UNIVERSIDADE DO SAGRADO CORAÇÃO

NATHALIA SLAGHAUNOFI CINEGAGLIA

IDENTIFICAÇÃO DE FERRAMENTAS PARA QUEBRA DE ESTEGANOGRAFIA NA PERÍCIA FORENSE COMPUTACIONAL: UM ESTUDO DE CASO

BAURU
2010

NATHALIA SLAGHAUNOFI CINEGAGLIA

IDENTIFICAÇÃO DE FERRAMENTAS PARA QUEBRA DE ESTEGANOGRAFIA NA PERÍCIA FORENSE COMPUTACIONAL: UM ESTUDO DE CASO

Trabalho de Conclusão de Curso apresentado ao Centro de Ciências Exatas e Naturais como parte dos requisitos para obtenção do título em Bacharel em Ciência da Computação sob orientação do Prof. Dr. Kelton Augusto Pontara da Costa.

BAURU
2010

NATHALIA SLAGHAUNOFI CINEGAGLIA

IDENTIFICAÇÃO DE FERRAMENTAS PARA QUEBRA DE ESTEGANOGRAFIA NA PERÍCIA FORENSE COMPUTACIONAL: UM ESTUDO DE CASO

Trabalho de Conclusão de Curso apresentado ao centro de Ciências Exatas e Naturais como parte dos requisitos para obtenção do título em Bacharel em Ciência da Computação sob orientação do Prof. Dr. Kelton Augusto Pontara da Costa.

BANCA EXAMINADORA:

_____
Prof. Dr. Kelton Augusto Pontara da Costa
Orientador

_____
Profa. Ms. Larissa Pavarini da Luz
Examinadora

_____
Prof. Esp. Henrique Pachioni Martins
Examinador

DATA:

Dedico este trabalho a todos os que me ajudaram, me incentivaram e me fizeram acreditar que seria capaz de concluí-lo.

AGRADECIMENTOS

Agradeço primeiramente a Deus, por estar em todos os momentos junto à mim.

Ao Prof. Dr. Kelton Augusto Pontara da Costa pela disponibilidade e dedicação na orientação deste trabalho.

Ao Prof. Esp. Henrique Pachioni Martins e à Profa. Ms. Larissa Pavarini da Luz, que na apresentação do projeto de pesquisa, fizeram correções e apresentaram sugestões.

A minha mãe Catarina, irmã Francielle e namorado Richard pela compreensão e incentivo.

Aos colegas de curso, pelo companheirismo.

A todos aqueles que, direta ou indiretamente, possibilitaram a realização desse trabalho.

RESUMO

A Perícia Forense Computacional tem como objetivo investigar computadores e/ou dispositivos eletrônicos envolvidos em atividades criminosas, contendo informações explícitas ou não. Algumas técnicas, que normalmente seriam utilizadas como mecanismos de segurança da informação para impossibilitar a invasão de hackers, podem ser aplicadas de maneira contrária, escondendo informações importantes. Assim surge a necessidade de estudar uma dessas técnicas, a esteganografia, que nesse contexto é conhecida como técnica anti-forense. A mesma consiste em esconder informações através de arquivos inofensivos, ou seja, com a utilização de uma imagem de capa. O estudo dessa técnica é conhecido como esteganálise, que norteia essa pesquisa, através da avaliação de técnicas e ferramentas já existentes que possibilitem a realização, identificação e quebra da esteganografia, identificando as técnicas utilizadas em cada ferramenta. Dessa forma os resultados obtidos com esse estudo auxiliam a perícia, arrecadando mais informações que devem compor mais satisfatoriamente o resultado da investigação.

Palavras-chave: Perícia Forense Computacional. Esteganografia. Segurança da Informação. Hacker. Anti-forense. Esteganálise.

ABSTRACT

The Computer Forensics aims to investigate computers and / or electronic devices involved in criminal activity, containing explicit information or not. Some techniques, which would normally be used as mechanisms for information security to preclude hacking, can be applied in reverse, hiding important information. Thus arises the need to study one of these techniques, steganography, which in this context is known as anti-forensic technique. This technique consists in hiding information through harmless files, more precisely, with the use of a cover image. The study of this technique is known as steganalysis, which guides this research, through evaluation of existing tools and techniques that make possible the identification, implementation and disruption of steganography, detailing the techniques used in each tool. Thereby the results obtained from this study assists the forensics, by collecting more information to make the most satisfactory outcome of the investigation.

Keywords: Computer Forensics. Steganography. Information Security. Hacker. Anti-forensics. Steganalysis.

LISTA DE ILUSTRAÇÕES

LISTA DE QUADROS

LISTA DE ABREVIATURAS E SIGLAS

ABNT	Associação Brasileira de Normas Técnicas
ANSI	*American National Standards Institute*
ASCII	*American Standard Code for Information Interchange*
BMP	*Bitmap*
DC	*Direct Current*
DCT	*Discrete Cosine Transform*
DPI	*Dots Per Inch*
DMZ	*Demilitarized Zone*
DoS	*Denial of Services Attacks*
FTK	*Forensic Tool Kit*
FTP	*File Transfer Protocol*
GIF	*Graphic Interchange Format*
HTML	*HyperText Markup Language*
ICMP	*Internet Control Message Protocol*
JPEG	*Joint Photografic Experts Group*
LSB	*Least Significant Bit*
MP3	*MP3 Audio File*
OOV	*Order of Volatility*
PDF	*Portable Document Format*
POP	Procedimentos Operacionais Padrão
PRTK	*Password Recovery Tool Kit*
RGB	*Red Green Blue*
SSH	*Secure Shell*
DCT	*Discrete cosine transform*
TCP/IP	*Transmission Control Protocol/Internet Protocol*
UDP	*User Datagram Protocol*
VOC	*Creative Labs Audio File*
WAV	*Wave Audio File*

SUMÁRIO

1 INTRODUÇÃO

A perícia forense computacional é uma área de estudo bastante recente, e que hoje é muito utilizada em investigações policiais, a fim de obterem provas e/ou evidências, que através de meios legais se tornem confiáveis e passíveis de aceitação perante uma investigação judicial. (FREITAS, 2006, p.2)

Desta forma, para que se possa obter um resultado eficaz em uma perícia forense computacional, é necessário ter conhecimento de técnicas que identifiquem dentro de um computador ou qualquer dispositivo eletrônico, a presença de arquivos que possam conter algo sigiloso ou algum indício que possam ajudar nas investigações forenses. A partir desta metodologia de filtragem de arquivos relevantes, o perito forense deve estar atento a uma categoria muito importante de arquivos, os esteganografados, que podem conter informações extremamente válidas.

Assim através do conhecimento da segurança da informação e da esteganografia, é possível iniciar do princípio básico da perícia forense computacional, que trata de procurar informações em todos os lugares, os óbvios e não óbvios, e também não ter preconceitos quanto às mesmas, ou seja, não ignorar certas informações ou arquivos por parecerem irrelevantes. (FARMER e VENEMA, 2007, p. 4)

Quanto à segurança da informação, o armazenamento seguro de todo e qualquer tipo de dispositivos eletrônicos, computadores e relatórios impressos, que contenham informações importantes, é indispensável para garantir que pessoas não autorizadas tenham acesso a esses meios físicos. (KATZAN, 1977, p. 4)

Para Tanenbaum (2003, p. 767), além de cuidados com o meio físico, é preciso ter cuidados com o meio lógico em relação a segurança lógica das informações. A mesma consiste em construir obstáculos para dificultar o acesso de estranhos. Para tanto é preciso conhecer o “inimigo” para saber como ele pensa e quais as possíveis maneiras que este pode utilizar para invadir. Exemplifica também alguns perfis dos invasores e quais são os seus objetivos, com o propósito de proteger a informação sob todos os aspectos que a tornem vulnerável.

Quanto à perícia forense computacional, esse trabalho pretende demonstrar os conceitos principais e identificar suas funcionalidades para a arrecadação de evidências e provas afim de que as mesmas possam fazer parte de uma investigação e ajudar a solucionar um caso.

Segundo Freitas (2006, p. 1) perícia vem do latim *peritia*, que pode ser definida como “exame de caráter técnico, vistoria”; e forense vem do latim *forense* “que se refere ao foro judicial”. Resumidamente, o autor define perícia forense como: vistoria e análise de informações contidas no meio em que ocorreu um crime, com o objetivo de relatar os resultados num laudo técnico a ser utilizado juridicamente.

Farmer e Venema (2007, p.4) explicam que para solucionar um enigma durante a perícia forense computacional, é necessário buscar informações em todos os lugares, sem preconceitos e sem buscar algo já esperado, ou seja, é preciso procurar quaisquer evidências e em qualquer lugar.

Para Freitas (2006, p. 2) a perícia forense computacional deve ser realizada seguindo preferencialmente quatro etapas, ou a maioria delas: “[...] todas as evidências devem ser identificadas, preservadas, analisadas e apresentadas.”

De acordo com Farmer e Venema (2007, p.5) uma tarefa muito importante a ser realizada no início da perícia forense é a cópia dos dados, antes de começar a processar as informações, assim é possível manter uma maior segurança, para o caso de haver qualquer problema que possa danificar ou até mesmo perder algumas informações. Os autores fazem analogia a ação de filmar a cena de um crime nas perícias comuns, com o objetivo de manter uma “versão intacta”, para não correr o risco de perder quaisquer indícios.

Conhecendo a perícia forense computacional, esse trabalho pretende estudar a técnica que permite esconder informações, visando descobrir, durante a perícia, se existem informações importantes por trás de arquivos aparentemente irrelevantes. Essa técnica é conhecida como esteganografia.

> [...] as pessoas que desejam se comunicar secretamente tentem a ocultar o fato de haver qualquer comunicação. A ciência de ocultar mensagens é chamada de esteganografia, das palavras gregas que correspondem a “escrita cifrada”. (TANENBAUM, 2003, p. 876).

Partindo desse raciocínio, a esteganografia, tem como objetivo possibilitar que mensagens sejam transmitidas com segurança e identificadas apenas pelo

destinatário correto. Sabe-se que antigamente os gregos já se utilizavam desta técnica: Heródoto escreveu uma mensagem secreta no couro cabeludo de um general antes de mandá-lo ao destino. Isso é uma situação que demonstra os primórdios do que é conhecido atualmente como esteganografia, ou seja, os “fins” são os mesmos (fazer com que a mensagem seja vista apenas pelo destino indicado, sem despertar interesse), os “meios” é que evoluíram. (TANENBAUM, 2003, p.876)

Segundo Deitel (2002, p. 611) “A esteganografia é a prática de esconder informações dentro de outras informações. O termo significa literalmente, ‘escrita encoberta’”. Desta forma, esteganografia é a maneira de guardar informações através outras informações, uma forma de mascarar.

Johnson e Jajodia (1998) citam algumas técnicas que realizam a esteganografia: camuflagem, tintas invisíveis e arranjo de caracteres. Explicam também que esteganografia e criptografia são conceitos próximos, mas com diferenças relevantes: a criptografia é a técnica que codifica a mensagem e a esteganografia camufla a mensagem dando-lhe uma cobertura ou capa disfarçando seu conteúdo.

Enfim, esse trabalho pretende demonstrar alguns conceitos de segurança da informação, e como identificar e realizar algumas técnicas para assegurá-la; dentre essas técnicas, a que será apresentada com maior ênfase é a esteganografia, com o objetivo de identificar procedimentos que possibilitem a quebra dessas barreiras que possam impedir um resultado eficaz na arrecadação de evidências. Para isso será realizado um estudo de caso das principais ferramentas para esse fim, demonstrando suas vantagens e desvantagens.

