Sistemas Operacionais *Open Source* Usados na Computação Forense

Renan Beck Oliveira[1], André Ricardo Zavan[1]

[1]Curso Superior de Tecnologia em Análise e Desenvolvimento de Sistemas Instituto Federal do Paraná (IFPR) – Campus de Paranavaí - CEP 87.706-340 – Paranavaí – PR– Brasil

renan.beckoliveira@hotmail.com, andre.zavan@ifpr.edu.br

Abstract. *These paper intends to evaluate open sources tools utilized on Forense Computer, analyzing and testing them on Kali, Santoku, Fdtk, BackBox and Caine operational systems, letting clear their main goal, that is, compare among procedures utilized to investigate possible crimes that could have being done. Besides, the investigative work done by Forense Computer compares to the police work on colect and analysis of the crime facts. Providing an test environment, we used virtualization tools (VmWare Workstation), a software that the goal is to virtualize operational systems over mentioned providing to the mendump; fcrackzip/ peepdf; reglookup; exifprobe tools could be installed, analyzed and tested in different systems and architectures.So, this paper intends to describe the importance of the forense computer , besides analyzing and testing the mentioned tools.*

Resumo. Este artigo propõe-se a avaliar ferramentas *open source* usadas na computação forense, analisando e testando-as nos sistemas operacionais *Kali*, *Santoku*, *Fdtk*, *BackBox* e *Caine*, deixando claro seu objetivo principal, a comparação entre os procedimentos utilizados para a investigação de possíveis crimes, que possam ter sido cometidos. Salienta-se que o trabalho investigativo realizado pela Computação Forense, equipara-se ao da polícia, no que tange à busca, coleta e apuração de fatos criminosos. Para propiciar um ambiente de testes, utilizaremos a ferramenta de virtualização *(VmWare Workstation)*, um *software* cuja finalidade é virtualizar os sistemas operacionais supracitados para possibilitar que as ferramentas *memdump; fcrackzip; peepdf; reglookup; exifprobe* possam ser instaladas, analisadas e testadas em diferentes sistemas e arquiteturas. O presente trabalho, portanto, pretende discorrer sobre a importância da computação forense além de analisar e testar as ferramentas citadas acima.

1. Introdução

Na atualidade, o ambiente virtual tem sido cada vez mais utilizado, com as mais diversas finalidades, entre elas, compras, pagamentos de contas, etc. As pessoas, têm a falsa

sensação de segurança quando instalam *software* antivírus de determinado fabricante, entretanto, os usuários na maioria das vezes, não são *experts* em tecnologia para saberem como usar a internet sem correr riscos, no que tange à segurança das informações (Eleutério, 2011).

Conforme avança a internet, os criminosos inovam para se apropriarem das informações pessoais dos usuários, disseminam *software* maliciosos na rede mundial de computadores, tanto que em 2016 42,4 milhões de brasileiros foram vítimas de crimes virtuais (DINO, 2017). Uma das motivações para este trabalho, sem dúvida, foi o fato de os crimes cibernéticos terem aumentado significativamente, o que também faz crescer a procura por métodos e procedimentos capazes de investigar tais crimes.

Assim, com o intuito de investigar possíveis crimes digitais, bem como proceder à detecção, recuperação e análise dos dados obtidos através das buscas, entra em cena, a computação forense na busca de provas para a resolução de cibercrimes (Eleutério, 2011).

2. Perícia Forense

Conforme (Freitas, 2003) aponta, diferentemente da Perícia Forense (termo policial), a Perícia Forense no âmbito computacional é tida como a coleta, recuperação, análise de dados, com intuito de desvendar o curso das ações, recriando cenários completos fidedignos.

2.2 Legislação

Até o ano de 2012 não havia uma lei que penalizasse crimes praticados na rede mundial de computadores, porém, após o incidente ocorrido com a atriz Carolina Dickman, foi aprovada em novembro de 2012 a lei 12737 que alterou o Código Penal Brasileiro, incluindo os artigos 154-A, 154-B, 266 e 298, passando a tipificar invasão de dispositivos informáticos (Brasil, 2012).

Em abril de 2014, a câmara dos deputados aprovou a Lei nº 12.965, que ficou conhecida como “Marco Civil da Internet”, vindo a estabelecer princípios, garantias, direitos e deveres para um bom uso da internet no país (Brasil, 2014).

