aplicativos de computador necessita armazenar e recuperar informações, e boa parte dessas informações necessitam ser guardadas por um bom período de tempo. Essas informações podem ser armazenadas em mídias internas ou externas e organizadas em unidades chamados arquivos. O gerenciamento desses arquivos é realizado por uma parte do sistema operacional, normalmente conhecido como sistema de arquivos, o qual deverá prover mecanismos de acesso às informações tais como: criação, alteração, proteção, entre outros (TANENBAUM e WOODHULL, 2000).

No ano de 2000, foi iniciada a venda do pen drive pela empresa Singapurense TREK TECHNOLOGIES, pertencente a IBM, a qual iniciou a comercialização das unidades DiskOnKey da M-Systems.

O Pen Drive, que também pode ser chamado de USB Flash Drive, nada mais é, do que um dispositivo de armazenamento móvel que tem memória flash, ao qual realiza a transferência de informações/dados a computadores, desde que o computador tenha compatibilidade com a entrada do dispositivo. O pen drive é composto pelos seguintes componentes periféricos:

- Placa de circuito impresso;
- Conector USB macho;
- Controlador USB Mass Storage;
- NAND flash, a qual armazena a informação;
- Oscilador de cristal;
- Interruptor de modo de escrita;
- Jumpers e pinos de testes: servem pra testes durante a sua produção;
- LED;
- Capa de proteção.

SISTEMAS DE ARQUIVOS

Um sistema de arquivos consiste em um conjunto de estruturas lógicas e de rotinas que permitem ao sistema operacional controlar o acesso ao disco rígido. Diferentes sistemas operacionais usam diferentes sistemas de arquivos (MORIMOTO, 2002).

Para (FILHO, 2012), os sistemas de arquivos relacionam à forma como os dados são armazenados, organizados e acessados em um local de armazenamento digital. É um artifício imposto pelo sistema operacional e não pelo hardware.

O MS Windows suporta quatro tipos de sistemas de arquivos: CDFS, UDF, FAT e NTFS. Cada sistema determina como os arquivos e diretórios são organizados, o formato dos nomes dos arquivos, desempenho e segurança de acesso aos dados. O CDFS (CD-ROM File System) oferece suporte a dispositivos como CD-ROM e DVD's. O UDF (Universal Disk Format) é uma evolução do CDFS, e também é voltado para CD's e DVD's (MACHADO e MAIA, 2013).

O sistema FAT foi desenvolvido para o sistema MS-DOS e, posteriormente, utilizado nas várias versões do MS Windows. O FAT utiliza esquema de listas encadeadas para estruturar o sistema de arquivos, está limitado a partições de no máximo 2GB, e apresenta baixo desempenho e segurança. O FAT 32 possui a maioria das limitações do sistema de arquivo FAT, porém permite partições de até 2TB (MACHADO e MAIA, 2013).

Segundo (MACHADO e MAIA, 2013), o NTFS (NT File System) foi desenvolvido especialmente para as novas versões do MS Windows, e utiliza o esquema de árvore-B para estruturar o sistema de arquivos, oferecendo alto grau de segurança e desempenho, além de inúmeras vantagens comparado aos sistemas FAT, como:

- Nomes de arquivos com até 255 caracteres, incluindo brancos e letras maiúsculas e minúsculas;
- Partições NTFS dispensam o uso de ferramentas de recuperação de erros;
- Proteção de arquivos e diretórios por grupos;
- Criptografia e compressão de arquivos;
- Suporte a volumes de até 2^{64} bytes;
- Ferramentas de desfragmentação e gerência de quotas em disco;
- Suporte a Unicode;
- Suporte a RAID 0, RAID 1 e RAID 5.

RECUPERAÇÃO DE ARQUIVOS

O crescimento da indústria de recuperação de dados e evidência digital vem acompanhado do forte crescimento do número de ferramentas para forense computacional (MOHAY, ANDERSON, *et al.*, 2003).

