Capítulo
6

Live forensics em ambiente Microsoft Windows

Bruno Werneck Pinto Hoelz, Frederico Imbroisi Mesquita e Pedro Auler

Abstract

Conventional digital analysis is being challenged by the presence of encrypted content and the large volume of data to be processed. Live forensics can be applied to secure access to hard disk data and to perform the triage of evidence. However, this kind of analysis is considered complex due to the variety of information to be processed in a short period of time. This work presents the main concepts of live forensics, including its advantages and disadvantages when compared to the conventional methodology. Additionally, it provides a set of procedures grounded on system commands and open source tools for the Windows operating system, installed in more than 90% of desktop computers in Brazil.

Resumo

A presença de conteúdo criptografado ou de grande volume de informações a serem periciados são os novos desafios para a perícia digital convencional. A análise live*, ou* live forensics*, pode ser utilizada para garantir o acesso ao conteúdo do disco rígido e realizar a triagem de evidências. Porém, trata-se de uma perícia complexa devido à grande variedade de informações a serem analisadas em um curto período de tempo. Este trabalho apresenta os principais conceitos da análise* live*, incluindo suas vantagens e desvantagens quando comparada com a perícia digital convencional. Além disso, fornece um conjunto de procedimentos apoiado em comandos e ferramentas de código aberto para o sistema operacional Windows, presente em mais de 90% dos computadores de mesa do Brasil.*

6.1. Introdução

A análise *live*, também chamada de *live forensics*, consiste em uma análise digital realizada por meio de procedimentos periciais e conduzida no equipamento computacional ainda em execução. O procedimento pode ser relativamente simples e rápido, tal como listar as portas de conexões abertas. Porém, pode também ser complexo e demorado,

como, por exemplo, realizar buscas por palavras-chave no disco rígido utilizando expressões regulares e copiar integralmente o conteúdo deste disco por meio da porta USB. O procedimento apresenta vantagens e desvantagens em relação à perícia digital convencional, também chamada de análise *post-mortem*, realizada após o desligamento do sistema.

Coletar dados digitais em um sistema já desligado traz a vantagem de tornar a sobrescrita acidental ou modificação de dados praticamente impossível. Por outro lado, não permite a aquisição de dados voláteis, que são perdidos durante o processo de desligamento do sistema. Além disso, há outras situações em que a recuperação de dados permanentes também é inviabilizada. É o caso, por exemplo, do uso de criptografia, quando só é possível recuperar as informações com o uso da senha de acesso correta. Mais uma vez, esse problema seria contornado caso a aquisição lógica dos dados criptografados tivesse se dado com o sistema ainda ligado. Outro exemplo é a aquisição de informações referentes ao estado da rede e suas portas relacionadas, que também são perdidas ao se desligar o sistema.

Por isso, a coleta de dados com o computador ainda ligado parece ser uma alternativa salvadora. Essa técnica permite a recuperação de valiosas informações que de outra maneira poderiam ser perdidas. Infelizmente, essa abordagem também tem suas limitações. A mais importante é que cada computador analisado possui um sistema operacional diferente instalado. Assim, o analista precisa ter conhecimento de uma grande variedade de *hardware*, *software* e sistemas operacionais. O examinador precisa verificar o sistema em análise e aplicar os princípios forenses corretamente, de maneira a não inviabilizar a futura aceitação das evidências coletadas, quando utilizadas no devido processo legal. Parte do processo de aquisição de dados voláteis consiste em executar aplicativos na CPU do sistema suspeito, podendo levar a potenciais alterações de dados de registros, memória RAM ou do próprio disco rígido. Tais alterações devem ser controladas e documentadas. Dependendo de como se dá a abordagem no local de aquisição dos dados voláteis, a alteração do sistema pode ser tão expressiva que pode inviabilizar o uso futuro das informações coletadas.

A popularização do uso de programas de criptografia, cada vez mais fáceis de utilizar, e, muitas vezes incorporados aos sistemas operacionais, está tornando mais comum o fato de se encontrar sistemas ligados utilizando esta tecnologia. Em geral, não é muito fácil detectar a criptografia em uso no sistema, já que o *software* utilizado pode ser muito discreto, deixando poucos rastros da sua presença. Assim, a abordagem na coleta de dados voláteis em sistemas ligados tem que se ser bastante criteriosa, a fim de detectar a presença de criptografia e, se for o caso, fazer uma cópia lógica do sistema antes de desligá-lo.

A grande capacidade de armazenamento da memória RAM, muitas vezes igual ou superior a quatro gigabytes nos computadores atuais, é capaz de guardar grande quantidade de dados, podendo incluir, entre outras informações, senhas usadas para criptografia. A análise com o volume ainda montado possibilita ainda a aquisição lógica deste volume, que, de outra forma, apareceria como um arquivo criptografado, difícil de ser detectado e praticamente impossível de ser acessado.

Podem ser encontradas várias evidências extremamente úteis na memória RAM como, por exemplo, o conteúdo inteiro ou parcial de arquivos apagados, senhas em texto

claro, *buffers* com conteúdo da área de transferência, informações sobre processos em execução ou já encerrados. Portanto, não é mais possível ignorar a memória volátil dos computadores durante a fase de coleta de dados e a análise subsequente.

Apesar dos recentes progressos da análise de memória, as dificuldades ainda são grandes, devido à falta de flexibilidade das ferramentas existentes, que geralmente só podem ser utilizadas nos sistemas operacionais específicos e nas respectivas versões para as quais foram codificadas. A razão para isso é que as estruturas de dados utilizados pelos sistemas operacionais mudam a cada nova versão, exigindo que as ferramentas forenses também precisem ser atualizadas.

Neste trabalho, serão abordados aspectos fundamentais de perícia digital, incluindo questões de terminologia, um breve histórico da evolução da área e seus principais desafios. Posteriormente, os conceitos da análise *live* – incluindo suas vantagens e desvantagens, quando comparada com a perícia digital convencional – são apresentados. Os procedimentos de coleta e análise de evidências são detalhados, juntamente com um conjunto de ferramentas gratuitas para o ambiente Microsoft Windows, presente em mais de 90% dos computadores de mesa do Brasil, conforme dados da NetMarketShare, apresentados na Figura 6.1.

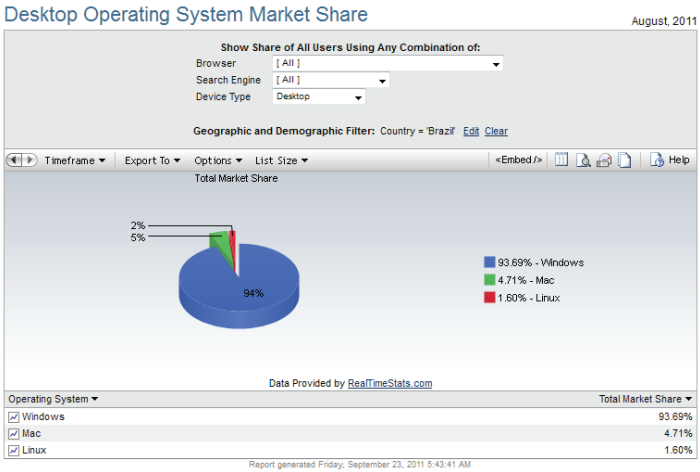

Figura 6.1. Base instalada de sistemas operacionais em computadores de mesa

6.2. Terminologia

Antes de aprofundar a discussão sobre análise *live*, é necessário esclarecer alguns pontos com relação à tradução dos termos em inglês e de outras denominações utilizadas. O termo em inglês associado à perícia em computadores é *Computer Forensics*. Em inglês, o termo *forensics* é definido como:

> a aplicação de conhecimento científico em problemas legais e especialmente

> a análise científica de evidências físicas como as encontradas em uma cena de crime[1].

Uma tradução muito utilizado comercialmente no Brasil é *forense computacional*, embora seja uma substantivação forçada do adjetivo forense, que não explicita a mesma semântica da definição do termo *forensics*. O termo mais próximo de *forensics* com base na definição apresentada seria, de fato, perícia e, por conseguinte, *Computer Forensics* poderia ser traduzido, adequadamente, como perícia em computadores.

Em relação à análise *live*, do inglês *live analysis*, este trabalho optou por manter o termo em inglês, tendo em vista que não há consenso sobre a tradução mais adequada. Opções como "ao vivo"ou simplesmente "viva"carregam uma conotação distinta. Análise "ao vivo"pode parecer mais adequado, no entanto indicaria que todo o processo de análise é realizado com o sistema ainda em execução, quando, na verdade, apenas a coleta é necessariamente feita nessa condição. Análise "viva", por outro lado, faz um contraponto à análise *post mortem*, termo comumente utilizado para se referir a análise de um sistema que foi desligado.

A definição utilizada neste trabalho para a Informática Forense é emprestada da definição de *Digital Forensic Science* discutida e apresentada em Palmer (2001). Assim, a Informática Forense é definida como:

> o uso de métodos cientificamente estabelecidos e comprovados para a preservação, coleta, validação, identificação, análise, interpretação, documentação e apresentação da evidência derivada de fontes digitais para o propósito de facilitar ou promover a reconstrução de eventos que causem a perturbação de operações planejadas.

O *Australian Institute of Criminology* apresenta uma definição mais adequada a esse ponto de vista, no qual a perícia em Informática é definida como:

> o processo de identificar, preservar, analisar e apresentar evidências digitais de uma maneira legalmente aceitável.

Ainda complementando a definição de Informática Forense, é preciso esclarecer a definição de evidência digital, que segundo Huebner et al. (2003) é:

> qualquer informação de valor probatório que é armazenada ou transmitida de forma digital.

Analogamente, o termo perícias em rede (*network forensics*) tem sido cada vez mais utilizado. Em Palmer (2001), é apresentada a seguinte definição de perícias em rede:

> é o uso de técnicas cientificamente comprovadas para coletar, unir, identificar, examinar, correlacionar, analisar e documentar evidências digitais de múltiplas fontes digitais processando e transmitindo ativamente, com o propósito

[1] Definição do dicionário Merriam-Webster disponível em `http://www.merriam-webster.com`

Tabela 6.1. Objetivos da Informática Forense em áreas diversas, adaptada de Palmer (2001)

Área	Objetivo primário	Objetivo secundário	Quando atua
Policial	Persecução penal	–	Depois do fato
Militar	Continuidade	Persecução penal	Tempo real
Comercial	Disponibilidade	Persecução penal	Tempo real

Tabela 6.2. Diferenças entre a segurança de computadores e a perícia em Informática, adaptada de Ruibin and Gaertner (2005)

Segurança	Perícia
Busca proteger o sistema de ataques	Não protege o sistema de ataques
Ação em tempo real ou logo após um incidente	Após os incidentes (*post mortem*)
Ambientes restritos para apresentação dos acontecimentos	A evidência é quase sempre apresentada para pessoal não técnico
Pode ser contornada por indivíduos confiáveis	A integridade da evidência é o mais importante

de descobrir fatos relacionados ao intento planejado ou sucesso apurado de atividades não autorizadas destinadas a perturbar, corromper ou comprometer componentes de sistema, bem como prover informação para auxiliar na resposta ou recuperação após atividades.

A definição das perícias em rede apresenta grande semelhança com as áreas de segurança de redes e resposta a incidentes, cujo objetivo principal é a proteção e a manutenção da disponibilidade de seus sistemas e redes. A Tabela 6.1, adaptada de Palmer (2001), apresenta os principais objetivos de cada área com relação à pesquisa e aplicação da perícia em Informática.

Em um cenário que não envolva o desejo de processar os atacantes, mas de proteger algum patrimônio ou informação, a ação invasora pode ser interrompida enquanto ocorre e, consequentemente, correções no sistema ou rede que implicam perda de evidências do ocorrido podem ser feitas. Ou seja, ações que podem ser boas práticas em respostas a falhas de segurança podem ser devastadoras do ponto de vista pericial. Essa situação é encontrada pelos profissionais de segurança de redes e detecção de intrusão, cujos procedimentos nem sempre estão em sintonia com os procedimentos periciais, já que a persecução penal, nesse caso, é uma preocupação secundária. Portanto, apesar das semelhanças entre a perícia em Informática e a segurança de computadores, tanto em termos de conhecimento quanto de ferramentas, existem algumas diferenças significativas, como as apresentadas por Ruibin and Gaertner (2005), aqui adaptadas e exibidas na Tabela 6.2.

A análise *live* busca aproximar os procedimentos das duas áreas. A realização do exame pericial tende a se aproximar do momento do incidente e a realizar procedimentos que, embora não mantenham o máximo de integridade da evidência, ainda são seguros do

ponto de vista jurídico.

6.3. Histórico

Segundo Huebner et al. (2003), o primeiro caso criminal relacionado ao uso de computadores conhecido foi registrado nos EUA, em 1966, e resultou em uma pena de cinco anos de prisão[2]. Nas décadas de 1970 e 1980, os computadores tornaram-se mais comuns e baratos, permitindo a utilização pessoal e comercial em maior escala. Com isso, a polícia identificou o surgimento de uma nova classe de crimes: os crimes relacionados a computadores. Nas décadas seguintes, a tecnologia passou por diversas evoluções significativas, que exigiram a adaptação contínua dos procedimentos periciais. Na década passada, a perícia em Informática caracterizou-se, predominantemente, pelo tratamento de dispositivos com capacidades de armazenamento relativamente pequenas e poucas quantidades de informação. Isso permitia que cópias completas dos discos rígidos originais fossem feitas para outro disco, sendo o exame realizado sobre a cópia, preservando, assim, a evidência original.