1.1 OBJETIVO GERAL

Este estudo tem o objetivo de demonstrar ferramentas que auxiliem na busca, identificação de arquivos que contenham informações escondidas, ou arquivos esteganografados. Com o propósito de desmascarar essas informações, ou seja, quebrar a esteganografia, para que mais evidências possam ser agregadas à perícia forense computacional.

1.2 OBJETIVOS ESPECÍFICOS

Os objetivos específicos desta pesquisa são:

- Identificar métodos para garantir a segurança da informação, e também verificar sua importância.
- Conceituar a Perícia Forense Computacional, identificar sua importância, seus métodos e algumas técnicas anti-forense.
- Demonstrar o funcionamento da esteganografia e como ela pode auxiliar/influenciar a perícia através do estudo das ferramentas da área.
- Identificar procedimentos que possibilitem a quebra da esteganografia.

1.3 JUSTIFICATIVA

Com a grande facilidade de enviar e receber informações sigilosas através da internet, proporcionalmente cresce também os crimes digitais, pois essas informações são alvos que atraem muitos interesses, e que podem variar desde o simples acesso a um e-mail até o roubo de quantias significativas das contas bancárias das vítimas.

Além do roubo de informações e danos financeiros causados, os invasores ainda causam danos às máquinas; de alguma forma o criminoso pode invadir um computador que aparentemente está seguro, a fim de desfrutar de informações, causar prejuízo ou até mesmo proporcionar satisfação pessoal.

Quanto às fraudes bancárias, a facilidade do uso da página do banco que atualmente possibilita a realização de diversos serviços, que antes eram realizados apenas nas agências bancárias, torna os mesmos mais suscetíveis e vulneráveis a acessos indevidos e fraudes. Desta forma, vem crescendo a cada dia as estatísticas de fraudes realizadas através desses serviços.

Assim surge o interesse em estudar a perícia forense computacional, que trata da investigação de crimes computacionais. Percorrendo todos os dispositivos eletrônicos que podem conter evidências de um crime.

Tendo em vista que essas informações não estão totalmente disponíveis, é indispensável ao perito forense ter conhecimento de algumas técnicas, como a esteganografia, que podem ser aplicadas aos arquivos, escondendo seu conteúdo real.

Aplicando o conhecimento da esteganografia e das ferramentas que possibilitem a identificação de arquivos esteganografados, o perito pode arrecadar maiores informações e provavelmente as mais importantes da perícia, obtendo assim um resultado mais eficaz.

1.4 ESTRUTURA DO TRABALHO

Este trabalho segue a metodologia proposta pela Universidade do Sagrado Coração-USC Bauru, baseada na ABNT, com a seguinte estruturação:

Nesta primeira parte, está descrita a introdução dos principais aspectos relacionados a essa pesquisa sobre perícia forense computacional; juntamente com a justificativa/motivação e por fim os objetivos, geral e os específicos, pretendidos para essa pesquisa.

Na segunda parte, está o referencial teórico que foi subdividido em capítulos de acordo com o assunto abordado.

O primeiro capítulo do referencial teórico, explica quais os problemas de segurança da informação na computação e também as vertentes que asseguram a mesma. Explica como as informações passaram a ser mais vulneráveis e quais são as causas para que isso ocorra. E apresenta também, algumas técnicas para defesa da invasão dos hackers.

O segundo capítulo faz um estudo sobre a perícia forense computacional, apresentados seus conceitos, demonstrando os procedimentos necessários para realizar a perícia forense. Apresenta ainda, técnicas anti-forense, explicando como funcionam algumas delas e como o perito pode identificar se foram utilizadas.

O terceiro capítulo aborda a técnica da esteganografia, seus conceitos, demonstrando alguns exemplos de utilização. Traz também algumas ferramentas que aplicam essa técnica. A esteganografia será utilizada posteriormente nos estudos de caso que concluirão esta pesquisa.

Na terceira parte, são apresentados os materiais e métodos utilizados para a realização do trabalho, na questão experimental, que trata de estudos de caso, que apresentaram as ferramentas para a identificação e a quebra da esteganografia durante a perícia forense computacional. Consta também o cronograma completo das atividades realizadas durante o trabalho.

A quarta parte apresenta os resultados, obtidos até o momento da primeira entrega do trabalho.

E por fim, as referências utilizadas, durante todo o desenvolvimento do trabalho, e as bibliografias utilizadas apenas como consulta.

2 REFERENCIAL TEÓRICO

2.1 SEGURANÇA DA INFORMAÇÃO

A segurança da informação é um assunto muito importante e imprescindível atualmente. Para ser garantida, a segurança da informação tem várias vertentes a serem verificadas e que serão descritas a seguir; dentre elas a integridade, a autenticidade, o não-repúdio, etc.

Segundo Katzan (1977, p. 4), a segurança da informação tem como objetivo, limitar a disponibilidade da informação apenas a pessoas autorizadas. Desta forma o autor dá ênfase aos cuidados necessários que se deve ter com o meio físico em que o computador ou dispositivo eletrônico se encontra, para que pessoas não autorizadas sejam impedidas de ter acesso ao local (físico) do computador, dispositivos removíveis, ou até mesmo relatórios impressos, em que o autor dá um exemplo simples de como devem ser descartados corretamente, utilizando máquinas de picar papel.

Assim, a segurança da informação deve ser avaliada sob vários aspectos, um deles é o lógico, que deve proteger a informação de corrupções e perdas. Outro cuidado é com o local físico em que as informações estão presentes, cuidando para que não sejam depredadas e não estejam ao alcance de pessoas não autorizadas.

“O objetivo da segurança de dados abrange desde uma fechadura na porta da sala de computadores até o uso de técnicas criptográficas sofisticadas e códigos de autorização.” (KATZAN, 1977, p. 4)

Para Forouzan (2004, p. 711) “[...] a segurança deve garantir quatro serviços: privacidade (confidencialidade), autenticação, integridade e o não-repúdio (a rejeição)”, onde o autor define:

- Privacidade: serviço de garantia que a informação conta com a confidencialidade, ou seja, a informação deve chegar limpa, ser legível ou fazer sentido apenas para o destinatário correto;
- Autenticação: serviço que garante que ao destinatário a certeza da identidade do remetente, atestando que a informação não foi remetida por um intruso tentando se passar pelo remetente correto, ou seja, deve conter algo que prove sua origem, como por exemplo, a assinatura digital;

- Integridade: serviço de garantia de que a informação chegará ao destino, completamente inalterada, ou seja, da mesma forma que o remetente transmitiu;
- Não-repúdio: serviço que garante a veracidade da origem da informação, sendo que esta deve ser feita pelo destinatário, ou seja, o destinatário deve ter como provar que a informação é realmente vinda de uma determinada origem, como no exemplo: “[...] quando um usuário envia uma mensagem para transferir dinheiro de uma conta para outra, o banco deve provar que o usuário realmente requisitou uma transação financeira” (FOROUZAN, 2004).

Para Lyra (2008, p. 4) a segurança da informação tem mais algumas vertentes, como:

- Disponibilidade: serviço que garante que a informação ficará disponível a todos que necessitarem e que forem autorizados a acessar a mesma;
- Legalidade: serviço que garante que o sistema está de acordo com os parâmetros das leis pertinentes;
- Auditoria: serviço que garante a possibilidade de auditar tudo que foi realizado pelo usuário, a fim de detectar fraudes ou até mesmo um ataque ao sistema.

Lyra (2008, p.4), explica que a privacidade também deve garantir o anonimato do remetente em casos específicos, como é o caso do sistema de voto eletrônico.

Exemplificando, Kurose e Ross (2006, p. 513) expõem o princípio básico para se chegar a essas propriedades, partindo da seguinte situação em que duas pessoas (A e B) desejam trocar informações, de “A” para “B”: “A” deseja que apenas “B” entenda a informação transmitida (privacidade); “B” precisa ter certeza que a mensagem recebida foi realmente enviada por “A” e não por um intruso querendo se passar por “A” (autenticação); “A” e “B” querem ter certeza de que a informação não se corrompeu e que não foi alterada (integridade); “B” deve ter como atestar que a mensagem foi remetida por “A”, sem que “A” possa negar o envio seu envio (não-repúdio); e por fim é muito importante que o sistema garanta os recursos necessários para a comunicação entre “A” e “B” (disponibilidade). (KUROSE E ROSS, 2006, p. 513).

Existem também os aspectos humanos e físicos que influenciam na segurança da informação, como pessoas que pretendem prejudicar uma organização ou outra pessoa, e para isso fazem uso de técnicas para obter informações sigilosas.

Segundo Lyra (2008, p.20), “a preocupação com a segurança da informação deve ser de todos os colaboradores da organização, e não apenas de um grupo de pessoas.”, portanto é preciso que todos os colaboradores tenham tal conhecimento. Para isso surgiu dentro da organização o “profissional da segurança” que é responsável por planejar e programar as metas de segurança aos colaboradores, bem como, monitorar, melhorar o sistema de segurança, investigar os incidentes e analisar os riscos envolvendo a segurança.

Porém, mesmo com os esses cuidados em relação aos colaboradores da organização também é preciso conhecer e identificar pessoas que não fazem parte da mesma, mas que podem comprometer a segurança e integridade das informações.

Tanenbaum (2003, p. 767), explica que é imprescindível estar atento a pessoas que possam de alguma forma, prejudicar a organização: “[...] a segurança se preocupa em garantir que pessoas mal-intencionadas não leiam ou, pior ainda, modifiquem secretamente mensagens enviadas a outros destinatários”. O Quadro 1, classifica tais pessoas de acordo com seus objetivos:

Adversário	**Objetivo**
Estudante	Divertir-se bisbilhotando as mensagens de correio eletrônico de outras pessoas
Cracker	Testar o sistema de segurança de alguém; roubar dados
Representante de vendas	Tentar representar toda a Europa e não apenas Andorra
Executivo	Descobrir a estratégia de marketing do concorrente
Ex-funcionário	Vingar-se por ter sido demitido
Contador	Desviar dinheiro de uma empresa
Corretor de valores	Negar uma promessa feita a um cliente através de uma mensagem de correio eletrônico
Vigarista	Roubar números de cartão de crédito e vendê-los
Espião	Descobrir segredos militares ou industriais de um inimigo
Terrorista	Roubar segredos de armas bacteriológicas

Quadro 1- Algumas pessoas que podem causar problemas de segurança e os motivos para fazê-lo.
Fonte: Tanenbaum, 2003, p. 768.

Com essas classificações é possível identificar o que tais pessoas pretendem e qual o nível de segurança necessário para cada tipo de invasor, com o objetivo de prevenir o acesso não autorizado às informações da organização.

Segundo Lyra (2008, p. 21), uma das metodologias que essas pessoas mal-intencionadas utilizam para atingir seus objetivos é a “engenharia social”, que são técnicas para obter informações importantes através de persuasão de pessoas ingênuas, mal informadas e prestativas que acreditam estar ajudando, mas que na realidade estão contribuindo para prejudicar alguém ou alguma organização. Porém essas técnicas, não se limitam apenas a relação com pessoas pode envolver outro tipo de busca, como por exemplo, colher informações no lixo, escuta de conversa telefônica e buscar papéis e relatórios sobre as mesas da organização.

Portanto, a engenharia social trata da utilização das habilidades do invasor em captar informações da vítima ou de alguma situação em que se encontra, sem que seja nota ou identificada.