A Computação Forense é a ciência que, através de técnicas e habilidades especializadas, trata da coleta, preservação e análise de dados eletrônicos em um incidente computacional ou que envolvam a computação como meio de praticá-lo (ELEUTÉRIO e MACHADO, 2011).

Muitas situações requerem a recuperação de dados no sistema de arquivos, seja por uma perda acidental ou intencional. Sendo assim foi desenvolvida uma variedade de aplicativos e ferramentas qualificados para essa recuperação de arquivos conhecidas como ferramentas forenses, que podem recuperar dados diretamente de um dispositivo físico ou lógico, como um disco rígido ou uma partição desse disco bem como recuperar dados a partir de uma imagem.

O FDTK (Forense Digital Toolkit) é uma ferramenta voltada para a prática da Forense Computacional. Foi a primeira ferramenta open source voltada para a área de computação forense desenvolvida por brasileiros e totalmente em português. Atualmente, a FDTK está na versão 3.0 e conta com a comunidade Linux para implementação de melhorias. Pode ser usado também como Live CD ou a partir de um pen drive rodando sobre qualquer Sistema Operacional (CAVALCANTI, 2016).

MATERIAIS E MÉTODOS

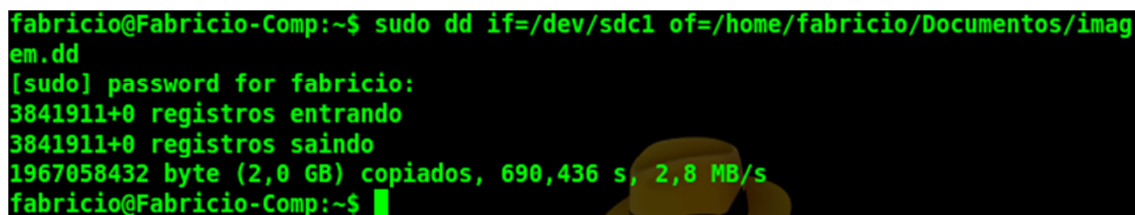
O estudo consiste em aplicar a ferramenta FDTK-Ubuntu, que é um software livre baseado em LINUX, para fazer a recuperação de arquivos de um dispositivo de armazenamento, depois de ter sido formatado e/ou ter tido seus arquivos totalmente deletados. Essa ferramenta facilita o melhor entendimento, ou então, permite um melhor aproveitamento de todos os recursos devido a estar totalmente em Português.

Para realização de todo o processo de recuperação de arquivo é necessário baixar o sistema operacional FDTK. Uma das vantagens desse SO é que se tem a possibilidade de instalação ser realizado em um dispositivo de armazenamento comum. Ou então, o SO pode ser instalado em uma máquina virtual, sendo assim, exigindo alguns requisitos a mais da máquina que está sendo instalado o SO, como por exemplo, um mínimo de memória temporária disponível para que a VM venha a trabalhar sem problemas. Para a realização de testes, deve-se, também, possuir um pen drive, de preferência, particionado para demandar menos tempo na realização dos testes, e então, aplicar os comandos necessários para realizar a recuperação. Não há restrição do tamanho do dispositivo de armazenamento em que se queira trabalhar, porém, quanto maior a capacidade de armazenamento, maior será a demora para realização da ação, devido a varredura em todos os bits do dispositivo.

RESULTADOS

1º passo: Para obter a recuperação dos arquivos, criamos primeiramente uma imagem do dispositivo, ao qual os arquivos foram apagados. Se deve trabalhar na cópia do arquivo, e não no próprio dispositivo. Isso evita eventual problema e perda de tudo que está no pen drive, nesse caso. Segue, como mostrado na Figura 1, o comando a ser aplicado para criar a imagem, e então, o resultado da criação:

Figura 1: Criação de Imagem Forense



```
fabricio@Fabricio-Comp:~$ sudo dd if=/dev/sdc1 of=/home/fabricio/Documents/imagem.dd
[sudo] password for fabricio:
3841911+0 registros entrando
3841911+0 registros saindo
1967058432 byte (2,0 GB) copiados, 690,436 s, 2,8 MB/s
fabricio@Fabricio-Comp:~$
```