Com o surgimento da Internet comercial no Brasil no início de 1995[3], surge uma nova demanda relacionada à prática de crimes com o auxílio da Internet, hoje conhecidos comumente como crimes cibernéticos. Com o crescimento explosivo da Internet, também cresceram as ocorrências de incidentes relacionados, como a invasão de servidores e a prática de *defacement* de páginas web.

Além do crescimento da Internet, é importante destacar, também, a evolução da telefonia móvel e dos meios de armazenamento de dados. Ambos tornaram-se extremamente acessíveis e hoje fazem parte da vida cotidiana. Da mesma forma, pode-se afirmar que fazem parte do cotidiano de criminosos, que muitas vezes fazem uso dessas tecnologias, mesmo que os crimes que cometam não tenham relação direta com a Informática. Hoje, discos rígidos com capacidade da ordem de terabytes podem ser adquiridos. Esse grande volume de dados e diversidade de mídias é um dos grandes desafios enfrentados pela Informática Forense, que precisa buscar novas soluções para cenários onde não é mais possível realizar uma cópia integral de todos os dados originais, como em ambientes de rede complexos.

No Brasil, cabe ainda considerar a carência de uma legislação específica para punir diversas condutas no meio cibernético, que em muitos países já são consideradas crimes, como a disseminação de programas maliciosos (*malware*). Diversos projetos de lei foram propostos ao longo dos anos, mas até a conclusão deste trabalho nenhum havia sido aprovado em caráter conclusivo.

6.4. Desafios da perícia digital

Muitos dos desafios enfrentados hoje na Informática Forense são produto dos grandes avanços tecnológicos observados nos últimos 15 anos. Esta seção apresenta alguns do desafios mais discutidos e que são os principais alvos de pesquisas.

Nas discussões do *First Digital Forensic Research Workshop*, ocorrido em 2001, e

[2]Tratava-se de um caso de furto de programa de computador.

[3]Considerando a data de criação do Comitê Gestor de Internet em maio de 1995.

apresentado em Palmer (2001), alguns dos desafios de alta prioridade citados então eram a confiabilidade da evidência digital e as perícias em ambientes de rede. Essas questões ainda estão presentes, atualmente, e em escala cada vez maior. As evidências digitais tornaram-se cada vez mais comuns e as redes de computadores, maiores e mais presentes. Comentou-se, então, que a ubiquidade dos sistemas de informática e equipamentos eletrônicos, cada vez mais, indicava que um dia todos os crimes teriam uma "ciberdimensão".

Uma tendência observada nos últimos anos é a expansão da perícia além do simples exame dos discos rígidos. A análise de memória volátil e de sistemas em operação tem recebido bastante atenção em termos de pesquisa e de ferramentas específicas. É interessante notar que a perícia em sistemas em operação "desrespeita"um dos princípios básicos da perícia em mídias de armazenamento, que é a preservação total dos dados originais. Isso porque qualquer atividade em um sistema em operação causa mudanças nos dados armazenados na memória. Como tal situação é inevitável, Huebner et al. (2003) argumentam que evidências coletadas dessa forma têm que ser aceitáveis em juízo.

Além das dificuldades discutidas acima, que afetam de maneira geral o trabalho pericial em Informática, Huebner et al. (2003) apresenta ainda diversos desafios técnicos como:

1. sistemas de arquivos que permitem ocultar dados do usuário comum, sendo visíveis apenas se utilizadas ferramentas especiais;

2. propriedades e mecanismos de sistemas operacionais e aplicativos sem documentação ou utilizados para ocultar dados;

3. armazenamento de dados *online*, que permite o armazenamento de dados em serviços da Internet, cujo acesso pode ser difícil ou pode encontrar-se fora da jurisdição legal daquela polícia e pode até requerer ações demoradas de cooperação internacional;

4. uso extensivo de criptografia forte, que sugere a necessidade de um trabalho maior de investigação para evitar que as evidências digitais do suspeito estejam protegidas dessa forma;

5. dispositivos móveis de alta capacidade e dimensões muito reduzidas, que podem ser facilmente destruídos ou ocultados, ou que podem ser utilizados para evitar que dados importantes fiquem armazenados nos discos rígidos dos computadores utilizados;

6. serviços *online* diversos como *webmail*, redes sociais ou programas de mensagens instantâneas, cujos vestígios encontram-se nas mãos dos provedores dos serviços, o que dificulta sua coleta.

O uso de técnicas cada vez mais sofisticadas de proteção e ocultação de dados, como criptografia integral de disco e esteganografia, também reforçam a necessidade de uma ação mais proativa para identificar e preservar vestígios. A análise *live* surge como uma possível resposta para alguns desses desafios.

6.5. Princípios de perícia digital

Segundo Palmer (2001), por definição, a perícia em Informática tem uma natureza investigativa e seus praticantes devem seguir um processo investigativo na realização de seu trabalho. Ao investigar crimes relacionados a computadores, deve ficar claro que os princípios básicos aplicados a cenas de crime "comuns"também se aplicam. Nesta seção, são apresentados alguns princípios de perícia digital que serão utilizados ao longo do trabalho. Também são detalhadas as fases do trabalho pericial, que se aplicam tanto à análise convencional quanto à análise *live*.

De acordo com Huebner et al. (2003), a primeira coisa de que um investigador deve estar ciente é o *Princípio da Troca de Locard*, segundo o qual

> qualquer pessoa ou coisa entrando em uma cena de crime leva algo da cena consigo ou deixa algo de si para trás quando sai da cena,

ou, como apresentado por Reith et al. (2002), toda atividade em um computador provavelmente produzirá uma modificação no sistema em que foi realizado como, por exemplo, modificações no sistema de arquivos ou, no mínimo, modificações na sua memória principal. Nesse sentido, um princípio básico é o da preservação dos vestígios originais. Sempre que possível, procura-se trabalhar sobre uma cópia integral e exata dos dados originais. Um alto nível de integridade dos vestígios é necessário em todos os exames periciais de Informática, já que materiais digitais são mais facilmente adulterados e forjados do que materiais físicos.

Com o desenrolar dos exames e a descoberta de evidências no material examinado, é importante manter a rastreabilidade dessas descobertas e de suas correlações. Da mesma forma, os dados são transformados e interpretados por ferramentas diversas. É desejável que todos os procedimentos realizados sejam claros e totalmente compreendidos pelos especialistas, embora nem sempre as ferramentas utilizadas permitam uma análise e avaliação mais profunda do seu funcionamento.

6.5.1. Cadeia de custódia

A cadeia de custódia é um processo usado para manter e documentar a história cronológica da evidência, para garantir a idoneidade e o rastreamento das evidências utilizadas em processos judiciais. A cadeia de custódia trata dos procedimentos que buscam garantir a idoneidade das evidências por meio da descrição e documentação detalhada de como a evidência foi encontrada e de como foi tratada dali por diante.

Todo o procedimento deve ser documentado para que fique registrado onde, quando e por quem a evidência foi descoberta, manipulada, coletada e armazenada. Quando a evidência passa para a responsabilidade de outra pessoa, esse fato, com todos os detalhes envolvidos, incluindo número de lacres e outros procedimentos de segurança, deve ser, também, cuidadosamente documentado. Turner (2005) discute extensivamente a importância de comprovar a procedência das evidências obtidas em meios digitais. A incapacidade de demonstrar a continuidade dessa cadeia de custódia em um processo tem um sério impacto na aceitação da prova.

6.5.2. Fases

Beebe and Clark (2005) sugerem uma divisão em seis fases, nas quais um número arbitrário de subfases pode ser definido, considerando a necessidade de sua execução conforme a natureza do caso. As seis fases são:

1. preparação (pré-incidente);
2. resposta ao incidente;
3. coleta de dados;
4. análise;
5. apresentação das descobertas;
6. encerramento do incidente.

A Figura 6.2 apresenta a relação das seis fases com a possibilidade de iteração entre elas ou de todo o processo.

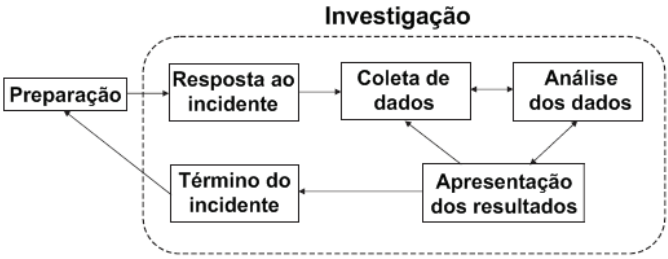

Figura 6.2. Fases do trabalho pericial, adaptada de Beebe and Clark (2005)

A diferença da análise *live* para a convencional está principalmente na fase de coleta de dados e análise, que são realizadas com o sistema ainda em execução. Na análise convencional, o sistema seria desligado antes de qualquer outro procedimento ser realizado.

6.5.3. Funções de *hash*

As funções de *hash* têm uma importância fundamental nos procedimentos da perícia digital. Elas são funções que relacionam uma entrada de tamanho variável a uma saída de tamanho fixo. Essas funções de *hash* são comumente utilizadas em criptografia, autenticação e assinatura digital. Alguns exemplos de algoritmos utilizados são MD5, SHA-1, SHA-256 e SHA-512. Uma característica fundamental das funções de *hash* é que é muito difícil encontrar dois conteúdos de entrada que produzam o mesmo resultado na saída, e, a partir da saída, é computacionalmente inviável encontrar a entrada. Dessa forma, essas funções são utilizadas para garantir a integridade de arquivos digitais. No caso de

modificação no conteúdo de um arquivo, mesmo que mínima, o valor da função de *hash* muda drasticamente, evidenciando a existência de alteração. O quadro a seguir apresenta um exemplo de modificação no conteúdo e seu reflexo no resultado da função de *hash* utilizando o algoritmo MD5.

```
> Conteúdo: <test>
>> MD5: a55ab7512f0d0ff4527d898d06afd5c5

> Conteúdo: <teste>
>> MD5: c0d810f61f37025e600cf41e716c8576
```

A utilização das funções de *hash* para garantir a integridade das evidências encontradas será abordada posteriormente neste trabalho, como parte dos procedimentos a serem executados durante a perícia, assim como uma indicação de ferramentas para realizar o processo.

6.6. Análise *live*

Conforme descrito inicialmente, a análise *live*, também chamada de *live forensics*, consiste em uma análise digital realizada por meio de procedimentos periciais e conduzida no equipamento computacional ainda em execução. Portanto, a análise *live* ocorre quando o sistema é mantido em execução e os investigadores usam o próprio sistema operacional da máquina para acessar os seus dados.

Segundo Anson and Bunting (2007), os ingredientes principais para realizar uma análise *live* bem-sucedida são:

- interagir o mínimo possível com o sistema em análise;
- utilizar ferramentas confiáveis;
- pensar e repensar, pois uma vez feito o procedimento em um sistema em execução, o sistema modificará o estado atual, sendo impossível retornar ao estado inicial;
- documentar todo o procedimento.

Para melhor compreender o potencial da análise *live* com relação a análise convencional, pode-se fazer uma reflexão sobre as fontes de dados existentes em um computador, conforme apresentado na seção a seguir.

6.6.1. Fontes de dados em um computador

Os componentes de um computador são agrupados em três componentes básicos: unidade central de processamento (CPU), a memória principal (RAM) e os dispositivos de entrada e saída. A Figura 6.3 ilustra a interação desses componentes.

Processadores O processador, ou Unidade Central de Processamento (CPU), tem como função principal unificar todo o sistema, controlando as funções realizadas pelos outros

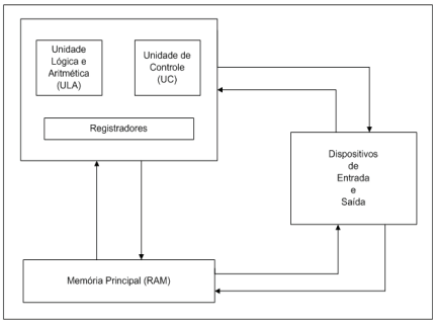

Figura 6.3. Interação entre os componentes básicos do computador

componentes. A CPU é composta por dois componentes básicos, a unidade de controle (UC), e a unidade lógica e aritmética (ULA).

A função da CPU é buscar instruções na memória e executá-las, em seguida. Seu ciclo básico de execução é buscar a instrução da memória, decodifica-la para determinar seus operandos e funções a executar, executá-la, e em seguida, tratar a instrução seguinte, até que o programa pare.

Registradores São dispositivos de alta velocidade, localizados fisicamente na CPU, para armazenamento temporário de dados. Um registrador é o elemento superior da pirâmide da memória, por possuir a maior velocidade de transferência dentro do sistema, menor capacidade de armazenamento e maior custo. O sistema operacional deve estar sempre atento ao estado e ao conteúdo dos registradores. Quando o sistema operacional compartilha a CPU com mais de um programa, necessita, às vezes, interromper um programa e iniciar outro. Nesse caso, é necessário que os dados contidos nos registradores sejam salvos, para que possam ser recuperados posteriormente, quando seu programa de origem voltar a ser executado.

Memória cache Esta memória, hierarquicamente, está abaixo da camada de registradores, sendo controlada, principalmente, por *hardware*. É uma memória de alta velocidade, mais lenta que os registradores, mas mais rápida que a memória principal. Os modernos computadores costumam ter dois ou até três níveis de cache, sendo o seu tamanho limitado pelo alto custo. A cada nível subsequente, diminui a velocidade e aumenta a capacidade de armazenamento. Todas as requisições da CPU que não podem ser atendidas pela memória cache são direcionadas para a memória principal.