2.1.1 Vulnerabilidades

Atualmente os dados de uma empresa informatizada, são geralmente armazenados em dispositivos eletrônicos, tornando a utilização do papel menos corriqueira. Uma das vantagens desse avanço é o acesso rápido a milhares de informações, com ganho de desempenho nas atividades da organização.

Essa realidade influenciou significativamente na vulnerabilidade dos dados informatizados, pois passaram a ser “mais suscetíveis a destruição, fraude, erro e uso indevido.” (LAUDON E LAUDON, 2004, p. 460)

Caruso e Steffen (1999, p. 177), destacam também que no princípio da informatização, as redes eram limitadas apenas a própria organização (uso interno), ou seja, apenas os colaboradores (funcionários) da empresa tinham acesso às informações, mas com o decorrer do tempo, as empresas agregaram também seus fornecedores e clientes, como usuários, que mesmo com acesso limitando, podem aumentar a vulnerabilidade a ataques externos.

Lyra (2008, p. 6), explica o que as informações têm propriedades importantes chamadas de ativo da informação: “é composto pela informação e tudo aquilo que a

suporta ou se utiliza dela.", ou seja, tudo que está agregado quando se fala de informação, a tecnologia em que ela está inserida, onde ela está localizada, o que a mantém, e as pessoas que a manipulam. Assim o autor afirma que o ativo da informação possui "vulnerabilidades ou fraquezas que podem gerar, intencionalmente ou não, a indisponibilidade, a quebra de confidencialidade ou integridade.", ou seja, podem causar falhas e ameaças a qualidade da informação.

Dessa forma a qualidade dos sistemas de informação influencia na vulnerabilidade dos dados, pois são eles que executam as funções que tornam os dados disponíveis, e que permitem sua manipulação.

Para Laudon e Laudon (2004, p.460), quando os sistemas de informação não funcionam em sua totalidade, ou não executam tarefas como esperado, podem causar sérios danos a organização, no caso de empresas que dependem da internet como ferramenta de venda ou especulação, estando fora de operação, causam danos a empresa a cada hora inoperante: "Um site de corretagem on-line, por exemplo, pode perder 5 milhões de dólares a cada oito horas por dia útil em que estiver fora de operação."

Segundo Laudon e Laudon (2004, p. 461) as principais causas que podem ameaçar sistemas de informação, tornando os dados vulneráveis são: falha de hardware ou de software; ações pessoais; invasão pelo terminal de acesso; roubo de dados, serviços, equipamentos; incêndio; problemas elétricos; erros de usuários; mudanças no programa e problemas de telecomunicação.

Freitas (2006, p. 126), exemplifica uma situação de vulnerabilidade da seguinte forma: se um invasor tem como alvo o site de uma empresa, para roubo de informações, o primeiro é a coleta de informações, e para saber quais informações estão disponíveis, o invasor pode utilizar um espelhamento de site, assim "o invasor começa a 'varredura de vulnerabilidade'. O invasor procura pela existência de páginas Web com vulnerabilidades conhecidas.", ou seja, páginas que tenham falhas na segurança.

Para Laudon e Laudon (2004, p.461), as ameaças podem ser originárias de diversos fatores, dentre eles, os técnicos, os organizacionais e os ambientais; e alerta também para o fato de que a partir de um desses fatores a situação pode se agravar significativamente, caso seja tomada alguma decisão errada. Os autores

explicam também que as redes de telecomunicações são muito mais suscetíveis a falhas, veja na Figura 1, alguns exemplos que ameaçam essas redes. As redes sem fio, baseadas em tecnologia rádio, por exemplo, podem ser facilmente invadidas, pois é fácil realizar a “varredura das faixas de radiofreqüência”. A internet foi projetada para permitir o acesso a pessoas com sistemas de informação diferentes, e por isso também apresenta certas vulnerabilidades.

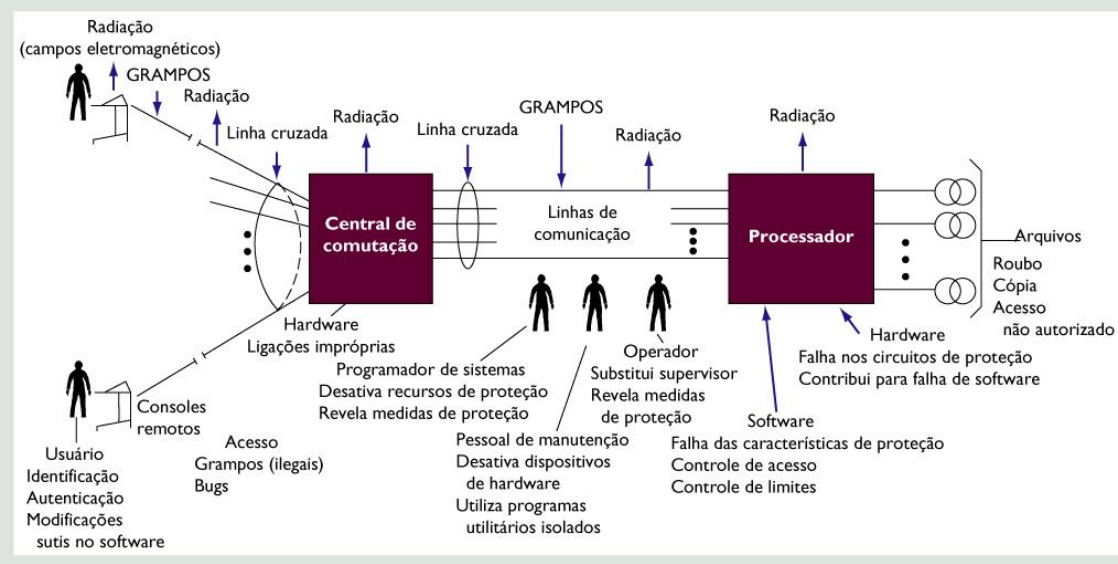

Figura 1- Vulnerabilidades da rede de telecomunicação.
Fonte: Laudon e Laudon, 2004, p. 461.

Nessa representação é possível verificar os diferentes fatores que afetam as redes de telecomunicação, sendo que eles podem ser causados por usuários e programadores mal-intencionados que podem desativar alguma proteção para facilitar o acesso indevido; ou por instalações incorretas dos hardwares e problemas no meio físico dessas redes que podem prejudicar o desempenho dos mesmos ocasionando falhas na proteção e até perda dos dados. (LAUDON E LAUDON, 2004, p. 461).

Complementando, Parodi (2008) explica que qualquer pessoa que tenha um computador, pode estar vulnerável a um ataque de golpistas digitais ou hackers, que atualmente não procuram apenas vitimar banqueiros, mas também as pessoas comuns, pois parecem estar menos conscientes e despreparadas diante de uma invasão. Explica também que essa invasão tem diversas razões: roubo de

identidade; instigar o vitimado a fornecer informações sigilosas de forma a fazê-lo acreditar estar fazendo o correto; atingir um computador que replique a mensagem ou ação a vários outros; utilizar o computador da vítima para cometer algum outro crime, a fim de esconder a verdadeira autoria do mesmo.

2.1.2 Conhecendo o invasor e técnicas anti-hacker

Conhecer o invasor é pensar como ele pensaria no momento de planejar e executar um ataque ou invasão, com o objetivo de tomar atitudes preventivas para evitar danos e prejuízos à organização. Identificados genericamente com hackers, essas pessoas tem objetivos diversos e provocam conseqüências de grandes proporções.

De acordo com Parodi (2008), hacker é um programador que desenvolve os hacks, que são modificações para melhoria, exploração ou extensão de uma programação já existente, podendo utilizar-se de falhas de um sistema para obter acesso não autorizado.

Segundo Laudon e Laudon (2004, p. 462), “hacker é uma pessoa que consegue acesso não autorizado a uma rede de computadores, com o intuito de obter lucro, intervir criminosamente ou desfrutar prazer pessoal.”

Algumas técnicas que possibilitam a prevenção e o conhecimento da ação desses programadores mal-intencionados são de fundamental importância. Saber quais os tipos de ataques, quais as ferramentas e métodos utilizados pelos hackers podem garantir maior segurança e menor vulnerabilidade dos dados.

De acordo com Ulbrich e Valle (2006, p. 331), existem basicamente dois tipos de ataque, os indiretos e os diretos, em que a eficiência de uma invasão pode ser a junção dos dois tipos e depende muito do conhecimento do hacker em cada processo executado durante a invasão. O ataque indireto pode ser resumido em ações que enganam o usuário, por exemplo, fazendo com que o mesmo forneça senhas e dados sigilosos de uma empresa. Já no ataque direto, o hacker tem contato real com a vítima, e colhe informações pessoalmente ou por telefone.

Alguns perigos que os hackers podem representar ao computador serão apresentados e também algumas ações preventivas para os mesmos.

Assunção (2002, p. 41), cita alguns perigos que os hackers podem trazer para o computador invadido. Um deles é através do *trojan*, que o autor explica ser um vírus inteligente, pois é controlado, à distância, pelo hacker que o instalou, e que pode resultar em “perda de arquivos, falhas na memória, erros em periféricos, etc.”, ou seja, o hacker pode invadir o computador, e mantê-lo sob sua influência, mexendo simplesmente o mouse ou até mesmo, utilizando o endereço de IP em outros tipos de ataque. O autor explica ainda que um *trojan* pode ser iniciado junto com a máquina, pois podem estar presentes nos arquivos de inicialização.

Para Ulbrich e Valle (2006, p. 331), o tratamento para um *trojan* é basicamente, seguir regras simples, como não aceitar arquivos de estranhos, principalmente quando não se conhece a procedência para saber se é algo nocivo ou não, o que é muito difícil ser identificado por um usuário comum; quanto a um usuário mais experiente, há algumas prevenções como utilizar sempre o antivírus e firewalls locais “como o *Zone Alarm*, *BlackIce* ou o *Tiny Firewall*” que ajudam na defesa contra os invasores. O autor destaca também um ponto muito importante, que é a “adoção de políticas de usuários e administração”, que num exemplo simples, é a definição de quais usuários poderão instalar programas.

Assunção (2002, p. 47), cita alguns métodos anti-*trojan*, dentre eles o de detecção por portas, que funciona da seguinte forma: “os programadores estudam as portas TCP e UDP utilizadas pelos *trojans* e criam um programa que abre essas portas.”, desta forma o invasor pode se confundir e acreditar que pelo fato de a porta já estar aberta, existe um *trojan* instalado, e cair numa armadilha, tendo seu endereço de IP identificado. Outro método é o da detecção pelo arquivo, que encontra um *trojan*, analisando sua estrutura e torna-se muito eficaz utilizando a detecção de portas em conjunto.

Assunção (2002, p.47), explica ainda que o método mais eficaz é a detecção por *string*, que detecta um *trojan*, sem falhas, considerando que o programa pode ou não estar comprimido ou que mude as portas de uso, de tempo em tempo; pois a comunicação se dá através de *strings*, enviadas entre cliente e servidor, assim é possível identificar através de scanner de porta, quais as outras portas do sistema que estão abertas e contém *string* de comunicação. O autor cita também os métodos

manuais de verificação de “registros, arquivos de inicialização, conferindo os programas carregados na memória, o tamanho dos arquivos, etc...”

Segundo Laudon e Laudon (2004, p.462), outro exemplo de perigo e invasão que o hacker pode causar a vítima é o “ataque de recusa de serviço” (ataque DoS – *denial of services attacks*), em que o hacker carrega quase que totalmente um servidor de rede ou web, com o objetivo de inutilizar seu uso, prejudicando os acessos devidos e os serviços que deveriam estar sendo executados, com isso pode também causar danos ao desempenho do computador.