3. *Software Open Source*

Tratam-se de s*oftware*, que têm códigos abertos, gratuitos, geralmente mantidos pelos desenvolvedores e compartilhados, de forma que, todos possam ter acesso ao mesmo (YWASAKI, 2008). Para que um *software*, possa ser chamado de *Open Source*, o mesmo deverá cumprir sem exceções, quatro requisitos fundamentais, segundo (Augusto Campos, 2006) são eles:

- **Liberdade de executar o programa para qualquer efeito:** Autoriza o usuário a utilizar o *software* sem limites, ou seja, para qualquer fim.

- **Liberdade de estudar como o programa funciona e adaptá-lo às suas necessidades:** Permite ao usuário modificar o *software* conforme desejar sem obrigações legais, como ocorre com *software* proprietário.
- **Liberdade de redistribuir cópias, de modo que se possa ajudar os outros:** Esta liberdade tem como característica, a redistribuição do *software* ao próximo, de forma a ajuda-lo, sem custo caso desejar.
- **Liberdade de aperfeiçoar o programa e fazer melhorias publicamente disponíveis, para toda comunidade se beneficiar:** Liberdade que permite aos usuários menos afinados com a linguagem de programação, acessar indiretamente à modificação do *software*, podendo ter um custo.

4. Testes e Resultados

Neste capítulo será apresentado a elaboração da pesquisa, os testes realizados, os sistemas operacionais experimentados, bem como as ferramentas *Open Source* utilizadas para coletar e analisar os dados encontrados, além de mostrar os resultados. O hardware utilizado será um *notebook* com um processador core i7 2.0 Gigahertz, 6 Gigabytes de memória RAM, placa de vídeo dedicada de 2 Gigabytes e *Hard Disk* de 1 Terabytes. Já as máquinas virtuais a serem virtualizadas utilizarão um *hardware* virtual de 50 Gigabytes de *Hard Disk*, memória de 1 Gigabyte cada, podendo assim instalar os sistemas operacionais e as ferramentas necessárias para realizar os testes pretendidos.

4.1 Metodologia

A metodologia utilizada neste trabalho foi segmentada em três etapas. A primeira foi a realização de uma pesquisa descritiva, visando investigar ferramentas *open source* usadas na computação forense, através de teorias publicadas em livros, internet e/ou outras obras do mesmo gênero (Eleutério, 2011). Durante a elaboração deste artigo foram estudados diversos artigos científicos que têm objetivo semelhante ao nosso, apresentando sistemas operacionais e ferramentas semelhantes.

Na segunda, realizamos os testes no ambiente, aplicando as ferramentas selecionadas durante a primeira etapa. Por fim, a terceira, consistiu na análise dos dados dos arquivos comuns (zip), (pdf), (reg), (jpg) e (dump), utilizando assim as ferramentas descritas na pesquisa.

Os resultados obtidos serão analisados levando em conta, o êxito obtido pela feramenta em sua busca, bem como o tipo de resultado obtido/esperado. Podendo assim constituir ou não evidências para um possível crime digital.

4.2 Ferramentas Utilizadas

4.2.1 Kali linux versão 2016

Distribuição Linux baseada em Debian voltada para testes de penetração e auditoria de segugrança da informação, forense computacional e engenharia reversa. É uma forma melhorada da antiga distribuição *BackTrack* Linux, reformulada e com suas ferramentas revistas. Utiliza como ferramenta de controle de versões o GIT, possui uma gama de ferramentas forenses pré instaladas (Kali Linux, 2016).

4.2.2 Santoku linux versão 0.4

Distribuição Linux baseada em Debian voltada para a forense computacional, análise de dados, análise de *malware*, engenharia reversa e segurança móvel, não sendo necessário instalar várias delas, pois já vem prontas para o uso e com base de dados atuais (Santoku Linux, 2016).

4.2.3 Fdtk linux versão 3

Distribuição baseada em várias distribuições linux entre elas, Ubuntu, Helix, deft, *BackTrack* entre outras, voltadas à análise e coleta de dados na forense computacional (Santos, 2008).

4.2.4 BackBox linux versão 4.6

Distribuição Linux baseada em Ubuntu voltada para segurança digital e testes de penetração, análise de sistemas e redes de computadores. Vêm com várias ferramentas forenses instaladas com o sistema, seu repositório está em constante atualização, é de fácil manutenção (Analista Ti, 2013).