Memória principal (RAM) Também conhecida como memória primária, real ou RAM (*random access memory*). Posições de memória frequentemente ou recentemente utilizadas são mantidas na memória cache. Quando o programa precisa ler um dado na memória, o *hardware* verifica se o dado está na memória cache. Se estiver (*cache hit*), nenhuma

requisição adicional é necessária. Caso contrário (*cache miss*), há necessidade de uma requisição adicional, enviada à memória principal, com perda substancial de tempo.

Memória secundária É um meio permanente de armazenamento. Enquanto os dados contidos em registradores, memória cache e memória principal são voláteis – sendo perdidos no momento de desligamento do computador – a memória secundária permanece armazenada mesmo depois do desligamento da máquina. Trata-se de uma memória de acesso bem mais lento, quando comparado às memórias voláteis. Sua vantagem, porém, está no menor custo e na alta capacidade de armazenamento. Exemplos desse tipo de memória são os discos rígidos e os flash drives.

Dispositivos de entrada e saída São os dispositivos que permitem a comunicação entre o computador e o mundo externo. Podem ser divididos em duas categorias: na primeira, estão os dispositivos utilizados como memória secundária e na segunda, os dispositivos que permitem a interação do ser humano com o computador, como teclado, monitor, mouse, impressora e scanners.

O objetivo de discutir os componentes da arquitetura básica do computador é chamar a atenção para as possíveis fontes de evidências digitais. Observa-se que a volatilitade diminui a medida que a capacidade de armazenamento aumenta. Elementos com maior volatilidade como registradores e caches não apresentam vantagem do ponto de vista pericial em relação ao conteúdo da memória RAM, que por ser menos volátil é mais fácil de ser analisada. Assim como os discos rígidos apresentam um volume bem maior de material potencialmente interessante para o exame pericial. Deve-se ter em mente que apesar dos dispositivos de entrada e saída não terem função primordial de armazenamento de dados, alguns deles fazem uso de algum tipo de memória para viabilizar sua operação. Como exemplo, algumas impressoras utilizam um disco rígido que armazena cópias de arquivos a serem impressos.

6.7. Diferenças com relação à perícia convencional

A perícia convencional ocorre com o sistema investigado desligado. Para evitar novas escritas no disco, remoção de arquivos temporários ou qualquer modificação no sistema, aconselha-se desligar o computador utilizando o procedimento *pull the plug*. Esse procedimento consiste na interrupção do fornecimento de energia ao equipamento pela retirada do cabo de energia da tomada. Após a coleta do equipamento computacional, uma cópia integral do disco rígido do sistema é realizada e, a partir daí, essa imagem é analisada em laboratório, utilizando-se um sistema operacional e aplicações forenses confiáveis (Carrier, 2006).

Diferentemente da perícia convencional, que fornece apenas uma visão limitada das informações do sistema, ferramentas para análise *live* podem informar ao investigador um cenário mais completo do estado do computador (Hay et al., 2009). Segundo Adelstein (2006), enquanto a perícia convencional tenta preservar os discos rígidos em um estado inalterado, as técnicas de análises em equipamentos computacionais ligados têm como objetivo tirar *snapshots* do estado da máquina, similar às fotografias de uma

cena de crime.

Ferramentas periciais, na maioria das vezes, são bem-sucedidas na extração de dados dessas mídias, inclusive na recuperação de arquivos apagados não sobrescritos e buscas por palavras-chave. A perícia convencional, apesar de amplamente usada atualmente na persecução penal, apresenta limitações nos seguintes casos:

1. impossibilidade de coletar o equipamento computacional;
2. necessidade de estabelecer o flagrante do suspeito;
3. uso de criptografia forte.

No item 1, existem casos em que não há permissão legal ou viabilidade técnica para coletar o equipamento computacional devido à importância deste para a organização. Equipamentos de grande porte, como *mainframes*, também inviabilizam sua apreensão pela dificuldade de transporte e armazenamento do *hardware*.

No item 2, o desligamento sumário do equipamento computacional inviabilizará o flagrante, já que não haverá a constatação dos requisitos necessários para a sua configuração. Sendo assim, a prova deve ser extraída e documentada antes do procedimento de desligamento e coleta do equipamento.

O item 3 refere-se ao mais recente desafio da perícia digital: o uso, cada vez mais difundido, de esquemas criptográficos robustos nos computadores pessoais, dificultando, consideravelmente, a extração de dados pela perícia convencional, podendo até mesmo inviabilizá-la por completo.

A análise *live* possui vantagens e desvantagens se comparada com a perícia convencional. Segundo Anson and Bunting (2007), o investigador deve determinar qual opção representa uma ameaça maior à perícia: a perda dos dados da memória RAM ou a modificação dos dados no disco rígido.

6.8. Vantagens da análise *live*

Devido à análise em um computador ligado fornecer uma visão mais completa do sistema investigado, com acesso a informações na memória RAM, ela pode ser usada para resolver algumas das limitações encontradas pela perícia convencional. Entre as vantagens obtidas no uso desse tipo análise, é possível elencar como as principais:

- extração de dados voláteis;
- triagem de equipamentos;
- triagem de dados;
- preservação de dados criptografados;
- possiblidade de estabelecer o flagrante.

Cada uma dessas vantagens é detalhada nas seções que se seguem.

6.8.1. Extração de dados voláteis

De acordo com Adelstein (2006), a análise *live* pode resguardar tanto as informações voláteis quanto as informações estáticas sobre o sistema de arquivos. De acordo com Carrier and Spafford (2005), o pré-processamento de dados na cena de crime é apenas uma das fases na investigação digital. A perícia em um computador ainda ligado permite ao investigador analisar um elemento indisponível durante a perícia convencional: a memória RAM. Sendo assim, o investigador terá acesso a informações não tipicamente escritas em disco, tais como: portas abertas, conexões de rede ativas, programas em execução, dados temporários, interação com usuário, chaves criptográficas e conteúdos não criptografados.

Com o uso apenas da perícia convencional, essas informações eram, simplesmente, ignoradas, o que pode ser prejudicial à investigação. Quanto maior a probabilidade de alteração nas informações do dispositivo computacional, maior é a prioridade de extração e preservação desses dados. Como a memória RAM é mais suscetível a mudanças, alerta que a extração deve seguir a ordem de volatilidade. Portanto, na maioria das vezes, é necessário que as informações sejam extraídas da memória antes da extração dos dados do disco rígido.

6.8.2. Triagem de equipamento

A presença crescente de computadores e mídias de armazenamento computacional na vida cotidiana refletiu-se também nas cenas de crime, onde comumente são encontrados computadores que apresentam relação com o fato sob investigação (Hoelz, 2009). Segundo Adelstein (2006), discos rígidos com mais capacidade de armazenamento aumentam o tempo necessário para análise, dificultando e encarecendo-a quando há a coleta de todos os discos rígidos.

A análise *live* permite ao investigador filtrar os equipamentos computacionais que são realmente de interesse à investigação. A busca por palavras-chave no disco rígido ou por aplicativos instalados na máquina, por exemplo, podem evitar a apreensão desnecessária de computadores. Em casos nos quais os resultados da perícia devem ser disponibilizados em um curto espaço de tempo, um modelo para triagem de equipamentos pode ser utilizado durante a análise *live* (Rogers et al., 2006). Esse modelo envolve consultas no computador investigado em busca de informações contidas nos arquivos de registro, histórico de Internet, mensagens eletrônicas entre outros. A triagem de equipamentos ajuda o perito a dedicar-se à perícia dos equipamentos computacionais relevantes, pois reduz a quantidade total de equipamentos apreendidos. Sendo assim, as análises resultam em um relatório com mais qualidade e tempestividade.

6.8.3. Triagem de dados

Devido ao atual aumento da quantidade de evidências digitais disponíveis, em breve será impossível obter todos os dados referentes ao caso. Com isso, o paradigma da análise *live* poderá se tornar o procedimento padrão. Técnicas como mineração de dados e uso de filtros de arquivos conhecidos estão sendo utilizadas para processar casos contendo grande volume de dados, porém não resolvem o problema por completo.

A extração seletiva de dados no computador investigado em execução pode facilitar a perícia posterior, principalmente em casos onde os dados estão em servidores

corporativos (banco de dados, servidores de e-mail), *mainframes* ou máquinas que contenham *hardware* que dificulte a perícia convencional, como RAID (*Redundant Array of Independent Disks*), por exemplo. Segundo Aquilina et al. (2008), nem sempre é possível extrair todos os dados de todas as máquinas envolvidas no incidente, sendo mais eficiente a extração de alguns dados de cada máquina para determinar quais sistemas realmente foram afetados.

6.8.4. Preservação de dados criptografados

A criptografia é um dos melhores métodos para ocultar informação e tem sido amplamente utilizada por criminosos para esconder o conteúdo de seus arquivos. O uso de volumes criptografados complica, significativamente, a perícia convencional. Supondo o uso de algoritmos fortes, métodos convencionais de investigação, em geral, possuem um baixo retorno em função do investimento. Uma vez que o sistema é desligado, a chave criptográfica necessária para acessar a mídia de armazenagem normalmente não está mais disponível (Hay et al., 2009). Houve avanços no processamento de dados criptografados, como o uso de *rainbow tables* e ataques por dicionário. No entanto, técnicas anti-forenses também evoluíram para dificultar a decifração desses conteúdos criptografados. Caso o sistema não seja desligado, a análise *live* permite ao investigador acessar os dados de forma transparente, como um usuário do sistema, e realizar uma cópia para analisá-los posteriormente.

6.8.5. Possibilidade de estabelecer flagrante

Outra grande vantagem do uso de técnicas de análise em computadores ligados é a possibilidade de constatação de uma situação de flagrante. A perícia convencional simplesmente ignorava tal possibilidade e desligava a máquina investigada, perdendo a oportunidade de registrar o estado da máquina, processos em execução e arquivos que serviriam para estabelecer a situação de flagrante.

6.9. Desvantagens da análise *live*

Apesar de resolver alguns problemas encontrados na perícia convencional, a perícia em equipamentos computacionais em execução também introduz novos desafios e possui suas próprias limitações. Além de aumentar a quantidade de informações que o perito deve analisar, é possível citar as seguintes desvantagens no uso desse tipo de análise:

1. aspectos legais e impossibilidade de reprodução do exame;
2. tempo gasto;
3. complexidade e variedade de cenários;
4. mudança de paradigma na investigação;
5. *rootkits* e *malware*.

Analogamente, cada uma dessas devantagens é detalhada nas seções que se seguem.

6.9.1. Aspectos legais e impossibilidade de reprodução do exame

É necessário considerar as indagações sobre aspectos legais como uma parte do esforço de pesquisa nas análises de sistemas em execução (Hay et al., 2009). Caso a prova digital não seja válida legalmente, pouco adianta os resultados da análise *live*.

Durante a realização da perícia em um computador ligado, é muito fácil contaminar a prova no sistema, exigindo que os procedimentos sejam feitos por um profissional qualificado. Durante a realização dos exames, o sistema se modifica continuamente, impossibilitando obter exatamente os mesmos resultados ao se repetir a perícia. A boa prática exige que o investigador, quando realizando um procedimento em uma máquina investigada, minimize o impacto e compreenda o efeito desse procedimento no sistema analisado.

Uma vez que a análise *live* extrai dados da memória volátil, esse exame não poderá ser repetido posteriormente produzindo exatamente os mesmos resultados. Essa impossibilidade de reprodução exige uma documentação precisa dos procedimentos realizados e uma atenção ainda maior na preservação da integridade dos dados extraídos durante a análise dos equipamentos computacionais ligados.

Os dados extraídos durante os exames devem seguir o mesmo tratamento dispensado à perícia convencional. Deve-se utilizar uma função de *hash* para garantir integridade desses dados. Além disso, a análise *live* deve se manter em harmonia com os preceitos da cadeia de custódia. O objetivo de manter, cuidadosamente, a cadeia de custódia não consiste apenas em proteger a integridade da evidência, mas, também, de tornar difícil ao advogado de defesa arguir que a evidência foi mal manipulada enquanto esteve na posse do investigador.

6.9.2. Tempo gasto

A utilização racional do tempo é sempre importante e, para tanto, a escolha entre as atividades de análise que devem ser feitas no ambiente em execução e quais podem ser realizadas posteriormente sem prejuízo às investigações é essencial.

Alguns procedimentos utilizados quando os computadores estão ligados, tais como a cópia integral do disco rígido, podem ser demorados. O local da análise *live*, ao menos no caso policial, não é o local mais adequado para realização de exames periciais. Sendo assim, espera-se que esses procedimentos sejam os mais breves possíveis. Para Adelstein (2006), o investigador pode realizar uma triagem e coletar dados essenciais, examiná-los e usar o resultado dessa análise para decidir o que é mais necessário.

6.9.3. Complexidade e variedade de cenários

A grande quantidade de aplicativos, sistemas operacionais e dispositivos computacionais encontrados durante a perícia em um equipamento ligado frustram a preparação do perito na realização desse tipo de análise. A preparação é requisito fundamental para o sucesso da análise, sendo inadmissível testar ferramentas e procedimentos durante a realização da análise *live* (Mandia et al., 2003).

O sucesso nesse tipo de análise depende de um treinamento constante do perito na área de computação, assim como do estudo e da evolução contínua dos procedimentos

periciais realizados em um computador ligado. Como visto, o investigador deve ter um conhecimento amplo em diversas áreas computacionais, porém é humanamente impossível exigir o domínio em todas as particularidades encontradas durante a análise *live*, sendo necessária a utilização de uma ferramenta para auxiliá-lo nesse processo.