Para McClure, Scambray e Kurtz (2003, p. 535), algumas das várias ferramentas que os invasores têm utilizado para realizarem o ataque de recusa de serviço, são: smurf e fraggle:

- *Smurf*: “O ataque *Smurf* é um dos mais assustadores ataques de DoS, devido aos efeitos de amplificação do ataque.”, em que o efeito de amplificação é resultado da solicitação *ping* broadcast direcionada para uma rede, onde essa solicitação, permite saber o que está ativo. O atacante envia pacotes *ICMP ECHO (protocolo de manutenção, onde “echo equest” verifica a conectividade entre hosts, e o “echo reply” é a resposta* – Scrimger, 2002, p. 119*)* forjados para o endereço retornado da solicitação, e o endereço da vítima também é forjado, com o objetivo de parecer que o sistema da vítima fez uma solicitação. Desta forma todos os sistemas da “rede amplificadora” responderão a vítima, saturando completamente sua largura de banda.
- *Fraggle*: “é basicamente um ataque *Smurf* que usa *UDP* em vez de *ICMP*”, e tem funcionamento equivalente, ou seja, pacotes *UDP* forjados são enviados a “rede amplificadora”, consumindo largura de banda. (McCLURE, SCAMBRAY e KURTZ, 2003, p. 541)

Assim, McClure, Scambray e Kurtz (2003, p. 542), explicam que como medidas de proteção contra o *Smurf*, “a funcionalidade do broadcast direcionado deverá ser desativada” no roteador de “saída”. E também, explicam que em alguns sistemas operacionais específicos, é possível configurá-los para que descartem sem qualquer aviso, os pacotes *ICMP ECHO*.

Outra ferramenta bastante utilizada contra a invasão de hacker é a *honeypot*, que pode ser traduzida como “pote de mel” e que pode ser entendida, neste contexto, como um recurso para atrair os invasores e obter informações do mesmo.

Spitzner (2003) define *honeypot* como uma ferramenta de segurança que esconde suas verdadeiras funções quando o sistema está sendo atacado, invadido, ou comprometido, ou seja, esta ferramenta permite a detecção de um ataque e detém o mesmo, possibilita analisar e capturar os ataques e também fornece informações importantes sobre os hackers. O autor explica ainda que existem níveis de interação entre a *honeypot* e o invasor, onde esses níveis podem ser definidos como:

- Baixo: refere-se a serviços emulados pela ferramenta *honeypot*, onde a interação com o invasor é baixa, com isso não é possível a coleta de muitas informações, normalmente, é obtido apenas informações sobre as tentativas de conexão.
- Médio: este nível também se refere a serviços emulados, mas a resposta às requisições equivalem a respostas reais, com isso a interação entre o invasor e a *honeypot* é um pouco maior e conseqüentemente a coleta de informações é mais satisfatória.
- Alto: nesse nível os serviços não são mais emulados, são reais, com máquinas reais e sistemas operacionais reais, com isso o nível de interação entre o invasor e a *honeypot* é muito maior que os demais, possibilitando uma arrecadação de informações sobre a invasão. (SPITZNER, 2003)

Spitzner (2003) adverte que quanto maior o nível de interação entre o invasor e a *honeypot*, maior também será o risco, pois quando ocorre uma invasão o autor tem acesso a um sistema operacional real, que pode causar danos maiores e pode atacar até mesmo outras máquinas na rede.

Segundo CHESWICK, BELLOVIN e RUBIN (2005, p. 33), a *honeypot* é a máquina que ninguém deveria mexer, pois qualquer fonte de tráfego é no mínimo um mau comportamento e muito provavelmente tem intenções maliciosas.

Outro exemplo de técnica anti-*hacker* é a *DMZ* (*Demilitarized Zone* – Zona desmilitarizada).

Segundo CHESWICK, BELLOVIN e RUBIN (2005, p. 33), a *DMZ*, é a zona de ligação entre dois *firewalls*. Nesta zona, é necessário um cuidado especial, pois nela que as informações sigilosas estão mais vulneráveis aos ataques. O acesso a essa zona deve ser somente da rede interna e “preferivelmente por uma conexão criptograficamente protegida, como ssh.”. A *DMZ* funciona basicamente como mais uma camada, um escudo que dificultará o acesso indevido: “Se um invasor passar pelo primeiro *firewall*, ele ganhará acesso a *DMZ*, mas não necessariamente à rede interna. Sem a *DMZ*, o primeiro ataque bem-sucedido poderia resultar em um comprometimento mais sério.”.

Assim, conhecendo alguns dos perigos a que o computador está exposto, e sabendo que a cada dia surgem novas ferramentas e técnicas que aprimoram essas táticas perigosas da invasão dos hackers, proporcionalmente surgem também as técnicas anti-hackers. Porém, mesmo com a segurança relativamente sob controle os crimes são praticados no meio computacional, sendo necessária uma perícia técnica no computador e dispositivos eletrônicos, para isso têm se a perícia forense computacional, que realiza essa perícia em que o laudo produzido é utilizado no julgamento de hackers.

2.2 PERÍCIA FORENSE COMPUTACIONAL

"Perícia forense" vem do latim, onde: "perícia" - *peritia* (exame de caráter técnico; vistoria), e "forense" - *forense* (se refere ao foro judicial). (FREITAS, 2006, p.1)

Segundo Costa (2005), "a computação é a organização e execução de rotinas e métodos de caráter repetitivo e sua relação com a informação", ou seja, a automatização de diversas áreas, abrangendo todos os tipos de problemas, que vão de "sistemas governamentais ao simples controle de batimentos cardíacos em um relógio de pulso". Enfim, em todos os componentes eletrônicos, há uma atividade computacional. E o autor conceitua "perícia forense", como o estudo técnico realizado por um profissional, relatado em um laudo; em que este é utilizado em investigações, para auxiliar no julgamento de crimes.

Para Farmer e Venema (2000), perícia forense computacional é definida como a técnica de arrecadação de evidências, sendo que essa técnica deve cuidar para que as mesmas permaneçam originais, sempre que possível; a fim de serem utilizadas como prova ou argumentação para esclarecer um fato ocorrido, ou seja, auxiliar numa investigação judicial.

Segundo Freitas (2006, p.1), pode se conceituar a perícia forense como a vistoria e análise de informações, para solucionar um crime ou enigma; sendo que o resultado desta deve ser utilizado em tribunais como provas.

De acordo com Farmer e Venema (2007, p.4), para realizar a perícia forense e chegar à solução do enigma, é preciso estar atento a qualquer tipo de informação, como se procurasse algo quase que desconhecido, ou seja, sem saber precisamente o que se procura; apenas considerando tudo aquilo que poderia ser fundamental, ou seja, necessário buscar informações em todos os lugares sem preconceitos ou buscando algo já esperado. Explica que também é preciso estar aberto a sugestões e novidades, para que não fique apenas no conhecimento comum:

> [...] confiar na experiência passada, ouvir os conselhos de outras pessoas e usar as ferramentas existentes. Mas também não tenha medo de transformar a sabedoria comum em um mito, criar suas próprias ferramentas e desenvolver uma metodologia própria quando isso for necessário para decifrar um caso. Do contrário, você pode acabar com uma

> pessoa que procura as chaves perdidas sob um poste de luz porque a luz aí é mais forte. (FARMER E VENEMA, 2007, p. 4)

Portanto, perícia forense computacional, é uma área da investigação jurídica que tem seu foco voltado a crimes que ocorreram num contexto computacional ou com o auxílio de qualquer meio eletrônico. Desta forma, é preciso que o profissional da perícia tenha vasto conhecimento na área, para saber onde procurar, o que procurar, de que forma procurar e como proceder com essas informações depois de encontradas; assim obtendo conclusões satisfatórias numa investigação.

2.2.1 Procedimentos da Perícia

A perícia forense é uma análise realizada seguindo-se metodologias, ou seja, procedimentos padrões, que se dispõem seqüencialmente, com o objetivo de regulamentar esse estudo técnico.

Segundo Freitas (2006, p. 2), a realização da perícia, é baseada em procedimentos que objetivam a conclusão da investigação criminal, onde esses podem ser resumidos basicamente em identificação, preservação, análise e apresentação: “[...] todas as evidências devem ser identificadas, preservadas, analisadas e apresentadas.”

De acordo com Adams (2000), esses procedimentos, devem estabelecer e manter um sistema de qualidade eficaz; e define os POPs (Procedimentos Operacionais Padrão) como documentos de controle de qualidade e orientações coerentes com princípios científicos e jurídicos que são essenciais para a aceitação das informações por parte dos tribunais e outros organismos; sendo que esses POPs devem ser desenvolvidos por autoridades, e devem ser revistos anualmente para que sejam atualizados e mais eficazes.

Assim é importante que o perito siga esses procedimentos para que chegue a um final satisfatório da perícia, pois cada um desses procedimentos dará à informação o tratamento adequado para que realmente possa ser utilizada e apresentada juridicamente.

2.2.1.1 Identificação

Na etapa de identificação das evidências, verifica-se que, para cada tipo de crime, têm-se tipos de evidências diferentes, e para isso o perito deve ter certo conhecimento referente a cada tipo de crime, e estar atento a novos métodos de prática desses crimes. (FREITAS, 2006, p.2)

Freitas (2006, p.2) exemplifica alguns dos tipos de evidência para cada tipo de crime:

> Por exemplo, em um caso de acesso não autorizado, o perito deverá procurar por arquivos de log, conexões e compartilhamentos suspeitos, já em caso de pornografia, buscará por imagens armazenadas no computador, histórico dos sites visitados recentemente, arquivos temporários do browser etc. (FREITAS, 2006, p.2)

Freitas (2006, p.2), explica que, sabendo os tipos de evidência para cada crime, é preciso saber também que alguns locais e objetos têm grande importância mesmo que não pareçam e por isso não devem ser ignorados no momento das buscas por evidências, como: qualquer tipo de dispositivo eletrônico de armazenamento, desde um HD até celulares, e também todo e qualquer documento impresso, números de telefones, nomes de pessoas, etc.

A principal fonte de informações numa perícia computacional, normalmente, é o computador, que pode conter informações aparentes ou não, e que podem ser decisivas, por serem mais diretas, sem gerar muitas dúvidas ou suposições, ou seja, no caso de encontrar um arquivo de log, desde que provada sua autenticidade e legalidade, não há muito que questionar, pode ser sim considerado como prova de algum fato ocorrido.

Segundo Farmer e Venema (2007, p. 9), para a identificação das informações num sistema de computador é preciso saber se as mesmas são autênticas e se não compõem uma armadilha que podem desviar a atenção do perito, prejudicando a reconstrução do que realmente aconteceu. Assim é preciso examinar cada fragmento de informação disponível a procura de inconsistências que mostrem uma tentativa de ocultar a realidade: “Quanto maior for o número das fontes de informações que você tiver e quanto maior for a independência dessas fontes entre si, maior será a confiabilidade de suas conclusões.”.

Para Freitas (2006, p.2), é indispensável identificar quais evidências podem ser realmente utilizadas durante a perícia: “Distinguir entre evidências relevantes e irrelevantes em uma análise ao vivo”, onde “análise ao vivo” é definida como:

> Perícia realizada no equipamento investigado ainda em funcionamento. Alguns itens que são realizados em uma análise ao vivo: identificação de processos em execução, observação de portas abertas no sistema e das conexões realizadas. (FREITAS, 2006, p.2)

Para Costa (2003), no período de identificação de evidências, é necessário interessar-se muito pelos discos rígidos removíveis e mídias, pois terão grandes chances de encontrar informações válidas.