Então, temos o comando:

```
sudo dd if=/dev/sdc1 of=/home/fabricio/Documents/imagem.dd
```

Onde:

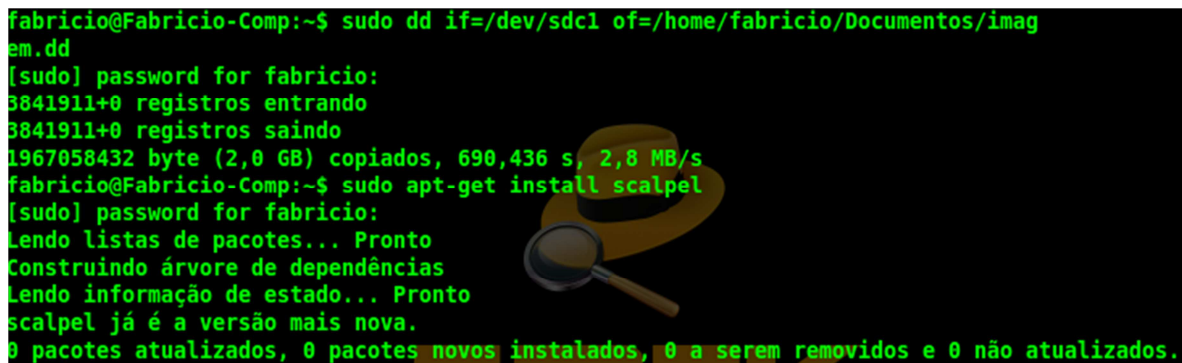
Sudo dd = comando LINUX

if = caminho origem do pen drive.

of = caminho destino para salvar a imagem

2º passo: Deve-se realizar a instalação do SCALPEL, como mostrado na Figura 2, logo abaixo:

Figura 2: Instalando o Scalpel



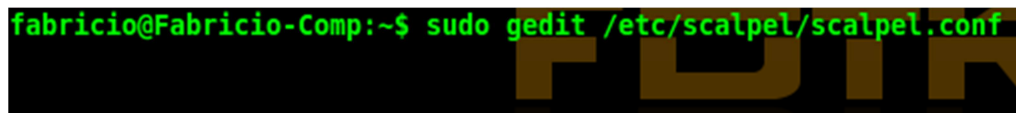
```
fabricio@Fabricio-Comp:~$ sudo dd if=/dev/sdc1 of=/home/fabricio/Documents/imagem.dd
[sudo] password for fabricio:
3841911+0 registros entrando
3841911+0 registros saindo
1967058432 byte (2,0 GB) copiados, 690,436 s, 2,8 MB/s
fabricio@Fabricio-Comp:~$ sudo apt-get install scalpel
[sudo] password for fabricio:
Lendo listas de pacotes... Pronto
Construindo árvore de dependências
Lendo informação de estado... Pronto
scalpel já é a versão mais nova.
0 pacotes atualizados, 0 pacotes novos instalados, 0 a serem removidos e 0 não atualizados.
```

No FDTK, por padrão, o SCALPEL já virá instalado.

3º passo: Deve se abrir o SCALPEL em um editor de texto, para então, ser definido qual a extensão do arquivo se deseja recuperar. Exemplo: .png, .jpeg, .pdf, .docx, entre outros.

Segue, como mostrado na Figura 3:

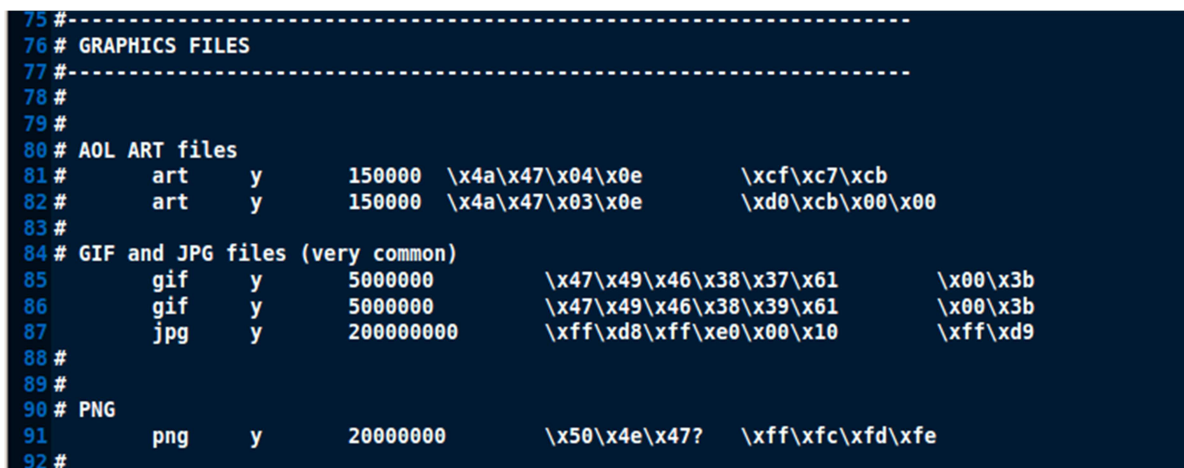
Figura 3: Abrindo o editor de texto



```
fabricao@Fabricio-Comp:~$ sudo gedit /etc/scalpel/scalpel.conf
```

Conforme na Figura 4, no editor de texto aberto para edição, deve-se retirar o “#” da frente da extensão em que se queira recuperar.

Figura 4: Exibição do editor de texto



```
75 #-----
76 # GRAPHICS FILES
77 #-----
78 #
79 #
80 # AOL ART files
81 #     art      y      150000  \x4a\x47\x04\x0e      \xcf\x7\xcb
82 #     art      y      150000  \x4a\x47\x03\x0e      \xd0\xcb\x00\x00
83 #
84 # GIF and JPG files (very common)
85 #     gif      y      5000000  \x47\x49\x46\x38\x37\x61      \x00\x3b
86 #     gif      y      5000000  \x47\x49\x46\x38\x39\x61      \x00\x3b
87 #     jpg      y      200000000  \xff\xd8\xff\xe0\x00\x10      \xff\xd9
88 #
89 #
90 # PNG
91 #     png      y      20000000  \x50\x4e\x47?  \xff\xfc\xfd\xfe
92 #
```


Por último, como mostrado na Figura 5, aplica-se o comando para a recuperação dos arquivos, ao qual foi selecionado a extensão no editor de texto do SCALPEL.

Figura 5: Recuperação dos arquivos

```
fabricio@Fabricio-Comp:~$ sudo scalpel /home/fabricio/Documents/imagem.dd -o /home/fabricio/Documents/Recuperado
Scalpel version 1.60
Written by Golden G. Richard III, based on Foremost 0.69.

Opening target "/home/fabricio/Documents/imagem.dd"

Image file pass 1/2.
/home/fabricio/Documents/imagem.dd: 100.0% |*****| 1.8 GB 00:00 ETA
Allocating work queues...
Work queues allocation complete. Building carve lists...
Carve lists built. Workload:
gif with header "\x47\x49\x46\x38\x37\x61" and footer "\x00\x3b" --> 0 files
gif with header "\x47\x49\x46\x38\x39\x61" and footer "\x00\x3b" --> 0 files
jpg with header "\xff\xd8\xff\xe0\x00\x10" and footer "\xff\xd9" --> 3915 files
png with header "\x50\x4e\x47\x3f" and footer "\xff\xfc\xfd\xfe" --> 24 files
doc with header "\xd0\xcf\x11\xe0\xal\xbl\xla\xel\x00\x00" and footer "\xd0\xcf\x11\xe0\xal\xbl\xla\xel\x00\x00" --> 11 files
doc with header "\xd0\xcf\x11\xe0\xal\xbl" and footer "" --> 11 files
pdf with header "\x25\x50\x44\x46" and footer "\x25\x45\x4f\x46\x0d" --> 0 files
pdf with header "\x25\x50\x44\x46" and footer "\x25\x45\x4f\x46\x0a" --> 1 files
Carving files from image.
Image file pass 2/2.
/home/fabricio/Documents/imagem.dd: 100.0% |*****| 1.8 GB 00:00 ETA
Processing of image file complete. Cleaning up...
Done.
Scalpel is done, files carved = 3962, elapsed = 103 seconds.
fabricio@Fabricio-Comp:~$
```