6.9.4. Mudança de paradigma na investigação

Para a realização de uma análise *live*, é indispensável que a máquina esteja ligada. Caso a máquina esteja desligada, não existe nada mais a fazer, além de coletar o equipamento para realização de perícia convencional e esperar que a máquina não esteja utilizando algum algoritmo de criptografia forte. Caso seja necessário realizar a análise de uma máquina ligada, a investigação deve se adaptar à necessidade imposta por esse tipo de análise, ou seja, realizar a coleta apenas se tiver certeza de que o equipamento computacional estará ligado.

6.9.5. *Rootkits* e *malware*

Um *rootkit* é um *software* que permite acesso privilegiado e contínuo a um computador, ao mesmo tempo em que fica invisível aos administradores do sistema, subvertendo as respostas normais e esperadas de comandos do sistema operacional ou de outros aplicativos. O termo *rootkit* é uma concatenação dos termos *root* e *kit*. *Root* é nome tradicional da conta com privilégios de administrador do sistema, nos sistemas operacionais UNIX, enquanto o termo *kit* refere-se aos componentes de *software* com integram a ferramenta. Tipicamente, o *rootkit* é instalado na máquina pelo atacante, após ter obtido poderes de administrador do sistema, explorando alguma vulnerabilidade conhecida ou tendo acesso à senha de administrador. Os *rootkits* são de difícil detecção, já que podem subverter o próprio *software* que supostamente deveria detectá-lo.

Rootkits podem ser classificados em dois níveis: de aplicativo e de *kernel*. O primeiro tipo é encontrado em aplicativos nativos do próprio sistema operacional, sendo capaz de omitir resultados de uma consulta desse aplicativo modificado por meio de um filtro pré-determinado. Já o segundo é incorporado ao sistema operacional e, por isso, utiliza um filtro para omitir resultados independentemente do aplicativo que está sendo executado. Segundo Carrier (2006), os *rootkits* de nível de aplicativo podem ser contornados utilizando executáveis conhecidos e trazidos pelo próprio investigador, mas os *rootkits* de nível de *kernel* necessitam da utilização de uma abordagem mais complexa, sendo necessário buscar inconsistências ao correlacionar diversas estruturas em memória.

Além dos *rootkits*, é importante estar atento para a presença de *malware*. Este termo vem do inglês (*malicious software*), significando *software* malicioso. Refere-se a programas desenvolvidos para alterar ou danificar o sistema, roubar informações ou provocar outras ações não realizadas pelo usuário atual. Exemplos comuns de *malware* incluem vírus, *worms*, *trojans* e *spyware*. Embora não afetem diretamente os resultados do exame, como no caso dos *rootkits*, podem ser utilizados como estratégia de defesa, com a alegação de que "a culpa foi do *malware*".

A análise e engenharia reversa de *malware* é um assunto amplo e de grande valia para o exame pericial, mas que está além do escopo deste trabalho.

6.10. Análise da memória RAM

A grande vantagem do uso da análise *live* em relação à análise convencional é o acesso aos dados residentes na memória do computador investigado. Apesar de ser substancialmente menor que a capacidade de armazenamento dos discos rígidos, a memória RAM é uma rica fonte de informações para investigação. É possível extrair da memória informações como: processos em execução, arquivos abertos, conexões estabelecidas, portas abertas e possíveis senhas. Em sistemas Windows pode-se extrair a lista de DLLs e os arquivos de registro (*hive files*), que permanecem residentes na memória. Essas informações, com exceção dos arquivos de registro – que também são acessíveis pelo disco rígido –, eram simplesmente ignoradas pela perícia convencional. Com o uso da perícia em equipamentos em execução, essas informações ganham papel de destaque, tornando esse tipo de análise uma importante fase da perícia digital.

A RAM é uma memória volátil e bastante dinâmica que exige cuidado durante sua análise. Devido à volatilidade da memória, a análise em equipamentos computacionais ligados oferece ao perito uma breve janela de oportunidade para extração de dados desta, antes que seja efetuado o desligamento e apreensão do equipamento computacional. A memória RAM de um computador ligado permanece continuamente alterando seu conteúdo. Sendo assim, torna-se necessária uma documentação criteriosa das ações realizadas pelo perito.

6.10.1. Análise com ferramentas específicas

Além de aplicativos próprios do sistema operacional, existem outras ferramentas digitais forenses, aplicativos de administração de rede e utilitários de diagnósticos, que podem ser usados durante a análise *live*. Não se pode subestimar a importância do processo monótono de criação de um kit de ferramentas para realizar esse tipo de análise. O tempo gasto nesse processo será compensado pelos resultados mais rápidos, profissionais e bem-sucedidos (Mandia et al., 2003).

Mesmo utilizando ferramentas específicas, é possível a ocorrência de falsos negativos caso um *rootkit* de nível de *kernel* esteja instalado no sistema operacional. Sendo assim, o sistema operacional pode omitir informações de interesse à investigação independentemente da ferramenta que esteja sendo utilizada.

6.10.2. Análise do *dump* da memória

Neste tipo de análise, o conteúdo integral da memória RAM é copiado para um arquivo denominado *dump*. Este arquivo é então processado em um ambiente preparado e controlado por meio do uso de analisadores, responsáveis por percorrer o conteúdo do arquivo em busca de padrões. Na análise do arquivo de *dump* da memória é feita a busca por assinaturas das estruturas de dados em memória.

Por não utilizar as APIs do próprio sistema operacional, essa abordagem de análise apresenta uma desvantagem: incompatibilidade com os diversos sistemas operacionais e versões. As estruturas de dados usadas e a organização dos elementos carregados em memória dependem do sistema operacional do computador. Suas assinaturas variam com cada versão do sistema operacional, incluindo atualizações significativas como *services packs*. É necessário conhecer profundamente essa organização para realizar a extração

desses elementos do arquivo da memória *dump*.

Apesar de não ser possível extrair novamente o conteúdo da memória após o desligamento da máquina investigada, é possível reanalisar o arquivo de *dump*. Essa propriedade é desejável para a filosofia da perícia digital, já que permite uma reprodução parcial do procedimento, além de permitir a extração de outros dados não extraídos durante a primeira análise.

6.10.3. Mecanismos para cópia da memória RAM

Diferentemente da duplicação de discos rígidos pela perícia convencional, a memória RAM não pode ser desconectada do sistema para realização de cópia devido à volatilidade de seus dados. Conforme a arquitetura física, uma vez que a memória não é mais alimentada de energia, o estado dos dados na RAM é desconhecido. Existem duas formas para realizar a cópia total dos dados da memória RAM para um arquivo de *dump*: com suporte de *software* ou de *hardware*.

Cópia com suporte de *software* Consiste na técnica mais utilizada para cópia do conteúdo da memória física. Existem diversos aplicativos que realizam essa cópia, porém como esses aplicativos são executados no próprio sistema operacional investigado, há uma alteração do conteúdo na memória. Sendo assim, o examinador deve dar preferência às ferramentas menos invasivas, ou seja, que alterem o mínimo possível do conteúdo da memória. Um conjunto contendo um bloqueador de escrita para extração de memória volátil de um computador investigado integrado pode ser utilizado para diminuir ainda mais a interação com o sistema.

Cópia com suporte de *hardware* Devido à volatilidade dos dados, não existe um *hardware* específico para realizar uma cópia do conteúdo da memória RAM, porém é possível utilizar a propriedade DMA (*Direct Memory Access*) de alguns dispositivos de *hardware* para realizar essa cópia. O DMA fornece transferência de dados entre o barramento PCI e a memória sem que seja necessário usar recursos do processador. A cópia física do conteúdo da memória RAM é possível pelo acesso direto à memória na porta *Firewire* (IEEE 1394). Apesar de menos invasiva em relação à cópia física por *software*, essa abordagem não é sempre possível. Porém, é recomendada quando o computador encontra-se bloqueado com senha pelo usuário. Essa técnica permite o acesso à memória do computador bloqueado e, por meio da alteração de processos em execução, seu desbloqueio.

6.11. Metodologia

A Figura 6.4 apresenta a metodologia para a realização da análise *live*. Um dos aspectos mais importantes do campo da computação forense é a documentação. Além de documentar seus próprios atos durante a coleta dos dados, o analista deve também documentar o ambiente antes de começar efetivamente a intervir nos sistemas. Para que a documentação seja feita da melhor forma possível, é recomendado que haja uma pessoa responsável exclusivamente por essa tarefa. Alguns itens que devem ser documentados com atenção incluem:

1. telas do computador, com resolução suficiente para leitura de textos ali presentes, se necessário;

2. conexões de rede, mostrando quaisquer cabos de rede conectados ao computador. As duas pontas do cabo devem ser fotografadas, para o caso em que o analista tenha que provar que o computador estava conectado a algum equipamento específico;

3. conexões de periféricos, para provar que estavam conectados ao computador.

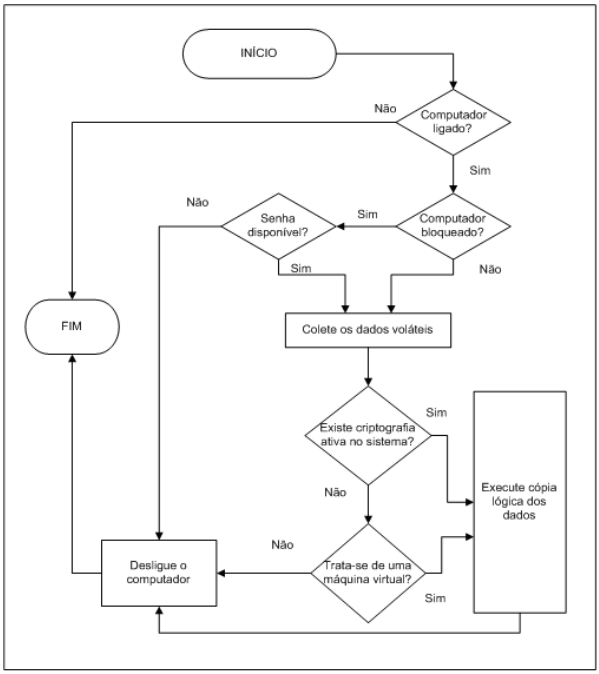

Figura 6.4. Metodologia para a realização da análise *live*

Deve-se impedir que um equipamento ligado seja desligado por intervenção humana ou que fique indisponível, por exemplo, em modo de hibernação automática ou de proteção de tela com senha, já que será alvo de captura de dados voláteis. É necessário, também, impedir que outras evidências sejam ocultadas, adulteradas ou destruídas. Um reconhecimento do local, a fim de localizar equipamentos, verificando as conexões citadas anteriormente deve ser realizado.

É recomendado que sejam tiradas fotografias para ilustrar o estado em que se encontra o ambiente. As fotografias, de preferência digitais, e com bastante resolução, devem fazer parte da documentação. A presença de testemunhas é sempre recomendada e em muitos casos é uma exigência processual. De preferência, devem ser utilizadas

máquinas fotográficas com capacidade para salvar as fotos no formato TIFF, evitando a perda de qualidade causada pela compressão para o formato JPEG.

Se o computador estiver desligado, deve assim permanecer e deve ser levado para exames posteriores, em laboratório. A integridade das fotografias deve ser garantida por meio de uma função de *hash*, cujo resultado deve fazer parte da documentação, permitindo verificação posterior da autenticidade das fotografias. Todas as informações relevantes para futuros exames, tais como senhas, nomes de usuários ou peculiaridades da configuração dos sistemas, devem ser documentadas. Muitas vezes elas podem ser solicitadas ao responsável técnico pelo local. Se o computador estiver ligado, deve-se tirar uma fotografia da sua tela e providenciar algum mecanismo que o impeça de hibernar ou ativar a proteção de tela, pois pode estar protegido por senha e não ser possível voltar a ter acesso ao sistema.

Em seguida, é inserido algum dispositivo externo como *pen drive* ou disco rígido contendo os aplicativos automatizados de coleta de dados voláteis. O simples ato de conectar esse dispositivo gerará alteração no registro do Windows. Portanto, deve-se registrar os dados do dispositivo usado a fim de identificá-lo posteriormente em meio aos resultados. As ferramentas são, então, executadas para realizar a coleta de dados. Os procedimentos de coleta são apresentados na seção seguinte.

Os resultados devem ser armazenados no próprio dispositivo externo. Eles devem ser analisados em sequência para identificar a possível presença de criptografia de volumes, criptografia de sistema ou virtualização do sistema operacional, o que exige procedimentos específicos.

Após a coleta, o dispositivo deve ser inserido em um computador do examinador para os procedimentos finais. Caso tenham sido tiradas fotografias durante o procedimento, elas devem ser juntadas aos resultados da coleta. Deve-se aplicar, então, uma função de *hash* em cada arquivo e gravar todos os valores em um arquivo que acompanhará os demais.

Para fins de logística, sugere-se que todos os arquivos sejam transferidos para mídias óticas, como CD ou DVD. Considerando os tamanhos comuns de memória física, basta um DVD, na maioria dos casos, ou dois, em casos excepcionais. Dessa forma, o analista forense continua com o dispositivo de coleta e a mídia ótica pode acompanhar o trâmite do processo judicial sem qualquer prejuízo. No entanto, um processo adicional é necessário para garantir que não haja substituição dessa mídia ou adulteração do seu conteúdo. Para isso, deve-se calcular o *hash* do arquivo de *hashes* citado anteriormente. O resultado deve ser registrado no relatório da perícia, que conterá também a assinatura do analista. Com isso, é possível garantir a integridade e a autenticidade dos resultados encaminhados.

Nos casos em que houver necessidade de cópia lógica de arquivos para discos rígidos externos, devido ao maior volume de dados, o próprio disco rígido utilizado deve acompanhar o restante do material de informática relacionado apreendido. Novamente, o mesmo procedimento de cálculo de *hashes* dos arquivos coletados deve ser realizado.