Farmer e Venema (2007, p. 6), ressaltam que é impossível colher todas as informações de um computador, principalmente porque quando se está capturando os dados de uma parte do computador, simultaneamente estão sendo alterados dados de outra parte:

> Memória, processos e arquivos podem mudar tão rapidamente que o simples volume do registro dessas flutuações de uma maneira precisa e sincronizada não é possível sem perturbar radicalmente a operação de um sistema de computador típico. (FARMER E VENEMA, 2007, p. 6)

Porém, Farmer e Venema (2007, p. 6), explicam que dessa forma, não é possível recuperar o passado, reconstruindo integralmente todos os passos realizados, mas que isso não se faz necessário para que sejam tiradas conclusões razoáveis sobre o ocorrido.

Adams (2000), explica que esta etapa visa arrecadar informações e/ou bens físicos, e que por isso os mesmos devem ser armazenados para as futuras consultas e utilização durante a perícia. Porém, é preciso lembrar que esta arrecadação deve ter embasamento legal, e que as leis mudam de acordo com a jurisdição em que as provas estão localizadas, ou seja, devem ser aceitas por um oficial da lei ou seu representante.

Enfim, a identificação e arrecadação de evidências, é o início da perícia e por isso exige muita atenção dos peritos para que não deixem de buscar informações em lugares óbvios e também os não tão óbvios; buscar evidências aparentes e as não tão aparentes. Porém depois de encontradas é preciso saber se as mesmas podem ser utilizadas como provas na perícia forense, se sim, o próximo passo é a preservação.

2.2.1.2 Preservação

A fase de preservação é tão importante quanto a fase de identificação, pois o trabalho do perito poderá ser prejudicado ou até mesmo invalidado se as evidências identificadas forem perdidas ou danificadas. Assim esta fase pretende proteger ao máximo de danos lógicos e físicos, toda e qualquer informação identificada e que deva ser coletada para futuras análises.

Farmer e Venema (2007) iniciam essa fase de preservação dizendo que o procedimento de coleta de informações feita no computador, deve ser feito com cautela e planejamento, em que a primeira medida a ser tomada é isolar o computador de outros usuários que possam acessá-lo pela rede, para evitar que sejam corrompidos. Os autores explicam que alguns dados são mais suscetíveis à corrupção por coleta que outros, e por isso é relevante que neste momento seja seguida a ordem de volatilidade (OOV – *order of volatility*) que é baseada no ciclo de vida dos arquivos e pode variar de nanossegundos a anos aproximadamente (Quadro 2), garantindo assim que os dados mais propícios a corrupção sejam salvos a tempo.

Tipos de dados	**Tempo de vida**
Registradores, memória periférica, caches, etc.	Nanossegundos
Memória principal	Dez nanossegundos
Estado da rede	Milissegundos
Processos em execução	Segundos
Disco	Minutos
Disquetes, mídias de backup, etc.	Anos
CD-ROMs, impressões, etc.	Dezenas de anos

Quadro 2- O ciclo de vida esperado dos dados.
Fonte: Farmer e Venema, 2007, p. 6.

Segundo Costa (2003, p.15), há ainda outra preocupação e cuidado ao manusear um computador da cena do crime, a fim de preservar informações, pois se o mesmo estiver ligado, e necessitar ser desligado é preciso cautela, pois caso seja cortada a corrente de energia bruscamente, pode causar falhas e danos graves, podendo prejudicar até sua inicialização. Entretanto o autor explica que no caso de sistema operacional Windows, é possível utilizar o recurso de hibernar, onde “ao se

religar o sistema, volta-se à condição anterior do desligamento [...]". Porém se o computador estiver desligado, também é necessário cautela ao ligá-lo, pois o autor explica que uma inicialização qualquer pode modificar de alguma forma as evidências "uma inicialização não controlada pode comprometer os dados, o ordenamento seqüencial das evidências e uma efetiva caracterização das provas."

Segundo Freitas (2006, p.3), na etapa de preservação das evidências, o objetivo principal é assegurar que a evidência permanecerá sempre inalterada. E que no caso de "análise ao vivo", a sugestão é a duplicação pericial: "que consiste em criar uma imagem (cópia perfeita) de um sistema. Através da imagem o perito poderá realizar suas análises, preservando assim suas provas originais.". O autor exemplifica alguns hardware e softwares utilizados na duplicação pericial (Quadro 3):

Softwares	Hardwares
EnCase	Forensic SF-5000
SafeBack	Forensic MD5
ILook	DIBS Advanced Forensic Workstation
ProDiscover	DIBS Mobile Forensic Workstation
Forensic Replicator	Estações F.R.E.D
Forensic Toolkit	Duplicadores CSC
GetDataBack	
dd for Windows	

Quadro 3- Ferramentas utilizadas na duplicação pericial.
Fonte: Freitas, 2006, p. 140.

Para Farmer e Venema (2007, p.5), a duplicação pericial, também chamada de cópia de segurança dos dados, é imprescindível, uma vez que, a análise é feita baseada em coleta e processamento dos dados, é mais seguro ter sempre a versão original, caso alguma informação seja corrompida ou perdida: "Os dados originais permanecem protegidos em um estado puro; qualquer análise deve ser realizada em uma cópia dos dados do computador.". O autor faz ainda uma analogia ao procedimento de filmar a "cena" do crime, a fim de preservar suas originalidades.

Costa (2003, p. 25), adverte que deve ser evitado o exame direto na mídia original, sendo assim a cópia de segurança só não deve ser realizada se não haver possibilidade e ferramentas para fazê-la. Explica que o procedimento de cópia bit a bit preserva integralmente o conteúdo da mídia, e que se for possível é interessante

que se faça mais de uma cópia, para o caso de as análises terem de ser repetidas ou no caso de reprocessamento das informações.

De acordo com Adams (2000), a cópia, deve ser uma reprodução exata de informações, sendo que deve ser preservado o local em que estão originando, ou seja, o destino da cópia deve preferencialmente ser um local idêntico ao original.

Segundo Costa (2003) é importante gerar uma chave “hash” para a(s) cópia(s) de segurança, que consiste em gerar um código único, a partir de cada bit do arquivo, de forma que se o mesmo gerador de chave for aplicado à cópia e ao original, deverá resultar na mesma chave, atestando a integridade entre o original e a cópia.

De acordo com Freitas (2006), as evidências devem ser preservadas adequadamente de quaisquer problemas mecânicos, elétricos ou eletromagnéticos. E destaca algumas dicas importantes, após a arrecadação as evidências devem ser: lacradas em sacos e etiquetadas, onde na etiqueta deve conter o número de identificação, o número do caso, data e horário da coleta e o nome da pessoa que está levando para custódia; os cabos e componentes apreendidos também devem ser etiquetados; na análise ao vivo, gravá-las em disquetes e proteger contra regravação; os HDs devem ser armazenados em sacos antiestática; no transporte a cautela é em relação a líquidos, umidade, sujeira, calor, etc.; e também devem ser armazenadas e trancadas para evitar adulteração.

2.2.1.3 Análise

O procedimento da análise das evidências vem logo após coleta e identificação, e exige certo trabalho, tanto quanto os outros procedimentos, pois é nessa fase que se mostra qual a importância de uma determinada evidência e mostra alguns detalhes técnicos que podem reforçar o valor da mesma.

Segundo Freitas (2003, p. 4), o objetivo da análise das informações é “identificar quem fez, quando fez, que dano causou e como foi realizado o crime.”, desta forma é imprescindível “saber o que procurar, onde procurar e como procurar”. O autor destaca as principais perguntas que o perito deve saber responder ao término da análise das evidências: “Qual a versão do sistema operacional que

estava sendo investigado?"; "Quem estava conectado ao sistema no momento do crime?"; "Quais arquivos foram usados pelo suspeito?"; "Quais portas estavam abertas no sistema operacional?"; "Quem logou ou tentou logar no computador recentemente?"; "Que eram os usuários e a quais grupos pertenciam?" e "Quais os arquivos excluídos?".

Freitas (2003, p. 4) lembra ainda que nessa fase é preciso atestar a autenticidade da evidência, bem como definir se a mesma é exata e completa, para convencer o júri, ou seja, se está em conformidade com a lei. E também, adverte que todas as ações realizadas nessa fase devem ser documentadas, pois essas anotações, ao decorrer da análise, devem ajudar na formulação da solução de um caso.

2.2.1.4 Apresentação

A fase de apresentação é normalmente a última fase, e pretende demonstrar todo o trabalho realizado durante a investigação, bem como seus resultados e conclusões. Consiste em apresentar um relatório ao júri, com todos os pontos verificados e analisados na investigação a fim de tornar conhecido o que, provavelmente, tenha ocorrido.

Segundo Freitas (2006, p. 5), durante a identificação, a preservação e a análise, o perito deve descrever e documentar todas as ações realizadas nesses processos, tendo em vista que toda essa documentação deve ser apresentada ao final da investigação. Assim: "O perito faz o laudo, e a partir das evidências a decisão é da Justiça.". Portanto na apresentação das evidências o perito deve ter redigido um relatório técnico sobre a investigação, com todos os fatos ocorridos, todas as análises e resultados.

Freitas (2006, p. 5) explica alguns detalhes importantes que devem ser considerados durante a redação do laudo, que deverá ser apresentado:

- Sua escrita deve se de maneira explícita, concisa, estruturada e sem ambigüidade, a fim de não deixar dúvidas sobre sua veracidade, durante sua análise.

- Em seu conteúdo deve estar todas as informações sobre os métodos utilizados na perícia, como por exemplo, os procedimentos de identificação, preservação e análise; e também conter informações sobre os hardwares e softwares utilizados.
- As afirmações e conclusões formuladas no laudo devem ter embasamento técnico e científico que possam comprová-las.

Essas dicas demonstram a importância da fase de apresentação, sendo que ela poderá conduzir a linha de raciocínio do júri, pois é competência do perito reconstruir o fato através da investigação, para que o júri possa chegar à conclusão final.

2.2.2 Cadeia de custódia

Cadeia de custódia é um documento que visa “acompanhar” as evidências em todos os lugares em que estiver. Funciona como um registro cronológico que pode garantir mais seriedade a evidência, pois tem o objetivo de rastreá-la.

Para Freitas (2006, p.5), cadeia de custódia é o documento que registra o local e a data em que a evidência estava sendo estudada, bem como o nome do responsável, no momento; e também uma pequena descrição justificando a posse da evidência pelo responsável em questão: “a cada vez que as evidências passarem de uma pessoa para a outra, ou de um tipo de mídia para outro, a transação deverá ser registrada.”. Assim a cadeia de custódia prova e declara por onde as evidências percorreram, determinando que a integridade das evidências não foi comprometida.

De acordo com Freitas (2006, p.5), o documento deve conter basicamente as seguintes informações: data e local, número do caso a qual ela pertence, nome do perito, descrição da evidência no cabeçalho do formulário. Na parte de movimentação, deve conter origem (nome e local) e destino (nome e local), data e motivo da custódia. Veja na Figura 2, um exemplo:

Data / Local			
Caso Nº.			
Perito			
Descrição da Evidência			
CADEIA DE CUSTÓDIA			
De Local	Data	Razão	Para Local
De Local	Data	Razão	Para Local
De Local	Data	Razão	Para Local
De Local	Data	Razão	Para Local
De Local	Data	Razão	Para Local
De Local	Data	Razão	Para Local

Figura 2- Formulário para Cadeia de Custódia
Fonte: Freitas, 2006, p. 6.