4.2.5 Caine linux versão 8

Distribuição Linux baseada em Ubuntu voltada para a computação forense, testes de penetração etc. Possui vários s*oftware* forenses nativos, gráficos simples, base de dados atualizados continuamente (Santos, 2008).

4.2.6 Memdump

Ferramenta capaz de realizar a leitura da memória RAM, retirando as informações situadas na memória física.

4.2.7 Fcrackzip

Ferramenta capaz de recuperar palavras-chave ou senhas de arquivos "*zip"*, de fácil instalação e utilização, possui vários métodos que podem ser usados para o *cracking* das senhas, disponibiliza até mesmo a opção de escolher quantas senhas por segundo serão tentadas, possibilitando ao usuário escolher a mais rápida.

4.2.8 Peepdf

Ferramenta utilizada para examinar arquivos ".pdf", à fim de recolher informações importantes sobre o arquivo, como por exemplo, se o mesmo contém dados prejudiciais

ao sistema. Aceita os filtros e tipos de codificação mais comuns, consegue analisar versões diferentes de um mesmo arquivo, fluxos dos objetos, bem como analisar arquivos com senhas.

4.2.9 Reglookup

Ferramenta usada para ler informações de registros *microsoft windows* NT/2000/XP, possui várias opções, como por exemplo, filtragem de resultados por tipos de dados, impressão de resultados em um formato padronizado, etc.

4.2.10 Exifprobe

Ferramenta utilizada para examinar o conteúdo e a estrutura de arquivos ".jpeg" e ".tiff", pode ser usado para examinar imagens corrompidas, reconhece praticamente todos os marcadores jpeg, relatando informações importantes como, tamanho, localização, formato dos dados, etc.

5. Resultados Obtidos

Uma busca simulada foi executada em quatro arquivos diferentes, sendo eles (.pdf), (.jpg), (.zip) e (.reg). Cada sistema operacional utilizados na pesquisa e acima listados foram virtualmente instalados através do Vmware. As ferramentas empregadas na pesquisa foram instaladas em todos os sistemas operacionais supracitados, logo após os testes utilizando cada ferramenta, os testes foram iniciados, como cada ferramenta analisa um arquivo diferente, o tempo e o hardware necessários para concluir cada teste variaram um em relação ao outro.

5.1 Memdump

Na situação abaixo foi aplicada em conjunto com seu parâmetro básico "-v", a fim de obter maiores detalhes de sua busca, na qual deverá ter um tempo máximo de cinco minutos para ser concluída. O teste será realizado em todos os sistemas operacionais descritos acima, demonstrando assim que a mesma funciona em todos os sistemas pesquisados. A figura 1 demonstra o resultado obtido com o teste realizado, trazendo como informações o tamanho do buffer, da página e do próprio *dump* da memória.

Figura 1 -"Teste com a ferramenta Memdump".

Fonte: Do Autor.

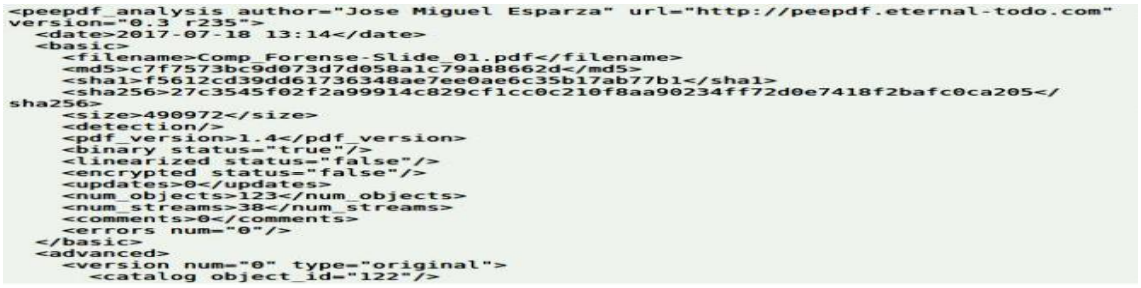
. 5.2 Fcrackzip

Ordenada e disposta em conjunto com seus parâmetros básicos "-x, -v, -l 3-8", a qual tem como objetivo usar o método de força bruta, detalhar a busca e limitar o tamanho do possível *password* a ser buscado. O teste foi realizado em todos os sistemas operacionais descritos mensurados nesta trabalho, demonstrando assim que a ferramenta é compatível

e funciona em todos os sistemas pesquisados. Escolheu-se um arquivo comum com a extensão (.zip), e através da ferramenta e de suas opções acima citadas, tal arquivo foi testado, o *hardware* utilizado foi insuficiente, porém, como o tempo de resposta da ferramenta chegou a 3 semanas, o arquivo alcançou um tamanho de 31 GigaBytes, não seria possível a leitura do mesmo, tornando inviável continuar os testes com a ferramenta devido a inviabilidade de *hardware.*