6.12. Coleta de dados

A coleta de dados voláteis deve ser feita em uma sequência que parta dos dados mais voláteis para os menos voláteis. Os dados mais voláteis tendem a desaparecer mais rapidamente, tendo a preferência na ordem de coleta. Um exemplo de ordem de coleta, partindo dos dados mais voláteis para os menos voláteis, é apresentado a seguir:

- memória RAM;
- registro do Windows;
- estado da rede;
- processos em execução;
- volumes criptografados montados.

As ferramentas preferencialmente utilizadas para a coleta são as de linha de comando, que comprometem menos recursos da máquina alvo, já que a ferramenta utilizada vai ocupar parte da memória RAM do sistema alvo. Existem diversas ferramentas voltadas para resposta a incidentes e segurança de Tecnologia da Informação (TI). Um dos problemas de se utilizar essas ferramentas está no fato de que o usuário tem que lembrar todos os comandos e parâmetros para executar as ferramentas corretamente em linha de comando. Em seguida, o investigador terá que consolidar os resultados de forma a realizar seu relatório. Assim, para utilizar essas ferramentas em todo o seu potencial, é necessário agregá-las em um aplicativo que as execute de forma automática e na ordem correta, atendendo aos princípios forenses relacionados, e salvando os resultados de forma integrada e lógica em um arquivo, para análise posterior.

Existem algumas soluções integradas para o problema da coleta de dados voláteis no mercado. Entretanto, a maioria delas tem fins comerciais e não permite fácil atualização e adequação. As soluções encontradas, em geral, têm foco na segurança da informação e visam uma resposta imediata, enquanto o foco pericial é na preservação e/ou recuperação de dados, senhas e informações relevantes para a análise.

Devido à rápida evolução da computação forense, dos sistemas de informática e dos crimes relacionados, as ferramentas de análise forense não conseguem acompanhar no mesmo passo. Para cada novo sistema operacional, nova versão ou novo programa de roubo de dados desenvolvido, há necessidade de adequação, atualização e adaptação das ferramentas.

Perícia em sistemas ligados exige uma abordagem muito mais complexa e criteriosa do que o exame tradicional, com o sistema desligado. Deve haver extremo cuidado para minimizar o impacto das ferramentas utilizadas. Na avaliação de ferramentas para coleta de dados voláteis, devem ser considerados, entre outros aspectos:

- o total de memória alocada pela ferramenta;
- o impacto da ferramenta nos Registros do Windows;

- o impacto da ferramenta no sistema de arquivos;
- o uso de DLLs presentes no sistema.

É aconselhável que se capture todos os dados voláteis possíveis. A ordem de coleta pode ser crucial para a investigação. O examinador deve avaliar o caso cuidadosamente, para decidir a ordem de coleta, partindo dos dados mais voláteis para os menos voláteis. Os princípios fundamentais que norteiam a extração de dados são os seguintes:

- deve-se coletar todos os dados que serão perdidos ao desligar o sistema;
- deve-se coletar, primeiramente, os dados mais voláteis, deixando os menos voláteis para o final;
- os dados devem ser coletados no menor tempo possível e levando em conta a sua importância;
- os dados coletados devem permanecer disponíveis para futuras análises, se necessárias, e os exames realizados devem ser tão repetíveis quanto possível;
- deve-se manter a integridade dos dados coletados;
- as ferramentas de coleta devem capturar os dados de forma fidedigna;
- as ações realizadas devem ser pertinentes ao caso.

Cabe lembrar que a inserção de qualquer dispositivo em um computador ligado vai produzir pequenas alterações no sistema. O uso apropriado das ferramentas de captura de dados voláteis e a inserção destes dispositivos não adicionará nenhuma evidência ao sistema. Executar uma ferramenta capaz de realizar a coleta de memória RAM, por exemplo, vai necessitar de uma pequena porção da própria memória a ser capturada. Assim, a inserção de um dispositivo USB também vai adicionar uma entrada no registro do Windows. Todas essas pequenas alterações não produzem grandes consequências no sistema como um todo e podem ser explicadas posteriormente, com o exame minucioso e detalhado do material coletado. Essas pequenas alterações são produzidas pela interação das ferramentas com o sistema operacional do Windows, interferindo, apenas, nos arquivos do sistema operacional, não acarretando nenhuma mudança importante no conteúdo dos dados salvos no sistema. Ainda assim, é preciso enfatizar a necessidade de documentar ao máximo todo o processo.

Os dados voláteis incluem qualquer dado armazenado, na memória ou em trânsito, que será perdido com a interrupção da energia ou quando o sistema for desligado. Dados voláteis são encontrados em registradores, memória cache e memória RAM. Dados voláteis passíveis de coleta incluem:

- data e hora do sistema;
- usuários ativos e suas credenciais de autenticação;

- informação sobre processos em execução;
- informações dos registros do Windows;
- dispositivos conectados ao sistema;
- informações do sistema;
- conexões de rede;
- estado da rede;
- conteúdo da área de transferência;
- histórico de comandos;
- arquivos abertos.

Durante a coleta, se constatada a presença de criptografia de volumes, criptografia de sistema ou virtualização do sistema operacional, deve-se realizar a cópia lógica dos arquivos encontrados no volume criptografado ou a cópia de todo o sistema, dependendo do tipo de criptografia utilizada. Se o sistema estiver sendo executado em máquina virtual, também deve ser realizada cópia lógica de todo o sistema. A cópia lógica pode ser realizada com o programa FTK Imager, apresentado na seção seguinte, e o resultado deve ser direcionado a um disco rígido externo conectado a outra porta USB.

Ao final, deve ser verificado se há mais algum dado a copiar, como arquivos lógicos ou memória física. Caso contrário, o dispositivo externo de coleta deve ser retirado e o computador, desligado – retirando-se o cabo da fonte de alimentação. No caso de computadores portáteis, além de retirar o cabo da fonte de alimentação, deve-se retirar a bateria.

É importante ressaltar que o estado de um computador ligado não é estático, pois este encontra-se em funcionamento e em constante alteração de dados tanto de memória quanto de disco e processos. Assim, uma coleta de memória levará a resultados diferentes, a cada vez que for executada. Consequentemente, não há como executar uma função de *hash* de memória, pois os resultados serão sempre diferentes, quando se baseia no tempo. O que se recomenda, e deve ser feito, é um *hash* do arquivo resultante da coleta de memória. O resultado do *hash* desse arquivo deve ser documentado, buscando garantir a cadeia de custódia e evitar que haja questionamentos futuros. Agindo dessa forma, teremos um arquivo contendo a cópia da memória física, com garantia de integridade, possibilitando a repetição dos exames, caso necessário.

6.12.1. Ferramentas para coleta do *dump* de memória

MDD Utilitário desenvolvido pela Mantech International Corporation, para coleta da memória física (RAM) dos computadores ligados. Esse programa é disponibilizado para órgãos governamentais e uso privado, sob licença GPL e é capaz de coletar imagens de memória dos sistemas Windows 2000, Windows Server 2003, Windows XP, Windows Vista e Windows Server 2008 (MDD, 2011). A memória é coletada em formato binário

e a integridade do arquivo resultante é verificada pelo algoritmo MD5. Posteriormente, o arquivo coletado pode ser verificado em laboratório, com programas específicos para este fim como, por exemplo, o Volatility. O programa foi projetado especificamente para coletar uma imagem da memória física de sistema com até 4 GB de memória. O comando para realizar o *dump* da memória e gravá-lo em um arquivo é apresentado a seguir e ilustrado pela Figura 6.5.

```
mdd_1.3 -o arquivo_de_saída.img
```

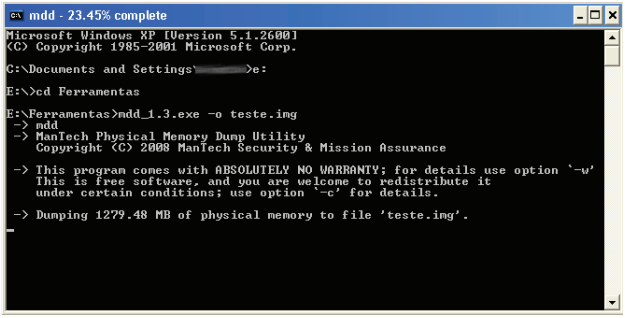

Figura 6.5. Tela de captura de memória do MDD

FTK Imager Utilitário fornecido gratuitamente pela AccessData. É capaz de criar *dumps* de memória em computadores de 32 e 64 bits. A Figura 6.6 apresenta a tela da ferramenta. Embora exista uma versão do FTK Imager para linha de comando, ainda não há suporte para captura de memória.

6.13. Ferramentas para análise indireta

Após realizar a captura da memória, é necessário analizar o conteúdo do *dump*, ou seja, realizar a análise indireta da memória. Duas ferramentas gratuitas para realizar esse procedimento são apresentadas a seguir. Como citado anteriormente, a abordagem de análise indireta da memória via arquivo de *dump* evita a alteração demasiada do estado da máquina, uma vez que substitui vários aplicativos de coleta de dados utilizados na análise direta da memória RAM. Além disso, a análise indireta permite a execução de novas consultas ou confirmações de resultados, o que não é possível utilizando a abordagem direta. Todavia, a análise do *dump* de memória depende fortemente da capacidade dos scripts em reconhecer a organização dos dados na memória RAM, sendo que cada sistema operacional, até mesmo em versões (*services packs*) diferentes, possuem formas distintas de armazenar dados em memória.

MANDIANT Memoryze A ferramenta pode realizar a captura da memória completa do sistema, da memória de um processo, DLL ou *driver* carregado na memória. Com base em um arquivo de *dump* ou a partir do conteúdo da memória em execução, ela pode:

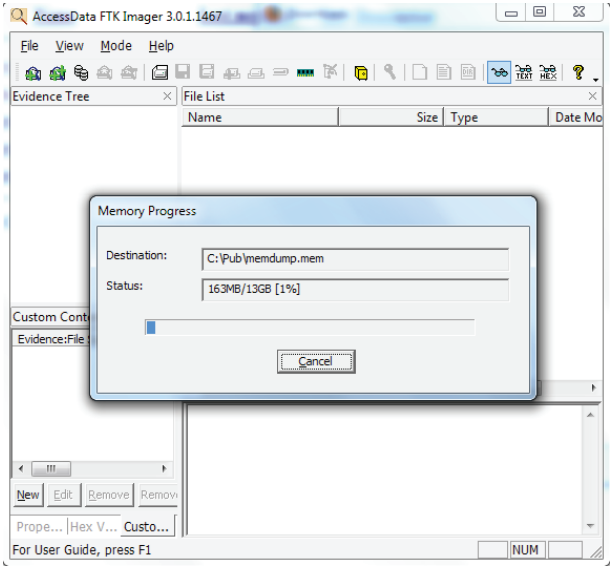

Figura 6.6. Tela de captura de memória do `FTK Imager`

- listar todos os processos em execução, incluindo *rootkits*;
- listar arquivos abertos e chaves de registro em uso;
- especificar funções importadas ou exportadas por executáveis e DLLs;
- listar *strings* em memória para cada processo;
- identificar *hooks*, comumente usados por *rootkits*.

A ferramenta recebe como entrada um arquivo XML com os parâmetros de execução, incluindo a localização do arquivo de *dump*, e gera relatórios em formato XML. Para facilitar seu uso, vários arquivos de lote (`.bat`) são fornecidos. Eles são listados a seguir:

- `MemoryDD.bat`: para obter a imagem da memória física;
- `ProcessDD.bat`: para obter a imagem do espaço de memória de um processo;
- `DriverDD.bat`: para obter a imagem de um *driver*;
- `Process.bat`: para enumerar tudas as informações de um processo incluindo *handles*, memória virtual, portas de rede e *strings*;
- `HookDetection.bat`: para procurar *hooks* no sistema operacional;

- `DriverSearch.bat`: para encontrar *drivers*;
- `DriverWalkList.bat`: para enumerar todos os módulos do *kernel* e *drivers*.