2.2.3 Técnicas Anti-Forense

As técnicas anti-forense, são aquelas que pretendem ultrapassar as barreiras que a perícia forense tenta colocar para impedir que os indícios de uma ação criminosa passem despercebidos durante a investigação. Consistem basicamente em técnicas que, nesse contexto, deixam de ser garantir a “segurança legal” e passam a atuar do lado contrário, ou seja, encobrem feitos ilegais, com o objetivo de garantir que o autor da ação não se torne conhecido.

Segundo Siles (2007), a ciência anti-forense, tem o objetivo de camuflar os rastros que podem ter sido deixados durante uma invasão computacional; essa ciência é composta de diversas técnicas que podem prejudicar a perícia forense

computacional, por isso, é de grande importância conhecê-las a fim de impedir que as informações de uma invasão passem despercebidas.

Siles (2007), explica que o objetivo de se estudar as técnicas anti-forense, é proporcionar ao perito os conhecimentos necessários para que possa identificar as técnicas que o criminoso utilizou para encobrir os rastros que podem ter sido deixados na invasão.

Algumas técnicas como criptografia, esteganografia, *wiping*, *slack space*, e *data hiding*, de acordo com o contexto, podem ser tratadas como anti-forense e serão descritas a seguir.

A criptografia é uma das ferramentas mais conhecidas para anti-forense que pode ser definida como o verdadeiro método para este objetivo. Mesmo sendo mais eficiente que a limpeza (técnica conhecida como *Wiping*), a criptografia tem suas fraquezas, dependendo do tipo utilizado, sendo dividida em: assimétrica e simétrica. (PHILIPP; COWEN; DAVIS, 2010, p. 205).

Segundo Tanenbaum (2003, p. 770), criptografia vem do grego, e significam “escrita secreta”, ou seja, trata de uma técnica que visa transformar a informação, conhecida como “texto simples” através de uma função que é comandada por uma “chave”, resultando em um “texto cifrado”. Sua utilização tem o objetivo de manter informações confidenciais e garantir sua integridade e autenticidade. O autor explica também que os algoritmos criptográficos usam: “transformações complexas que envolvem substituições e permutações para transformar o texto simples em texto cifrado.”

Philip; Cowen; Davis (2010, p. 206), explicam algumas técnicas para identificar e acessar a chave simétrica e assimétrica, com ferramentas como *Forensic Tool Kit* (FTK) e *Password Recovery Toolkit* (PRTK). Quanto à utilização da FTK: quando os dados são trazidos para FTK, é possível executar um teste de entropia sobre os dados para determinar se poderia ser criptografados com um algoritmo conhecido. O resultado da entropia é exibido quando a evidência é adicionada à FTK. Assim, terminada a indexação e análise, os arquivos criptografados poderão ser identificados. Quanto a PRTK, o autor explica que se trata de uma ferramenta robusta para recuperação da chave do arquivo criptografado.

Enfim, o objetivo dessa técnica, no contexto anti-forense, é descobrir arquivos criptografados e suas respectivas chaves para a descriptografar.

Segundo Stallings (2008, p. 34), outra técnica é a esteganografia, que é o método que esconde a existência da mensagem, e explica um exemplo simples de sua aplicação: o rearranjo de palavras e letras dentro de um texto aparentemente inofensivo que deve soletrar a mensagem real.

Esse assunto será abordado no item 2.3, que deverá apresentar e estudar mais especificamente, por ser um dos objetivos dessa pesquisa.

Para Philip; Cowen; Davis (2010, p. 206), existe diversas ferramentas confiáveis para detectar esteganografia atualmente. *Stegdetect, é uma delas,* permite que sejam inspecionados no computador os arquivos JPEG que contenham informações ocultas, essa ferramenta é capaz de descobrir esteganografia aplicada através de diversas técnicas. A principal aplicação comercial existente hoje é a *Stego Suite* por *Wetstone*. Além de identificar os arquivos esteganografados, é possível também procurar no sistema a presença de programas de esteganografia instalados.

A técnica *Wiping* significa “limpando”, e nesse contexto que dizer limpado dados, e também é uma técnica anti-forense relativamente bastante eficiente, pois não dá muitas chances de recuperação dos dados limpados do disco.

Philip; Cowen; Davis (2010, p. 206), explicam que *wiping* é uma técnica que causa sérios problemas quando aplicada corretamente. Pois quaisquer dados que tenham sido verdadeiramente varridos do disco, foram substituídos pelo menos uma vez. Assim, usando as ferramentas existentes atualmente, não é possível acessar dados que tenham sido substituídos, daí a causa dos sérios problemas. O que se pode fazer é determinar se ferramentas de limpeza de disco foram instaladas.

Outra técnica é a *Slack Space* é um resquício de dados existentes em um setor de dados que foi substituído. Especificamente, o *slack space* é a área do setor que não foi totalmente substituída por uma recente gravação em disco. Os setores são fixados em seu tamanho, por isso, por exemplo, se está sendo utilizado 3k de dados para um setor de 64k, os 61k restantes de dados não seriam reutilizados. Em vez disso, este espaço não utilizado do setor ainda pode conter dados que foram escritos antes. (PHILIP; COWEN; DAVIS, 2010, p. 206)

Segundo Stewart; Tittel e Chapple (2008, p. 188), *Data Hiding*, é a técnica de esconder dados, ou seja, evitar que os dados sejam descobertos ou acessados por pessoas não autorizados, que no contexto anti-forense, o perito também não deixa de ser uma “pessoa não autorizada”.

Essas técnicas visam modificar, mascarar e esconder a informação sendo utilizadas como ferramentas de segurança. Por outro lado, essas ferramentas podem ser utilizadas como algo que atrapalhe ou prejudique o bom desempenho da investigação, pois poderão esconder indícios fundamentais durante a perícia forense computacional, desse princípio surge sua importância.

2.3 ESTEGANOGRAFIA

Com o crescente desenvolvimento dos meios de comunicação, descobriu-se a praticidade de trocar informações, sejam elas sigilosas ou não, através da internet. As organizações têm utilizado muito esse recurso, um exemplo disso são os investimentos realizados à distância. Essa praticidade exige certa segurança, pois informações sigilosas despertam interesses de pessoas maliciosas que podem utilizá-las como arma.

Mas existem técnicas que permitem transmitir essas informações de maneira mais segura, como é o caso da esteganografia, que possibilita esconder a informação desejada em arquivos simples e que poderiam passar facilmente despercebidos. Porém, essa técnica pode ser usada de maneira maliciosa, escondendo informações que podem ser evidências de um crime.

Definição: “A ciência de ocultar mensagens é chamada de esteganografia, das palavras gregas que correspondem a ‘escrita cifrada’.” (TANENBAUM, 2003, p. 876).

Para Tanenbaum (2003, p.876), a esteganografia, tem como objetivo possibilitar que mensagens sejam transmitidas com segurança e identificadas apenas pelo destinatário correto. O autor cita como exemplo o tempo em que os gregos já utilizavam desta técnica, em uma passagem que diz que Heródoto transmitiu uma mensagem secreta no couro cabeludo de um general, ou seja, os “fins” são os mesmos, os “meios” é que evoluíram.

Para Johnson e Jajodia (1998) a esteganografia é a técnica que visa esconder o maior número de informações de um determinado arquivo, tendo como exemplo alguns métodos de camuflagem, as tintas invisíveis e arranjo de caracteres.

Johnson e Jajodia (1998) explicam também que esteganografia e criptografia são conceitos próximos, com a diferença que a criptografia, faz com que a mensagem se torne um código que poucos poderiam decifrar, mas deixa claro que há a presença de alguma informação e que provavelmente esta é secreta, despertando a curiosidade e a suspeita, em conseqüência disso pode atrair olhares maliciosos; já a esteganografia esconde completamente a mensagem dentro de outra informação ou imagem, o que limita muito sua descoberta.

Deitel (2002, p.611) como síntese dos conceitos de esteganografia, explica que: um arquivo texto com informações sigilosas, que deve de ser enviado com plena segurança, onde apenas o destinatário correto poderá ver estas informações, o mesmo deve então receber o tratamento da esteganografia, ou seja, por exemplo, mascarar o mesmo com uma imagem qualquer, que não desperte interesse algum.

A prática da esteganografia, já vem sendo utilizada a tempos, com ferramentas não tão avançadas quanto hoje, mas que pareciam ser bem eficientes, e tinham o mesmo objetivo.

Stallings (2008, p.35) apresenta algumas delas:

- Marcação de caractere: algumas letras de um texto impresso ou datilografado são sobrescritas por lápis, para compor a mensagem escondida. Essa sobrescrita normalmente não é visível, a menos que o papel seja mantido em ângulo com uma fonte de luz clara.
- Tinta invisível: caracterizada pelo uso de diversas substâncias que são invisíveis, a não ser com a aplicação de alguma outra substância que a torne visível.
- Perfuração: pequenos furos feitos em algumas letras do texto, que são praticamente invisíveis, mas que se o papel for colocado contra a luz poderá definir a mensagem.
- Fita corretiva de máquina de escrever: aplicada em um texto já datilografado com fita preta normal, em que a mensagem deseja é datilografada por cima, formando uma marca visível apenas com luz forte.

Stallings (2008, p.35) complementa: “Embora essas técnicas possam parecer arcaicas, elas possuem equivalentes contemporâneos.”.

No contexto atual, muitas nomenclaturas são usadas na explicação de esteganografia, em que essas deverão ser utilizadas durante o texto:

Segundo Johnson e Jajodia (1998), a incorporação da informação a ser protegida, tem as seguintes terminologias: imagem de capa (imagem que deve parecer inocente, não despertar interesse), mensagem ou informação a ser ocultada (texto simples ou cifrado, outra imagem, ou qualquer coisa que possa ser incorporada a um fluxo de *bits*). A *stego-image* (estego-imagem) é o resultado dessa combinação, a capa e a mensagem embutida. Existe ainda a *stego-key* (estego-

chave) que é a chave que pode ser utilizada na codificação e decodificação da mensagem, com o objetivo de restringir a detecção e recuperação da mensagem.

A esteganografia pode ser realizada em arquivos de imagem, arquivos de áudio e vídeo, e até mesmo em pacotes do protocolo TCP/IP. Serão citadas a seguir a utilizações em imagem, áudio e vídeo, essas duas últimas com a técnica da marca d'água.

2.3.1 Imagem

Algumas das principais aplicações de esteganografia estão relacionadas a esconder informações dentro de imagens, dessa técnica surgem outras três técnicas: a inserção do bit menos significativo; máscara e filtro; e algoritmos e transformações.

Johnson e Jajodia (1998), explicam essas técnicas podem ser aplicadas ao mesmo arquivo, porém podem ter resultados diferentes.

2.3.1.1 Inserção do bit menos significativo - LSB

A inserção do *bit* menos significativo ou bit de baixa ordem da imagem de capa pode ser entendido como a utilização desses *bits*, preenchendo-os com as informações a serem escondidas.

Tanenbaum (2003, p. 876) mostra como funciona a esteganografia em uma imagem de 1024 x 768 pixels. "O método de codificação esteganográfico utiliza o *bit* de baixa ordem de cada valor RGB como um canal oculto."