5.3 Peepdf

Foi aplicada em conjunto com seus parâmetros básicos "-x, info, ", a qual tem como objetivo exportar o resultado em formato .xml, detalhar um elemento específico do arquivo. O teste foi realizado em todos os sistemas operacionais apresentados no trabalho, demonstrando assim que a mesma funciona e possui compatibilidade em todos os sistemas pesquisados. A figura 2 acima demonstra que a ferramenta coletou informações do arquivo como, se o mesmo é criptografado ou não, o número de objetos, *streams*, erros, comentários, a versão do arquivo entre outras informações.



```
<peepdf_analysis author="Jose Miguel Esparza" url="http://peepdf.eternal-todo.com"
version="0.3 r235">
  <date>2017-07-18 13:14</date>
  <basic>
    <filename>Comp_Forense-Slide_01.pdf</filename>
    <md5>c7f7573bc9d073d7d058a1c79a88662d</md5>
    <sha1>f5612cd39dd61736348ae7ee0ae6c35b17ab77b1</sha1>
    <sha256>27c3545f02f2a99914c829cf1cc0c210f8aa90234ff72d0e7418f2bafc0ca205</
sha256>
    <size>490972</size>
    <detection/>
    <pdf_version>1.4</pdf_version>
    <binary status="true"/>
    <linearized status="false"/>
    <encrypted status="false"/>
    <updates>0</updates>
    <num_objects>123</num_objects>
    <num_streams>38</num_streams>
    <comments>0</comments>
    <errors num="0"/>
  </basic>
  <advanced>
    <version num="0" type="original">
      <catalog object_id="122"/>
```


Figura 2 -"Teste com a ferramenta *Peepdf*".

Fonte: Do Autor

5.4 Reglookup

Empregada em conjunto com seu parâmetro básico " -v ", a fim de obter maiores detalhes de sua busca. O teste será realizado em todos os sistemas operacionais descritos acima, demonstrando assim que a mesma funciona em todos os sistemas pesquisados. Escolheu-se um arquivo comum (.reg), e através da ferramenta e de sua opção acima citada, tal arquivo foi testado. Como descrito na figura 3, a ferramenta não conseguiu criar a estrutura do arquivo a ser lido, e, portanto, o resultado não foi o esperado. Após o erro no teste tentamos utilizar outro arquivo de registro, porém sem sucesso também, tentamos editar os arquivos usados nos testes com o editor de texto "vi", demonstrando assim que os arquivos não estavam vazios, ao conseguir fazer a leitura do arquivo constatou-se que o erro era na ferramenta.

Figura 3 -"Teste com a ferramenta Reglookup".

Fonte: Do Autor

5.5 Exifprobe

Empregada em conjunto com seu parâmetro básico " -R ", a qual tem como objetivo retornar o resultado da busca em forma de relatório. O teste será realizado em todos os sistemas operacionais descritos acima, demonstrando assim que a mesma funciona em todos os sistemas pesquisados e estudados nesta pesquisa. Conforme demonstra na figura 4, a ferramenta retorna como resultado informações sobre o arquivo analisado como, nome, tamanho, versão, densidade dos eixos x e y, componentes, bits entre outras informações relevantes para uma perícia forense.

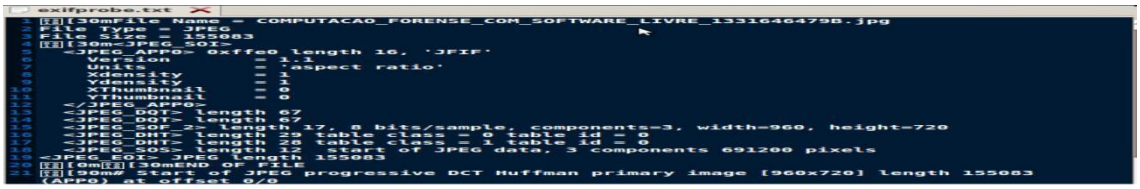


Figura 4 -" Teste com a ferramenta Exifprobe".