A Figura 6.7 apresenta um exemplo da análise das portas abertas por um processo e o estado da conexão. A saída em XML foi simplificada para facilitar a compreensão.

```
<?xml version="1.0" encoding="UTF-8"?>
<itemList><ProcessItem>
<pid>5400</pid>
<parentpid>5612</parentpid>
<path>Z:\thunderbird</path>
<name>thunderbird.exe</name>
<SecurityID>
S-1-5-21-3064900221-1125138106-471721996-1000
</SecurityID>
<SecurityType>SidTypeUser</SecurityType>
<startTime>2011-09-22T12:22:18Z</startTime>
<PortList>
<PortItem><pid>5400</pid>
    <protocol>TCP</protocol>
    <localPort>64058</localPort>
    <localIP>127.0.0.1</localIP>
    <remotePort>3128</remotePort>
    <remoteIP>10.2.0.50</remoteIP>
    <state>ESTABLISHED</state>
</PortItem>
    <PortItem><pid>5400</pid>
    <protocol>TCP</protocol>
    <localPort>58868</localPort>
    <localIP>127.0.0.1</localIP>
    <remotePort>58869</remotePort>
    <remoteIP>127.0.0.1</remoteIP>
    <state>ESTABLISHED</state>
</PortItem>
</PortList>
</ProcessItem></itemList>
```

Figura 6.7. Análise das portas abertas com a ferramenta `memoryze`

Volatility O *framework* Volatility consiste em um conjunto de *scripts* capazes de extrair informações de execução a partir da análise do arquivo de *dump* de memória do computador investigado. Essa ferramenta percorre o conteúdo do arquivo em busca de assinaturas e estruturas de dados dos elementos contidos em memória, tais como: processos em execução, arquivos abertos, conexões estabelecidas, portas abertas, entre outros. Uma lista completa de parâmetros do Volatility 1.3a é apresentada a seguir:

- `connections`: exibe a lista de conexões estabelecidas com endereço local, endereço remoto e o PID (*Process ID*);
- `connscan`: além da lista de conexões estabelecidas, exibe conexões escondidas e histórico de conexões, caso estejam ainda em memória;
- `connscan2`: exibe as mesmas informações da opção `connscan`;
- `datetime`: exibe hora e data do sistema em que foi feita a cópia da memória RAM;

- `dlllist`: lista as DLLs na memória referenciadas para cada processo;
- `dmp2raw`: converte um arquivo *dump* de falha de sistema em um arquivo de *dump* da memória;
- `dmpchk`: exibe as informações de um arquivo *dump* de falha;
- `files`: lista os arquivos abertos para cada processo;
- `hibinfo`: converte o arquivo de hibernação em arquivo de *dump* da memória;
- `ident`: exibe propriedades do arquivo de *dump* da memória, tais como: nome do arquivo, tipo do sistema operacional e hora de criação do arquivo;
- `memdmp`: extrai a memória endereçada de um processo;
- `memmap`: exibe o mapa de memória;
- `modscan`: busca módulos com os atributos de nome, base e tamanho;
- `modscan2`: exibe as mesmas informações da opção `modscan`;
- `modules`: exibe a lista com os módulos carregados;
- `procdump`: extrai o conteúdo de um processo para um arquivo executável;
- `pslist`: exibe a lista de processos em execução;
- `psscan`: busca por objetos `EPROCESS`;
- `psscan2`: exibe as mesmas informações da opção `psscan`;
- `regobjkeys`: lista as chaves de registro abertas para cada processo;
- `sockets`: lista as *sockets* abertos, contendo os atributos PID, porta, código do protocolo e data de criação;
- `sockscan`: busca por *sockets* abertos;
- `sockscan2`: exibe as mesmas informações da opção `sockscan`;
- `strings`: realiza a correspondência entre os deslocamentos físicos e os endereços virtuais;
- `thrdscan`: busca por objetos `ETHREAD`;
- `thrdscan2`: exibe as mesmas informações da opção `thrdscan`;
- `vaddump`: extrai as seções VAD (*Virtual Address Descriptors*) para um arquivo;
- `vadinfo`: extrai as informações VAD;
- `vadwalk`: percorre a árvore VAD.

A Figura 6.8 apresenta a listagem de processos em execução obtida com a opção `pslist`. É importante notar a presença de dois processos na lista abaixo: `cmd.exe` e `mdd_1.3.exe`, que são produtos do processo de coleta. Além deles, deve-se destacar o uso de criptografia (`truecrypt.exe`) e de máquinas virtuais (`VMWareUser.exe`), que podem ser ameaças ao exame pericial e exigem uma coleta mais cuidadosa.

Name	Pid	PPid	Thds	Hnds	Time
System	4	0	61	261	Thu Jan 01 00:00:00 1970
smss.exe	552	4	3	19	Fri Mar 18 14:23:29 2011
csrss.exe	616	552	12	378	Fri Mar 18 14:23:33 2011
winlogon.exe	640	552	21	644	Fri Mar 18 14:23:34 2011
services.exe	684	640	17	344	Fri Mar 18 14:23:35 2011
lsass.exe	696	640	19	346	Fri Mar 18 14:23:35 2011
vmacthlp.exe	852	684	1	25	Fri Mar 18 14:23:37 2011
svchost.exe	880	684	17	193	Fri Mar 18 14:23:37 2011
svchost.exe	964	684	9	265	Fri Mar 18 14:23:38 2011
svchost.exe	1060	684	71	1414	Fri Mar 18 14:23:39 2011
svchost.exe	1128	684	4	74	Fri Mar 18 14:23:39 2011
svchost.exe	1308	684	15	203	Fri Mar 18 14:23:40 2011
spoolsv.exe	1512	684	12	134	Fri Mar 18 14:23:43 2011
explorer.exe	1612	1596	10	357	Fri Mar 18 14:23:43 2011
VMwareTray.exe	1956	1612	1	76	Fri Mar 18 14:23:46 2011
VMwareUser.exe	1980	1612	6	192	Fri Mar 18 14:23:46 2011
vmtoolsd.exe	1716	684	4	225	Fri Mar 18 14:23:59 2011
VMUpgradeHelper	2036	684	3	98	Fri Mar 18 14:23:56 2011
TPAutoConnSvc.e	324	684	5	99	Fri Mar 18 14:23:58 2011
alg.exe	1172	684	6	108	Fri Mar 18 14:24:00 2011
wscntfy.exe	1456	1060	1	28	Fri Mar 18 14:24:01 2011
TPAutoConnect.e	172	324	1	60	Fri Mar 18 14:24:02 2011
wuauclt.exe	952	1060	3	131	Tue Mar 22 11:45:09 2011
TrueCrypt.exe	216	1612	2	108	Tue Mar 22 15:30:55 2011
cmd.exe	1460	1612	1	32	Tue Mar 22 15:36:17 2011
mdd_1.3.exe	256	1460	1	25	Tue Mar 22 15:36:53 2011

Figura 6.8. Listagem de processos exe execução com o `volatility`

6.14. Ferramentas para análise direta

Esta seção apresenta um conjunto de ferramentas e comandos do sistema operacional para realizar a coleta de dados com o sistema em execução. Essa abordagem se opõe àquela da análise do *dump* de memória. Cabe lembrar que a execução dessas ferramentas, invariavelmente, gerará alterações na memória do sistema e seu uso deve ser documentado detalhadamente.

6.14.1. Comandos do sistema operacional

Os sistemas Windows disponibilizam alguns comandos que podem ser utilizados para diagnosticar e resolver problemas do computador, além de servirem para coletar dados de interesse. Alguns comandos úteis aos procedimentos de coleta de dados são: `cmd`, `time`, `date`, `echo`, `ipconfig`, `netstat` e `tasklist`. Parte desses comandos exige privilégios de administrador do sistema para serem executados. Para a coleta de dados, não é aconselhável que se utilize o `cmd.exe` do sistema investigado, que pode responder de forma imprevisível por estar comprometido por algum *rootkit* instalado na máquina. O kit de coleta de dados utilizado pelo analista deve conter uma cópia do `cmd.exe` sabidamente confiável, para que os comandos ali executados tenham a resposta correta e esperada.

Regedit O editor de registros do Windows possibilita a realização de cópia das chaves da máquina local e do usuário atual com o comando a seguir. O resultado é gravado em

formato texto no arquivo fornecido como parâmetro. A Figura 6.9 apresenta a saída do comando.

```
regedit /e registro.txt
```

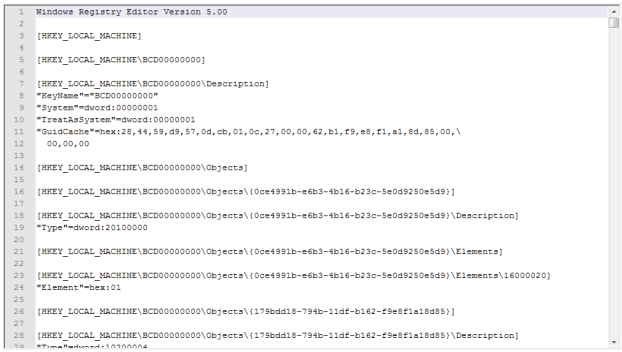

Figura 6.9. Saída do comando `regedit /e`

Outros comandos de sistema do Windows – nome do computador, nome do usuário da sessão, data e hora do sistema e lista de arquivos recentemente abertos pelo usuário da sessão – podem ser executados, conforme especificado a seguir. Todos os resultados são adicionados ao `arquivo_de_saída.txt`.

```
echo %COMPUTERNAME% > arquivo_de_saída.txt
echo %USERNAME% > arquivo_de_saída.txt
dir "%UserProfile%\Recent" > arquivo_de_saída.txt
date /t > arquivo_de_saída.txt
time /t > arquivo_de_saída.txt
```

A lista de processos e serviços em execução pode ser obtida com o comando `tasklist`. A Figura 6.10 apresenta a saída do comando.

```
tasklist /V
```

O comando `netstat` fornece informações sobre conexões de rede, como as portas abertas por processos e as conexões estabelecidas. Um exemplo do comando é exibido na Figura 6.11.

```
netstat -ano
```

Informações sobre as interfaces de rede, incluindo endereços físicos e endereços IP em uso, podem ser obtidas com o uso do comando `ipconfig`. A saída é exibida na Figura 6.12.

```
ipconfig /all
```

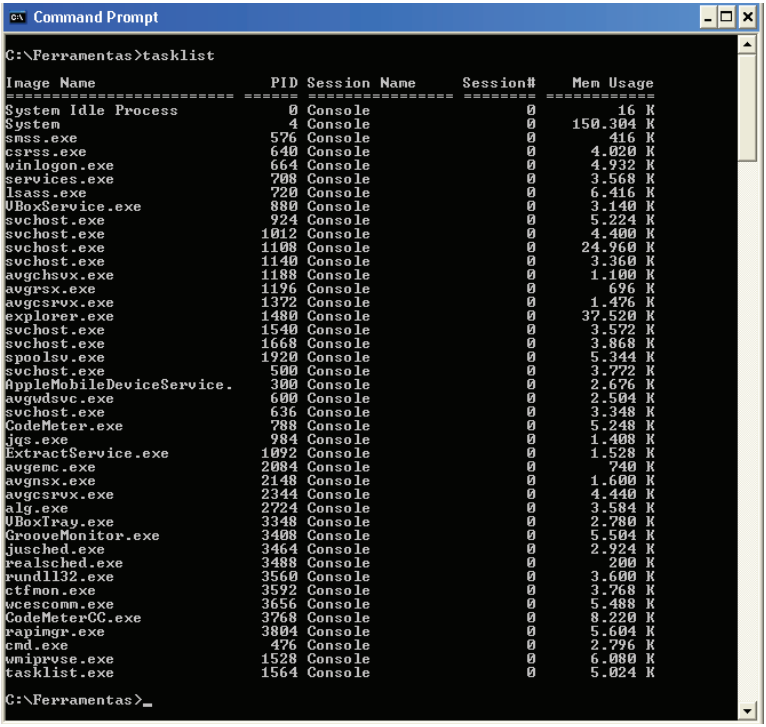

Figura 6.10. Saída do comando `tasklist /V`

O comando a seguir apresenta uma lista dos comandos recentemente utilizados pelo usuário. A Figura 6.13 ilustra o comando.

```
doskey /history > arquivo_de_saída.txt
```

6.14.2. Informações do sistema

Esta seção apresenta utilitários que obtém informações do sistema em execução como arquivos aberto, DLLs carregadas, usuários registrados, data de instalação, dentre outras.

Handle v3.45 Utilitário que mostra a relação de processos com arquivos e pastas abertos. Pode ser executado em sistemas Windows XP ou mais recentes. Necessita ser executado por usuário com poderes de administrador do sistema.

```
handle -accepteula > arquivo_de_saída.txt
```

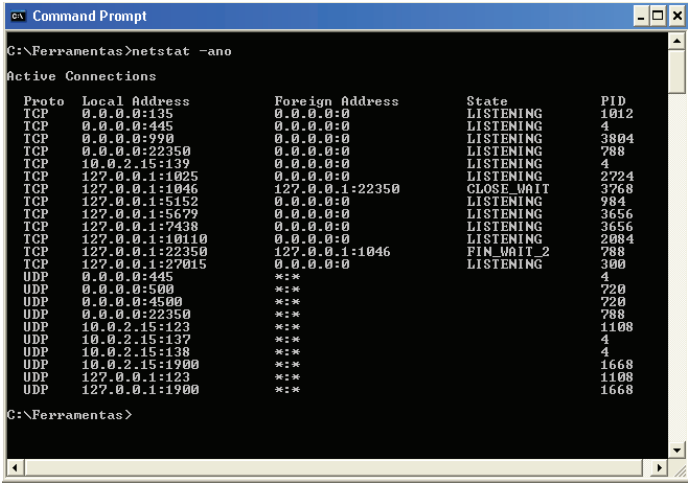

Figura 6.11. Exemplo do comando `netstat`

ListDLLs v3.0 Este utilitário relaciona as DLLs carregadas no sistema. Pode ser executado em sistemas Windows XP ou mais recentes, retornando o nome completo dos módulos carregados. Além disso, sinaliza as DLLs que apresentam número de versão diferente dos seus arquivos correspondentes gravados em disco (isso ocorre quando um arquivo é atualizado depois que o programa carrega suas DLLs), podendo ainda informar quais DLLs foram realocadas.

```
listdlls
```

PsFile v1.02 Este utilitário de linha de comando mostra os arquivos que foram abertos remotamente. Pode ser executado em sistemas Windows XP ou mais recentes.

```
psfile -accepteula
```

PsInfo v1.77 Ferramenta de linha de comando que retorna informações importantes sobre o sistema, incluindo tipo de instalação, usuário registrado, organização, número e tipo de processador, quantidade de memória física, data de instalação do sistema, entre outras. Pode ser executado em sistemas Windows XP ou mais recentes.

```
psinfo -accepteula
```

PsList v1.29 Ferramenta de linha de comando que lista os processos em execução. Pode ser executado em sistemas Windows XP ou mais recentes.