Segundo Tanenbaum (2003, p. 876), no sistema RGB de cores, cada pixel é formado pela combinação das cores vermelha, verde e azul, com 8 *bits* (1 *byte*) para cada cor, ou seja, para cada *pixel* temos um conjunto de 24 *bits*. Nessas condições, o autor explica que o olho não consegue ver diferença entre cores com 24 *bits*, e cores com 21 *bits*. Assim, utilizando o esquema de cores com 21 *bits*, restariam ainda, em uma imagem de 1.024 x 768 *pixels*, um total de: 1.024 x 768 x 3 x 8 = 2.359.296 *bits* (*bits* de baixa ordem) ou 294.912 *bytes*. Esses *bytes* poderiam ser utilizados para armazenar informações secretas, que ficariam completamente

invisíveis. A esse “espaço” que se obtém em um arquivo, para esconder informações, dá-se o nome de “largura de banda esteganográfica”.

Segundo Johnson e Jajodia (1998), uma imagem de 24 *bits* poderia atrair atenção durante sua transmissão, e por isso sugere alguns meios de compressão dessa imagem, mas explica que temos dois tipos, com perdas e sem perdas, em que ambos poupam espaço de armazenamento, mas possuem resultados diferentes, podendo comprometer a informação contida na banda esteganográfica, quando descompactada. Essa compactação é caracterizada como a forma de salvar a imagem:

- Compressão *Loss Less* - GIF (*Graphic Interchange Format*) ou BMP (arquivo *Bitmap*): mantém a integridade da imagem original.
- Compressão *Lossy* - JPEG (*Joint Photografic Experts Group*): economiza espaço, porém oferece aproximações perto da alta qualidade, mas não uma cópia exata.

2.3.1.2 Máscara e filtro

De acordo com Johnson e Jajodia (1998), essa técnica é aplicada apenas a imagem com tons de cinza (*grayscale*). Tende a ser mais robusta que a LSB, no que diz respeito à compressão, corte e processamento de imagem, por incorporar a informação às partes significativas da imagem. A técnica funciona da seguinte maneira: oculta informações pela marcação da imagem, semelhante a uma marca d’água no papel, mas com a diferença em relação à LSB, pois se a imagem for modificada por métodos de compressão a marcação não será perdida.

2.3.1.3 Algoritmos e transformações

Johnson e Jajodia (1998) comparam essas técnicas com a LSB, pois a LSB é uma maneira rápida e fácil de esconder informações, mas é vulnerável a pequenas alterações resultantes do processamento de imagens e compressão lossy, que é a principal vantagem da imagem JPEG. Existem algoritmos que realizam a compressão de imagens JPEG através do arredondamento do coeficiente DCT

(transformação discreta do cosseno) para compactar. A ferramenta *Jpeg-Jsteg* é um exemplo.

Segundo, Johnson e Jajodia (1998) propriedades da imagem como a iluminação também podem ser manipuladas. Técnicas semelhantes usar a codificação padrão redundante ou métodos de propagação de espectro (*spread spectrum methods*) para espalhar informações ocultas em toda a imagem de capa. Explicam também que essa técnica pode ajudar a proteger contra o processamento de imagem, como corte e rotação, e esconder a informação num nível mais profundo que o simples mascaramento. E que suporta a manipulação de imagens mais facilmente do que a técnica LSB.

2.3.2 Técnicas com marca d'água

Para Deitel (2002, p. 611), outra aplicação da esteganografia utilizada hoje, é a marca d'água digital, que tem a finalidade de proteger a propriedade intelectual, onde é adicionado ao documento digital, um símbolo, uma frase ou uma indicação qualquer que possa ser identificado o verdadeiro autor do mesmo.

Tanenbaum (2003, p. 877), também referencia a utilização da esteganografia, como marca d'água, para garantir os direitos autorais. E explica que, no caso da necessidade de provar a autoria diante de um tribunal, o verdadeiro dono poderá revelar a localização da marca d'água, assegurando todos os direitos sobre a mesma. Explica que não só os arquivos de imagem podem ser utilizados na esteganografia, mas também os arquivos de áudio, que podem ser bem eficientes no resultado, assim como um arquivo HTML.

Deitel (2002, p.611) explica que esta marca d'água digital pode ser visível ou invisível, onde, por exemplo, no caso de uma música, existem softwares que aplicam a esteganografia, para criar uma marca d'água digital imperceptível, modificando um trecho da música.

Segundo Deitel (2002, p.611), um software que realiza este tipo de técnica é o *GiovanniTM*: "O software de inserção de marca d'água digital *GiovanniTM*, da *Blue Skipe*, utiliza chaves criptográficas para gerar e embutir marcas d'água digitais com esteganografia em música e imagens digitais.". Explica também que o mesmo utiliza-

se da criptografia para gerar uma chave secreta, feito isso, identifica uma área que não modifique o visual ou o áudio, a partir daí utiliza-se da esteganografia para marcar o arquivo, fazendo com que, se a marca for removida, o conteúdo se danifique.

2.3.3 Ferramentas para esteganografia

Existem diversas ferramentas disponíveis atualmente para esteganografar informações, sejam eles arquivos de texto, de imagem ou de música. Algumas dessas ferramentas especificam o tipo de arquivo a ser esteganografado e qual será o tipo do arquivo de destino; há também algumas particularidades quanto ao nível de segurança que podem proporcionar.

Coelho e Bento (2004), apresentam algumas ferramentas de esteganografia de forma sintética, explicando suas principais características:

- ***Camouflage*** (plataforma *Windows*)**:** ao ser instalada, a ferramenta cria duas opções no menu: "*Camouflage*" e "*Uncamouflage*". Clicando com o botão direito do mouse no arquivo desejado o usuário poderá realizar essas duas operações. Cuidados devem ser tomados ao aplicar a arquivos txts, pois ao abrir em um editor paralelo, poderá mostrar sinais que tem algo escondido. E também ao transferir o arquivo esteganografado via FTP, pois a maioria das transferências desse protocolo utiliza codificação ASCII, o que poderá corromper a informação oculta.
- ***Ezstego*** (plataformas que suportem Java)**:** utiliza a imagem GIF, substituindo os últimos *bits* significativos da imagem por dados a serem escondidos. Não disponibiliza meios de tratar a imagem.
- ***Hide and Seek*** (plataforma DOS)**:** suas versões mais atuais, apesar de ser plataforma DOS, já têm interface gráfica. Utiliza a imagem GIF, ocultando arquivos texto utilizando ou não a *passphrase* (frase-senha), que pode proporcionar mais segurança.
- ***Hide4PGP*** (plataformas DOS, OS/2 e Linux)**:** esconde qualquer formato de arquivo em imagens tipo bitmap com 256 ou 16.7 milhões de

cores; arquivos WAV com 8, 12 ou 16 *bits* sem compressão, mono ou estéreo; e arquivos VOC de 8 *bits*.

- ***JPHide e JPSeek*** (plataforma DOS, *Windows* e *Linux*)**:** permitem esconder arquivos em imagens JPEG, existem várias ferramentas semelhantes. A característica dessa ferramenta é que sua eficácia depende da taxa de inserção, ou seja, quanto menor essa taxa, menor será a quantidade informação escondida, porém será ainda menos possível sua detecção. O que se tem é que a taxa ideal é abaixo de 5%, pois acima de 15% os efeitos começam a ficar visíveis dependendo do tipo de imagem.
- ***Mandelsteg*** **e** ***Gifextract*** (plataforma Unix)**:** essas ferramentas permitem esconder dados em imagens GIF, onde o mandelsteg cria a imagem armazenando dados em bits específicos e o *gifextract* pode ser utilizado para extrair esses dados. A *mandelsteg* conta com diversos comandos que possibilitam esteganografias diferentes, ou seja, comando que alteram suas configurações, definindo o grau de segurança, a quantidade de dados a ser armazenado, em quais áreas armazenar e a quantidade de cores na paleta que será utilizar no processo.
- ***MP3Stego*** (plataformas *Windows* 95/98/NT e Unix)**:** esconde informações em arquivos MP3 durante o processo de compressão, em que os dados são primeiramente criptografados, embutidos num arquivo WAV e depois escondidos no arquivo MP3.
- ***S-Tools*** (plataformas *Windows* 95/98/NT e Unix)**:** realize esteganografia em imagem e som, sendo possível esconder vários arquivos em um só, em que cada arquivo é comprimido individualmente ou não e escondido juntamente com seu nome. Os dados são criptografados utilizando a *passphrase* (frase-senha) que é inserida para gerar a *stego-key*.
- ***Sams Big Play Maker*** (plataforma *Windows* 95/98/NT)**:** esteganografa através da conversão de um texto qualquer em “peça de teatro”, com a utilização de uma lista de palavras e frases que substituem o caractere escondido.
- ***Secret Space*** (plataforma *Windows* 95/98/NT)**:** trabalha com esteganografia em textos, sendo que o texto que receberá a informação deve

ter entre 1.000 e 25.000 caracteres, dependendo do tamanho da informação. Sua principal característica é gravar, junto a informação escondida, a data de sua realização, possibilitando certo controle e segurança.

- ***Stella*** (plataformas que suportem Java)**:** possibilita esconder e extrair a mensagem com diferentes técnicas. Detecta a mensagem oculta ou uma assinatura, efetua a comparação da imagem original com a esteganografada e oferece informações estatísticas da imagem de capa.
- ***Wbstego4*** (plataforma *Windows* 95/98/NT)**:** é uma aplicação de 32 bits para Windows, que permite esconder qualquer tipo de arquivo bitmap (16, 256 ou 16.7 milhões de cores), arquivo de texto ASCII ou ANSI, arquivo H-TML e arquivo PDF. Utiliza a substituição dos últimos bits significativos da imagem.

Algumas técnicas que permitem esconder informações foram apresentadas num contexto atual, mas sabe-se que a esteganografia já é aplicada a tempos. A esteganografia é utilizada tanto para garantir a segurança da informação quanto na ação de hackers que pretendem se esconder através dela, ou seja, se utilizada de forma lícita, pode resultar em um recurso bem eficiente de segurança, porém essa segurança pode servir de forma ilícita, possibilitando esconder informações ilegais.

Como introdução a algumas das ferramentas utilizadas para a identificação da esteganografia que consistirá no estudo de caso que será a segunda parte desse trabalho, tem-se os seguintes exemplos:

- ***Stegdetect***:

Segundo Kipper (2004, p. 187), é uma ferramenta automática de detecção de esteganografia presente em imagens. É capaz de detectar diferentes técnicas esteganográficas aplicadas em imagens do tipo JPEG.

- ***Stegbreak***

Para Kipper (2004, p.187), *Stegbreak* é um programa que usa o dicionário de adivinhação para quebrar senhas de codificação. *Stegbreak* é usado para quebrar esteganografia utilizando o dicionário contra algumas ferramentas já citadas que atribuem a esteganografia a arquivos.

- ***Stego Watch***

Segundo Kipper (2004, p.187), *Stego Watch* é um serviço forense que monitora e analisa arquivos de imagens suspeitos de conter esteganografia. O projeto "*Steganography Detection and Recovery Toolkit*", tem o objetivo de desenvolver um conjunto de testes estatísticos capazes de detectar mensagens secretas em arquivos de computador e transmissões eletrônicas. Uma parte importante da pesquisa é o desenvolvimento dos métodos de detecção de esteganografia para algoritmos.

As funcionalidades e os resultados obtidos com a utilização dessas e de outras ferramentas para quebra de esteganografia serão estudados analiticamente de forma a apresentar seus prós e contras na aplicação em evidências que podem conter informações escondidas, durante a perícia forense computacional.