Fonte: Do Autor

6. Conclusão

A proposta suscitada por este trabalho foi a avaliação de ferramentas *Open Source* usadas na computação forense, testá-las em sistemas operacionais específicos e analisar os resultados obtidos, com o intuito de comparar os procedimentos utilizados para a investigação de possíveis crimes, que possam ter sido cometidos.

Portanto, a principal contribuição desta pesquisa, sem dúvida foi tornar conhecidas as ferramentas usadas para auxiliar a investigação de crimes digitais, bem como analisá-las, testá-las e verificar os resultados obtidos através dos testes, podendo assim consolidar a existência ou não de um crime digital, não menos importante, informar o leitor sobre *softwares open source* e sua importância no mundo digital.

Para os próximos trabalhos poderão ser abordadas as ferramentas *fcrackzip*, *reglookup* novamente para entender e analisar as limitações das ferramentas encontradas no presente artigo, por que elas ocorrem e como corrigi-las.

7. Referências Bibliográficas

Computação Forense com Software Livre. Disponível em: https://analistati.com/computacao-forense-com-software-livre/. Acessado em: 08/09/2017.

CAMPOS, Augusto. O que é software livre. BR-Linux. Florianópolis, m/arço de 2006. Disponível em: http://br-linux.org/linux/faq-softwarelivre. Acessado em: 07/09/2017.

DINO, Divulgadores de Notícias. Crimes virtuais afetam 42,4 milhões de brasileiros. http://economia.estadao.com.br/noticias/releases-ae,crimes-virtuais-afetam-42-milhoes-de-brasileiros,70001644185. Acessado em: 29/08/2017.

Distribuições em Software Livre para Forense Computacional. Disponível em: http://revista.espiritolivre.org/ivforumrel/wp-content/uploads/2012/10/GilbertoSudre_DistribuicoesbaseadasemSLparaForense_IVForumREL.pdf. Acessado em: 08/09/2017.

ELEUTÉRIO, Pedro Monteiro da Silva; MACHADO, Márcio Pereira. Desvendando a Computação Forense. Novatec Editora Ltda. 1ª edição/2011. Disponível em: http://docs14.minhateca.com.br/164789555,BR,0,0,Desvendando-a-Computa%C3%A7%C3%A3o-Forense.pdf. Acessado em: 10/02/2017.

FREITAS, Andrey Rodrigues de. Perícia Forense Aplicada a Informática. Disponível em:http://www.truzzi.com.br/blog/wp-content/uploads/2010/08/Monografia.pdf. Acessado em: 15/06/2017.

KALI. KALI LINUX. Official Kali Linux Documentation. Disponível em: https://docs.kali.org/pdf/kali-book-pt-br.pdf. Acessado em: 08/09/2017.

BRASIL. Lei nr. 12.965 de 23 marco de 2014. Estabelece principios, garantias, direitos e deveres para o uso da internet no brasil. Disponivel em: http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2014/lei/l12965.htm. Acesso: 05/02/2017.

BRASIL. Lei n. 12.737 de Abril de 2012. Dispõe sobre a tipificação criminal de delitos informáticos. Disponível em: http://www.planalto.gov.br/ccivil_03/_ato2011-2014/2012/lei/l12737.htm. Acessado em: 15/06/2017.

LOPES, Petter Anderson. Perícia Forense Computacional. Disponível em: https://periciacomputacional.com/pericia-forense-computacional/. Acessado em 15 de junho de 2017.

QUEIROZ, Claudemir e VARGAS, Raffael; Investigação e Perícia Forense Computacional.1. ed. Rio de Janeiro:Brazport, 2010.

SANTOS, Rodrigo Franco dos. Ferramentas de Computação Forense Baseadas em *Software* Livre. Faculdade Impacta de Tecnologia (FIT) Orientador: RITCHER, Renato(Msc).Disponível em: http://docplayer.com.br/4051996-Ferramentas-de-computacao-forense-baseadas-em-software-livre.html. Acessado em: 08/01/2017.

SANTOKU. Santoku Linux.Disponível em: https://santoku-linux.com/about-santoku/. Acessado em: 08/09/2017.