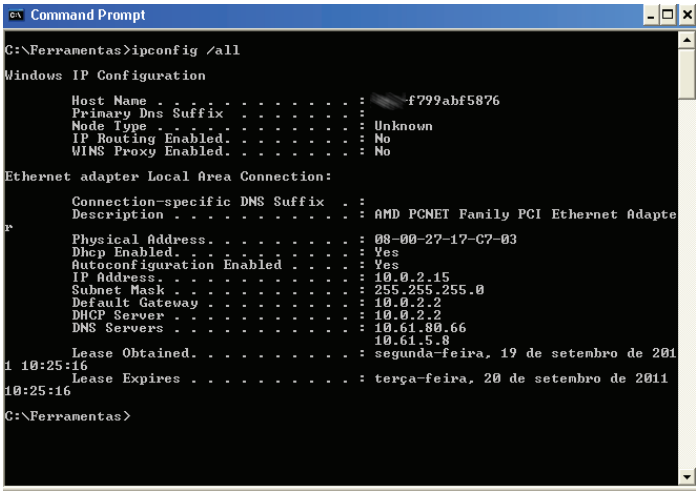

Figura 6.12. Exemplo do comando `ipconfig /all`

```
pslist -accepteula
```

PsLoggedOn v1.34 Este utilitário permite determinar quem está utilizando ativamente o sistema, seja localmente ou remotamente. Pode ser executado em sistemas Windows XP ou mais recentes.

```
psloggedon -accepteula
```

pclip Copia o conteúdo da área de transferência.

```
pclip > arquivo_de_saída.txt
```

6.14.3. Navegadores de Internet

Esta seção apresenta utilitários que extraem informações dos navegadores de Internet como histórico de navegação, senhas gravadas, arquivos temporários do *cache* e buscas realizadas.

IEHistoryView v1.61 Utilitário que permite visualizar as páginas acessadas com o navegador Internet Explorer.

```
iehv /stext arquivo_de_saída.txt
```

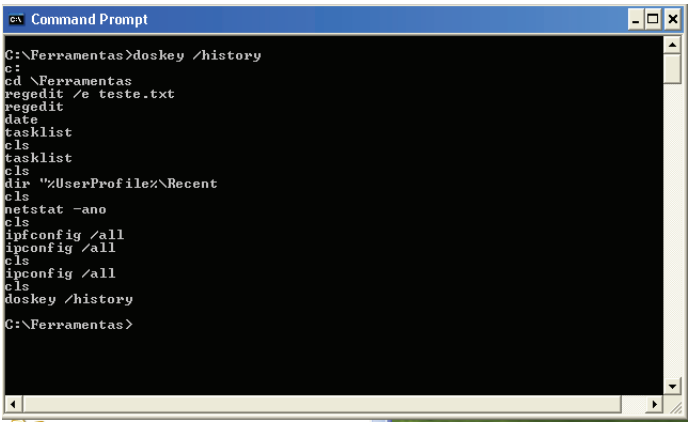

Figura 6.13. Exemplo do comando `doskey /history`

MozillaHistoryView v1.35 Programa utilitário que permite visualizar as páginas acessadas com o navegador Mozilla Firefox.

```
mozillahistoryview /stext arquivo_de_saída.txt
```

ChromeHistoryView v1.00 Utilitário que permite visualizar as páginas acessadas com o navegador Google Chrome. A Figura 6.16 apresenta um exemplo da saída na tela.

```
chromehistoryview /stext arquivo_de_saída.txt
```

IE PassView v1.26 Utilitário que revela as senhas armazenadas pelo navegador Internet Explorer, suportando desde a versão 4.0 até a 9.0.

```
iepv /stext arquivo_de_saída.txt
```

`ChromePass v1.20` Utilitário que revela as senhas armazenadas pelo navegador Google Chrome.

```
ChromePass /stext arquivo_de_saída.txt
```

PasswordFox v1.30 Utilitário que revela as senhas armazenadas pelo navegador Mozilla Firefox, suportando qualquer versão do Windows 2000, XP, Server 2003, Vista, até o Windows 7. Caso uma senha mestre esteja sendo utilizada para proteger as senhas, ela pode ser especificada pelo parâmetro `/master`. Caso ela seja desconhecida, não será possível recuperar as senhas. Ao contrário do Google Chrome, o Firefox permite armazenar senhas incorretas. A Figura 6.17 exibe a tela do aplicativo.

Figura 6.14. Saída do aplicativo `handle`

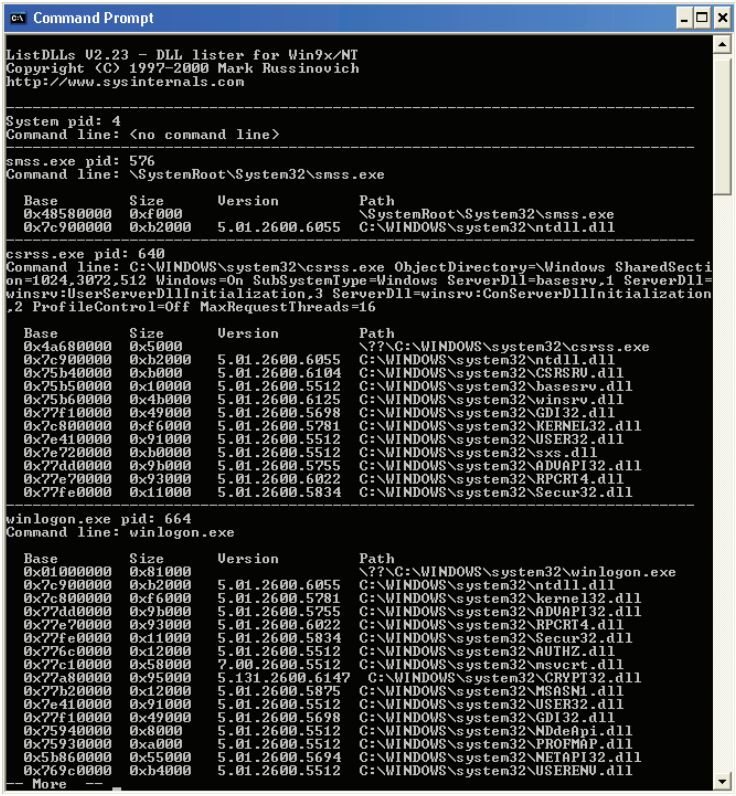

Figura 6.15. Saída do aplicativo `ListDLLs`

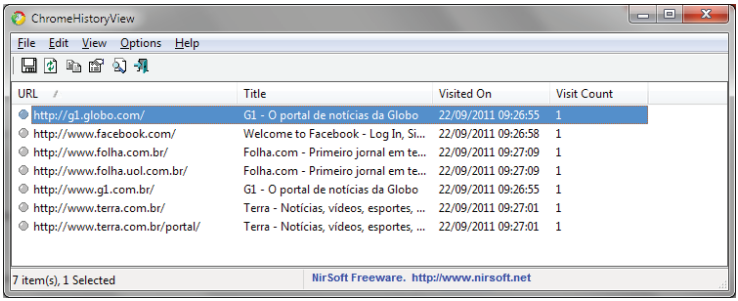

Figura 6.16. Tela do aplicativo `ChromeHistoryView`

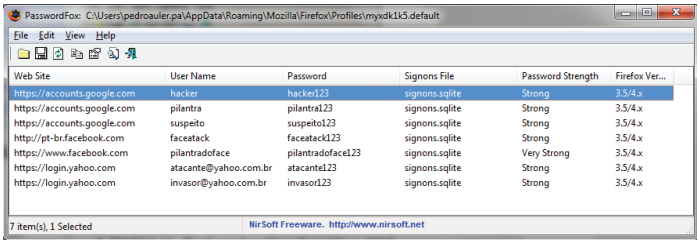

Figura 6.17. Tela do aplicativo `PasswordFox`

```
passwordfox /stext arquivo_de_saída.txt
```

MyLastSearch v1.50 Utilitário que permite para visualizar os últimos termos pesquisados em diversos programas de busca (Google, Yahoo e MSN) e em sites de redes sociais (Twitter, Facebook, MySpace). A Figura 6.18 ilustra os resultados obtidos com esta ferramenta. No caso de buscar realizadas com o recurso de resultados instantâneos, como no navegador Google Chrome, uma busca é registrada para cada letra digitada.

```
mylastsearch /stext arquivo_de_saída.txt
```

6.14.4. Mensageria e comunicação

Utilitários que extraem informações de aplicativos de mensageria e comunicação são apresentados nesta seção, incluindo registros de conversas, ligação utilizando VoIP e senhas gravadas.

SkypeLogView v1.21 Este utilitário acessa os arquivos de log criados pelo Skype e mostra detalhes, como chamadas realizadas ou recebidas, mensagens de chat e transferências de arquivos. A Figura 6.19 ilustra os resultados obtidos com essa ferramenta.

```
skypelogview /stext arquivo_de_saída.txt
```

MessenPass v1.42 Utilitário que permite a recuperação de senhas do usuário atualmente ativo no sistema e somente funciona se o usuário configurar o programa para salvar as senhas utilizadas. Os programas de mensagem instantânea suportados são os seguintes: MSN Messenger, Windows Messenger (em Windows XP), Windows Live Messenger (em Windows XP/Vista/7), Yahoo Messenger (versões 5.x e 6.x), Google Talk, ICQ Lite (versões 4.x/5.x/2003), AOL Instant Messenger (versão 4.6 ou abaixo, AIM 6.x e AIM Pro), Trillian, Trillian Astra, Miranda, GAIM/Pidgin, MySpace IM, PaltalkScene e Digsby.

```
mspass /stext arquivo_de_saída.txt
```

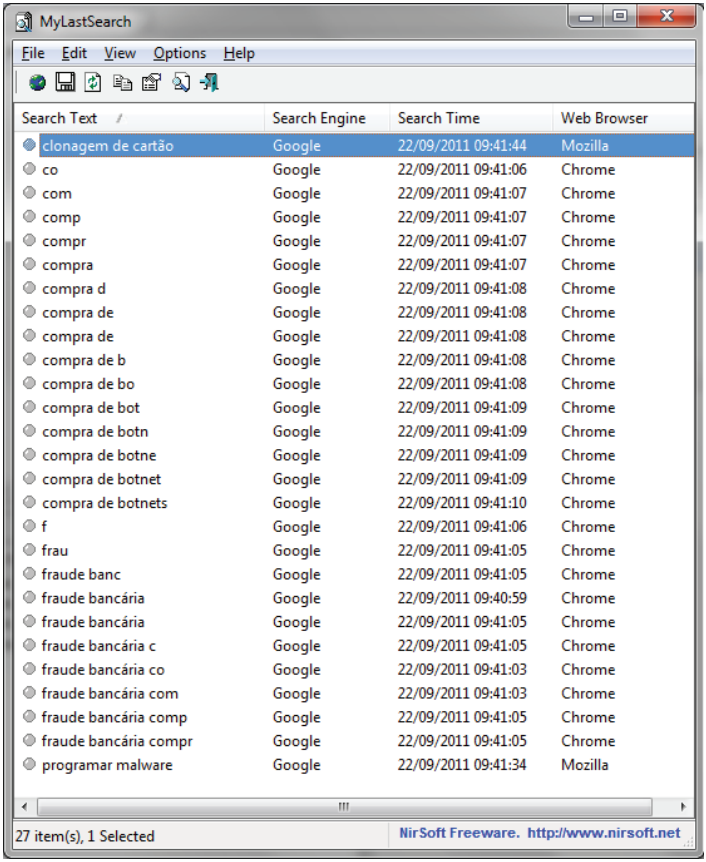

Figura 6.18. Tela do aplicativo `MyLastSearch`

Mail PassView v1.73 Este utilitário pode ser executado em qualquer versão do Windows, desde a versão 98 até o Windows 7, e permite recuperar senhas de programas de e-mail, tais como: Outlook Express, Microsoft Outlook 2000 (POP3 e SMTP), Microsoft Oulook 2002/2003/2007/2010 (POP3, IMAP, HTTP e SMTP), Windows Mail, Windows Live Mail, IncrediMail, Eudora, Netscape 6.x/7.x (se a senha não estiver criptografada com senha mestre), Mozilla Thunderbird (se a senha não estiver criptografada com senha mestre), Group Mail Free, Yahoo! Mail (se a senha estiver salva em alguma aplicação do Yahoo! Messenger) e Gmail (se a senha estiver salva na aplicação Gmail Notifier, Google Desktop ou Google Talk).

```
mailpv /stext arquivo_de_saída.txt
```

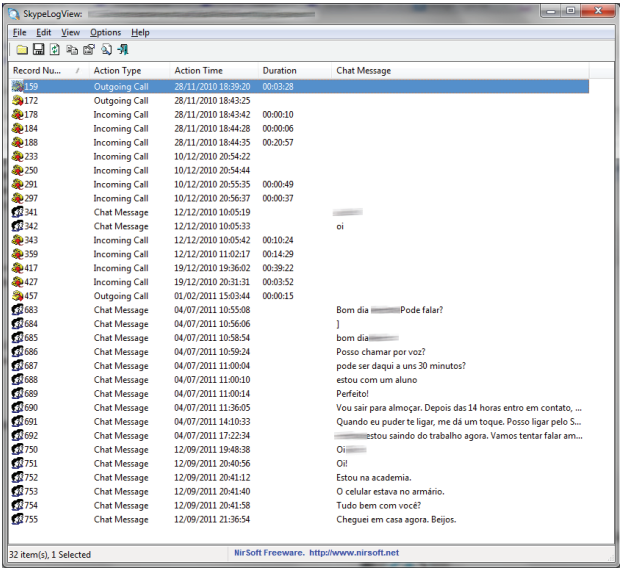

Figura 6.19. Tela do aplicativo `SkypeLogView`

6.14.5. Dispositivos

Informações sobre dispositivos USB, conexões sem fio, volumes criptografados e máquinas virtuais podem ser extraídas com o auxílio dos utilitários apresentados nesta seção.