Para o comando aplicado, temos o arquivo de imagem em que se deve fazer a recuperação, e para a origem da recuperação, foi criada uma pasta “Recuperado”, para todos os arquivos que forem recuperados, fiquem nessa pasta.

Abaixo, na Tabela 1, será mostrado um comparativo utilizado para os testes de recuperação. Temos o resultado de que, nesse teste, independente de quantas vezes os testes foram executados, a recuperação dos arquivos perdidos foram quase os mesmos:

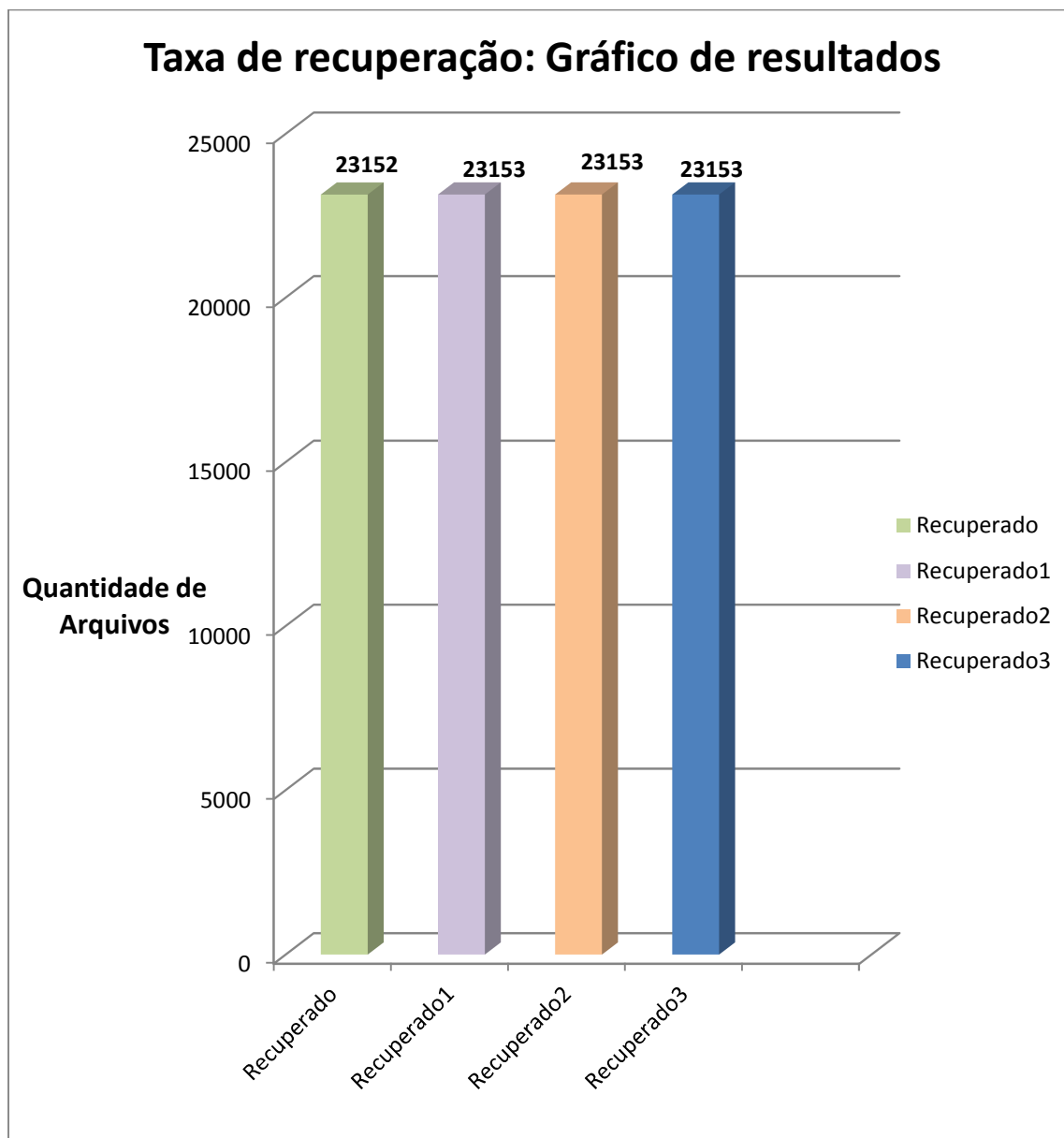
Tabela 1 – Comparativo dos testes para recuperação dos arquivos