WirelessKeyView v1.35 Utilitário que recupera as senhas de internet sem fio (WEP/WPA) armazenadas no computador, em sistemas Windows XP e Vista.

```
wirelesskeyview /stext arquivo_de_saída.txt
```

USBDeview v1.89 Utilitário que lista os dispositivos USB conectados ao computador, bem como aqueles que estiveram conectados recentemente.

```
usbdeview /stext arquivo_de_saída.txt
```

Encrypted Disk Detector (EDD) 1.2.0 Ferramenta de linha de comando que verifica a presença de volumes criptografados para os programas TrueCrypt, PGP, Safeboot (McAfee Endpoint Encryption) e Bitlocker. O programa foi testado pelo desenvolvedor em sistemas Windows XP, Vista e 7 (32 e 64 bits), necessitando de apenas 40 KB de espaço em disco e, aproximadamente, 3 MB de memória. Atualmente, o programa é disponi-

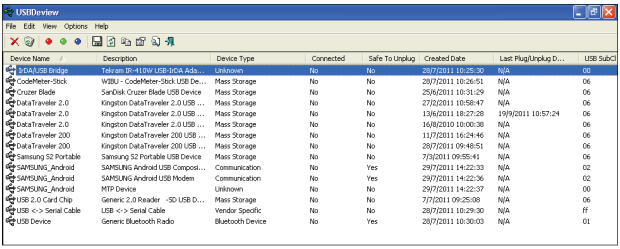

Figura 6.20. Tela do aplicativo `usbdeview`

bilizado gratuitamente. A Figura 6.21 apresenta um exemplo em que não foi detectado qualquer volume criptografado.

```
edd120.exe /batch arquivo_de_saída.txt
```

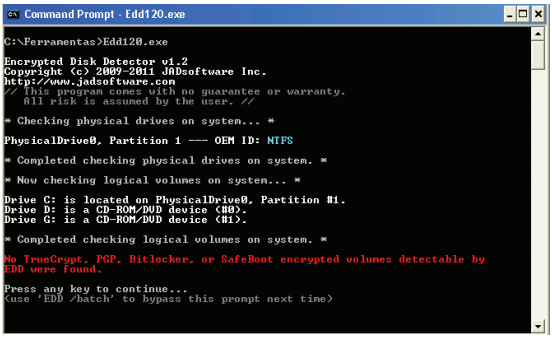

Figura 6.21. Saída do aplicativo `EDD`

ScoopyNG v1.0 Este utilitário combina as técnicas de outras duas ferramentas mais antigas, ScoobyDoo e Jerry, e incorpora algumas novas técnicas para determinar se o sistema operacional corrente está sendo executado dentro de uma máquina virtual VMware ou em um sistema nativo. O aplicativo funciona em qualquer CPU moderna, independentemente do número de processadores utilizados. Além disso, é capaz de detectar a presença do VMware mesmo quando utilizados mecanismos anti-detecção.

```
scoopyng > arquivo_de_saída.txt
```

6.14.6. Ferramentas auxiliares

As ferramentas apresentadas nesta seção auxiliam o trabalho pericial na manipulação dos resultados gerados pelas demais ferramentas.

strings v2.41 Utilitário que permite a extração de caracteres UNICODE (ou ASCII) em arquivos, incluindo arquivos executáveis e DLLs.

```
strings -accepteula arquivo_de_entrada
```

grep Este é um aplicativo de linha de comando proveniente de sistemas Unix/Linux, capaz de fazer buscas no conteúdo de arquivos ou saídas de outros comandos executados, estando também disponível para ambiente Windows. O quadro a seguir apresenta dois exemplos de utilização. No primeiro, é realizada a procura pela palavra "truecrypt"nos processos listados pelo comando `pslist`. A opção `-i` faz com que a diferença entre letras maiúsculas e minúsculas seja ignorada. No segundo, é feita a procura pela palavra "truecrypt"no arquivo `teste.txt`.

```
pslist | grep -i truecrypt
grep -i truecrypt teste.txt
```

fsum v2.52 Utilitário de linha de comando para verificar a integridade de arquivos. Permite a escolha entre várias funções de *hash*. O quadro a seguir apresenta três exemplos. No primeiro, a ferramenta é executada sobre a pasta "Resultados", de forma recursiva (`-r`), e a saída é gravada no arquivo `hashes.txt`. No segundo, a ferramenta calcula o *hash* do arquivo `hashes.txt`, salvando o resultado em `hash_do_hashes.txt`. No terceiro, é pedida a verificação (`-c`) dos *hashes* contidos no arquivo `hashes.txt`, mostrando apenas eventuais falhas de verificação na tela (`-jf`).

```
fsum -d".\Resultados" -r -sha256 * > hashes.txt
fsum -d"." hashes.txt > hash_do_hashes.txt
fsum -jf -c hashes.txt
```

Entre as ferramentas citadas, as distribuídas pela empresa Nirsoft são as seguintes: IEHistoryView v1.60, MozillaHistoryView v1.31, MyLastSearch v1.50, SkypeLogView v1.21, MessenPass v1.41, Mail PassView v1.72, IE PassView v1.26. Todas elas podem ser executadas em linha de comando sem utilização de interface gráfica, podendo o resultado da pesquisa ser direcionado para um arquivo de texto (`/stext arquivo.txt`), para um arquivo HTML (`/shtml arquivo.html`), para um arquivo XML (`/sxml arquivo.xml`) ou outros formatos, que podem ser consultados no site da Nirsoft. Os utilitários são distribuídos gratuitamente, podendo ser utilizados livremente para uso particular ou empresarial, desde que não haja fins lucrativos ou cobranças de qualquer natureza para recuperar senhas de eventuais clientes, a não ser com autorização expressa dos autores do *software*. Os aplicativos podem ser livremente distribuídos, desde que todos os arquivos do pacote sejam incluídos sem qualquer modificação e que não haja nenhum tipo de cobrança financeira.

Algumas ferramentas de linha de comando da Microsoft Sysinternals costumam retornar uma janela perguntando se o usuário aceita as condições da licença de uso. O inconveniente dessa janela é que a coleta automatizada fica interrompida até que o usuário aceite ou não a licença do *software*. Para evitar este inconveniente em uma ferramenta

automatizada de linha de comando, em que os resultados são dirigidos a um arquivo para análise posterior, deve ser utilizada a opção `-accepteula` logo após o comando, aceitando, assim, as condições da licença e evitando o aparecimento da janela.

Algumas vezes, os aplicativos utilizados para fins periciais são classificados incorretamente como vírus ou cavalos-de-troia pelos antivírus eventualmente instalados na máquina em análise. Isso devido ao fato de algumas das ferramentas recuperarem informações sensíveis, como senhas, chaves de instalação e registros do Windows. Logo, esse comportamento é considerado suspeito pelos antivírus. Assim, é recomendável que se desabilite qualquer antivírus instalado antes de iniciar a coleta dos dados, quando possível.

6.15. Considerações finais

A análise de memória RAM tem ser tornado parte importante de qualquer investigação forense computacional, permitindo o acesso aos dados voláteis não encontrados em uma imagem de disco rígido. Apesar dos recentes progressos da análise de memória, as dificuldades ainda são grandes, devido à falta de flexibilidade das ferramentas existentes, que geralmente só podem ser utilizadas nos sistemas operacionais específicos e respectivas versões para os quais foram codificadas.

Como citado anteriormente, a abordagem forense tradicional consiste em retirar o cabo de energia da máquina suspeita, para analisar os dados presentes na mídia de armazenamento posteriormente, em laboratório. Esta técnica pode levar à perda de importantes evidências presentes nos dados voláteis, devido ao crescente uso de criptografia de disco e de sistema. No caso de utilização de criptografia, principalmente de disco, conseguir acesso aos dados com o sistema ainda ligado é de vital importância. Além disso, há uma forte tendência a se utilizar armazenamento remoto de dados, em servidores remotos, por meio de conexões de rede ou Internet. Neste caso, o desligamento precoce do sistema pode inviabilizar a coleta de dados remotos que estão acessíveis somente naquele momento.

Um dos princípios mais importantes da perícia é o *Princípio da Troca de Locard*, que, adaptado à Informática Forense, afirma que ocorrem mudanças em um sistema de informática ativo, simplesmente pela passagem do tempo. Isso ocorre devido aos processos que estão em execução, aos dados que estão sendo gravados na memória ou apagados, às conexões de rede sendo criadas ou finalizadas, e assim por diante. Se as mudanças ocorrem simplesmente pela passagem do tempo, são agravadas quando o investigador executa seus programas de coleta de dados. Afinal, a execução de ferramentas no sistema provoca o seu carregamento na memória RAM, sobrescrevendo outros dados ali presentes.

A coleta de dados em computadores ligados deve ser realizada com impacto mínimo sobre a integridade do sistema. Há pouco tempo, os resultados obtidos dessa forma não eram bem aceitos na Justiça, devido à interferência do analista forense no sistema original. Entretanto, não há mais como fugir dessas técnicas, para que não haja perda definitiva de informações valiosas. A cadeia de custódia deve ser criteriosamente documentada, com a ajuda de *hashes* do material coletado, utilização de fotografias e filmagens e presença de testemunhas durante a coleta. Pequenas interferências no sistema podem e devem ser aceitas, desde que bem documentadas, com o objetivo maior de preservar a informação vital para a investigação e para o processo judicial.

A computação forense é uma área que evolui rapidamente. Como consequência, os crimes de informática também evoluem na mesma proporção. Mais ainda, os sistemas de informática evoluem em velocidade superior à das ferramentas de análise desenvolvidas. Apesar da intensa pesquisa realizada nos últimos anos, a captura e a análise da memória física em sistemas operacionais baseados em Windows ainda está em um estágio inicial de compreensão e desenvolvimento e ainda não existe uma técnica totalmente eficiente. Apesar disso, a análise *live* é capaz de recuperar informações valiosas que, de outra maneira, seriam perdidas se fosse utilizada a técnica de retirar o cabo de energia. É altamente recomendável que a análise tradicional, baseada em disco rígido, seja complementada com a análise *live*, que está se tornando cada vez mais importante e, em alguns casos, determinante, na medida em que as ferramentas de captura e análise se tornam mais sofisticadas e eficientes.

Referências

Adelstein, F. (2006). Diagnosing your system without killing it first. *Communications of the ACM*, 49:63–66.

Anson, S. and Bunting, S. (2007). *Mastering Windows Network Forensics and Investigation*. Sybex.

Aquilina, J. M., Casey, E., and Malin, C. H. (2008). *Malware Forensics - Investigating and Analyzing Malicious Code*. Syngress.

Beebe, N. L. and Clark, J. G. (2005). A hierarchical, objectives-based framework for the digital investigations process. *Digital Investigation*, 2(2):147–167.

Carrier, B. (2006). Risks of live digital forensic analysis. *Communications of the ACM*, 49:56–61.

Carrier, B. D. and Spafford, E. H. (2005). Automated digital evidence target definition using outlier analysis and existing evidence. In *Digital Forensic Research Workshop (DFRWS)*.

FTK Imager (2011). <http://accessdata.com/support/adownloads>. Último acesso em 22/09/2011.

Hay, B., Nance, K., and Bishop, M. (2009). Live analysis: Progress and challenges. digital forensics. *IEEE Security and Privacy*, 7:30–7.

Hoelz, B. W. P. (2009). Madik: Uma abordagem multiagente para o exame pericial de sistemas computacionais. Master's thesis, Universidade de Brasília, Brasília.

Huebner, E., Bem, D., and Bem, O. (2003). Computer Forensics: Past, Present And Future. *Information Security Technical Report*, 8(2):32–36.

Mandia, K., Prosise, C., and Pepe, M. (2003). *Incident Response & Computer Forensics*. McGraw-Hill, 2nd edition.

MANDIANT Memoryze (2011). <http://www.mandiant.com/products/free_software/memoryze>. Último acesso em 22/09/2011.

MDD (2011). <http://sourceforge.net/projects/mdd>. Último acesso em 22/09/2011.

Microsoft Sysinternals (2011). <http://technet.microsoft.com/sysinternals>. Último acesso em 22/09/2011.

NetMarketShare (2011). <http://www.netmarketshare.com/>. Último acesso em 23/09/2011.

Nirsoft (2011). <http://www.nirsoft.com/>. Último acesso em 22/09/2011.

Palmer, G. (2001). A Road Map for Digital Forensic Research. Technical Report DTR - T001-01 FINAL, DFRWS. Report from the First Digital Forensic Research Workshop (DFRWS).

Reith, M., Carr, C., and Gunsch, G. (2002). An Examination of Digital Forensic Models. *International Journal of Digital Evidence*, 1(3).

Rogers, M. K., Mislan, R., Goldman, J., Wedge, T., and Debrota, S. (2006). Computer forensics field triage process model. In *Conference on Digital Forensics, Security and Law*, pages 27–40.

Ruibin, G. and Gaertner, M. (2005). Case-Relevance Information Investigation: Binding Computer Intelligence to the Current Computer Forensic Framework. *International Journal of Digital Evidence*, 4(1).

Turner, P. (2005). Digital provenance - interpretation, verification and corroboration. *Digital Investigation*, 2:45–49.

Volatility (2011). <https://www.volatilesystems.com/default/volatility. Último acesso em 22/09/2011.