Taxa de recuperação: Tabela de resultados		
#	Total Arquivo	Total MB
Recuperado	23152	686,5
Recuperado1	23153	687,8
Recuperado2	23153	687,8
Recuperado3	23153	687,8

Fonte: Elaboração Própria

No Gráfico 1, é possível observar o resultado das recuperações dos arquivos através de cada teste realizado.

Gráfico 1 – Resultado dos arquivos recuperados em cada teste



Fonte: Elaboração Própria.

CONCLUSÃO

Neste artigo foi abordado um pouco do conceito sobre a computação forense, sua metodologia e como ela vem a ser útil em diversas ocasiões, como na recuperação de arquivos de dispositivos de armazenamento por meio de softwares livres.

Pela observação dos aspectos analisados, conclui-se que é possível fazer a recuperação de arquivos de qualquer dispositivo de armazenamento que tenha sido formatado, ou então, que seus arquivos tivessem sido apagados.

Logo é sempre importante ressaltar em manter os arquivos em mais de um dispositivo seja pendrive, HD entre outros, para que não ocorra o risco de uma eventual perda acidental ou intencional.

Concluimos que as ferramentas forenses podem sim ser uma solução muito viável para a recuperação de arquivos, e é recomendável para esse tipo de situação, onde normalmente não se possui muitas alternativas confiáveis.

REFERÊNCIAS

CAVALCANTI, B. B. Crimes digitais: A fragilidade da legislação brasileira no Direito Digital e demonstração de perícia forense, Rio De Janeiro, 2016.

ELEUTÉRIO, P. M. D. S.; MACHADO, M. P. **Desvendando a Computação Forense**. São Paulo: Novatec , 2011.

FARMER, D.; VENEMA, W. **Perícia Forense Computacional - Teoria e prática aplicada**. São Paulo: Pearson Prentice Hall, 2005.

FILHO, J. E. M. **Descobrimdo o Linux: Entenda o Sistema Operacional GNU/Linux**. 3. ed. São Paulo: Novatec, 2012.

MACHADO, F. B.; MAIA, L. P. **Arquitetura de Sistemas Operacionais**. 5. ed. Rio De Janeiro: LTC, 2013.

MOHAY, G. et al. **Computer and Intrusion Forensics**. [S.l.]: Artech House, 2003.

MORIMOTO, C. E. **Hardware Manual Completo**. 3. ed. [S.l.]: [s.n.], 2002.

QUEIROZ, C.; VARGAS, R. **Investigação e perícia forense computacional: certificações, leis processuais e estudos de caso**. Rio de Janeiro: Brasport, 2010.

SILVA, V. A.; OLIVEIRA, C. H. D. Análise De Ferramentas Livres Para Perícia Forense Computacional. **Caderno de Estudos Tecnológicos - Faculdade de Tecnologia de Ourinhos**, São Paulo, 2014.

TANENBAUM, A. S.; WOODHULL, A. S. **Sistemas Operacionais: Projeto e Implementação**. 2. ed. Porto Alegre: Bookman, 2000.