3 MATERIAIS E MÉTODOS

Durante o desenvolvimento do referencial teórico foram apresentadas diversas informações para garantir a segurança da informação nas organizações, bem como pessoas interessadas em quebrar essa segurança, por várias razões. Tais pessoas, conhecidas como hackers, em suas ações criminosas, também fazem uso das técnicas de segurança para que não sejam apanhados numa investigação.

A esteganografia pode estar presente dentre essas técnicas, como visto, existem diversas ferramentas que possibilitam a esteganografia, com diversas técnicas, voltados a diversos tipos de arquivos e que funcionam em diversas plataformas.

No desenvolvimento da parte prática desse trabalho o objetivo foi auxiliar a detecção de esteganografia em qualquer arquivo, tido como suspeito ou não, durante a perícia forense computacional, com a finalidade de tornar ainda mais abrangente a gama de arquivos que podem auxiliar na formulação do laudo pericial.

As ferramentas escolhidas para teste e simulação da esteganografia e da esteganálise, foram identificadas durante a pesquisa bibliográfica sem seguir um parâmetro de escolha, baseando-se somente em livros e artigos da internet que citaram com maior ênfase os seguintes softwares: *Camouflage*, *JPHide*, *Camouflage_Password_Finder*, *StegDetect, MP3Stego*.

3.1 RESULTADOS OBTIDOS

As pesquisas bibliográficas realizadas foram utilizadas para melhor compreensão do estudo, com o objetivo de proporcionar maior embasamento teórico para desenvolver a segunda fase do trabalho, que esteve focado em analisar as técnicas e ferramentas encontradas para realizar os demais objetivos da pesquisa, que tratam de identificar e analisar ferramentas para a quebra e realização da esteganografia durante a Perícia Forense Computacional. As mesmas serão descritas a seguir.

A ferramenta *Camouflage*, realiza com maior eficiência a camuflagem de informações através de imagens. A técnica também pode ser utilizada apenas com textos, porém com menos eficiência. A Figura 3 apresenta a tela inicial da ferramenta em questão.

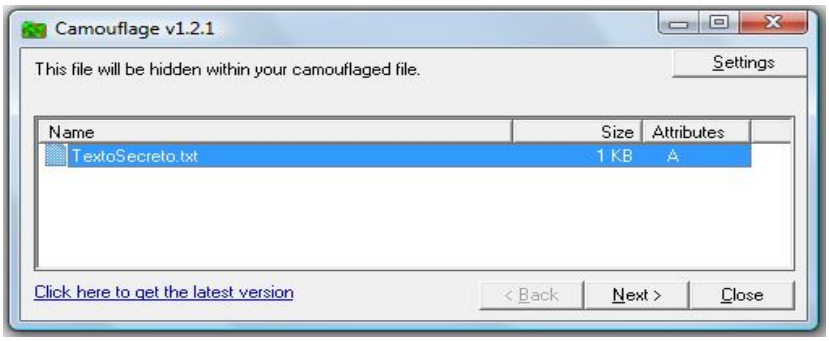

Figura 3- Tela inicial da ferramenta *Camouflage*.

Esta ferramenta, esta disponível para os seguintes sistemas operacionais: *Windows 95, 98, ME, NT, 2000, Millenium e XP*.

Para identificar se a ferramenta está instalada no computador, é necessário clicar com o botão direito do mouse, sobre o arquivo que se pretende esconder, assim abrirá as opções originais do *Windows*, porém com duas novas: *Camouflage* e *Uncamouflage*.

Escolhendo a opção *Camouflage*, abrirá uma tela (Figura 3) que mostra o tamanho e o atributo do arquivo escolhido. O próximo passo é escolher a imagem de capa, como exemplificado na Figura 4:

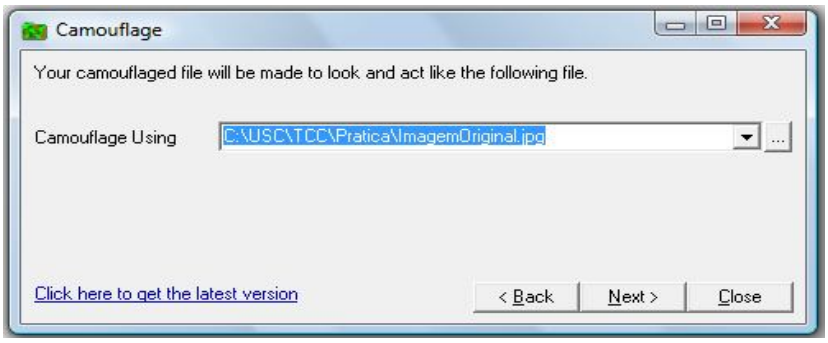

Figura 4- Escolha da imagem de capa - ferramenta *Camouflage*

Para não sobrepor a imagem original, é possível criar uma nova imagem resultante (arquivo secreto + imagem de capa), como mostra a Figura 5:

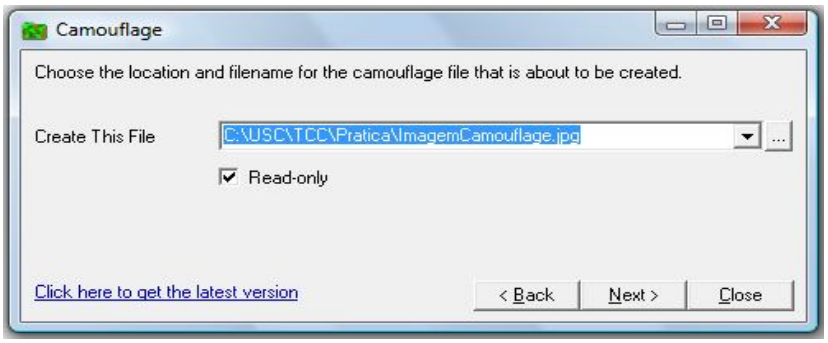

Figura 5- Definição do arquivo resultante - ferramenta *Camouflage*

Para garantir a segurança do arquivo resultante, é preciso definir uma senha, que permitirá futuro acesso ao arquivo secreto, porém a ferramenta não obriga o usuário a utilizar essa opção. Veja na Figura 6, a realização desse passo:

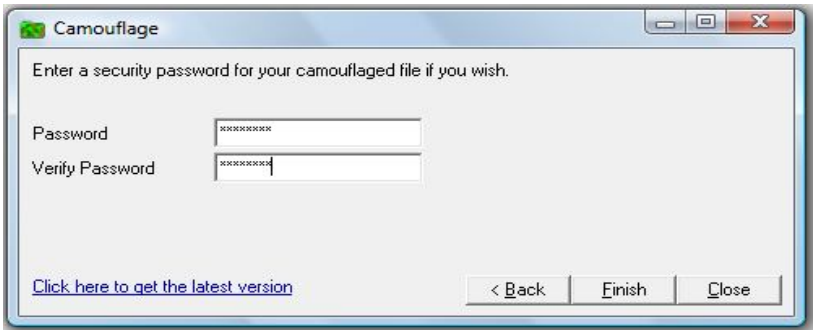

Figura 6- Definição da senha de acesso - ferramenta *Camouflage*

Assim, a realização da esteganografia está finalizada. Para visualizar o arquivo secreto é necessário clicar com o botão direito sobre o arquivo resultante e escolher a opção *Uncamouflage*, que solicitará a senha de acesso e apresentará os arquivos disponíveis dentro do escolhido. Veja na Figura 7, a realização desse passo:

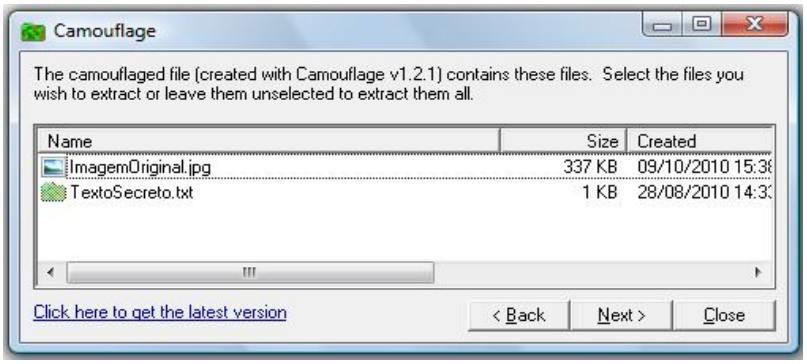

Figura 7- Visualização dos arquivos presentes no arquivo resultante - ferramenta *Camouflage*

Algumas configurações importantes dessa ferramenta podem ser alteradas através do botão "*Settings*" (Figura 3), dentre elas: a opção de visualizar ou não as opções *Camouflage* e *Uncamouflage*, dessa forma é possível disfarçar a presença da ferramenta no computador; é possível também apenas alterar o nome dessas duas opções, outra forma de mascarar a ferramenta.

Com essa ferramenta é possível realizar esteganografia através de arquivos textos, ou seja, com o arquivo texto como arquivo de capa. Na Figura 8 está um exemplo de arquivo secreto:

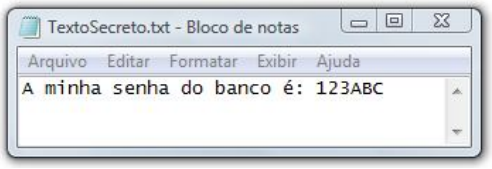

Figura 8- Texto secreto - ferramenta *Camouflage*

A imagem de capa, nesse caso sera um arquivo texto de capa, um tipo TXT, que está representado na Figura 9:

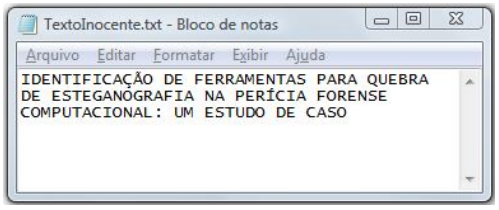

Figura 9- Arquivo TXT de capa - ferramenta *Camouflage*

O resultado final desse procedimento contém as o texto do arquivo de capa e alguns caracteres ilegíveis, onde estão presentes as informações do arquivo secreto e a senha para acessar a informação escondida. Veja o arquivo resultante na Figura 10:

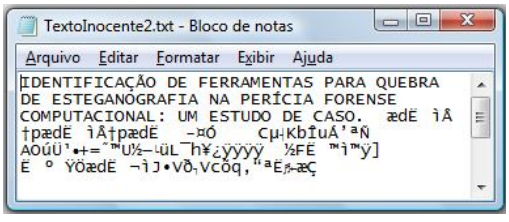

Figura 10- Arquivo resultante - ferramenta *Camouflage*

Como é visível na Figura 10, a camuflagem "Texto – Texto" não é muito eficiente, pois é possível visualizar caracteres estranhos no arquivo resultante, o que deixa explícito que pode haver algo estranho com esse arquivo durante a investigação pericial.

Outra forma de utilizar a ferramenta *Camouflage*, é a mais convencional, tendo um arquivo de imagem como capa, com a seguinte estrutura: texto secreto + imagem de capa = imagem esteganografada.

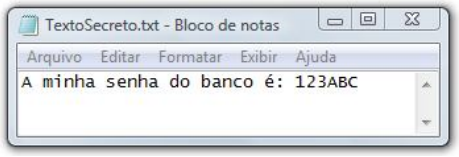

Figura 11- Texto secreto - ferramenta *Camouflage*

A Figura 11 está representando o arquivo que deverá ser mascarado, podendo ser um arquivo qualquer, como por exemplo: TXT, DOC, PDF, arquivos compactados (ZIP e RAR), dentre outros.

Figura 12- Imagem de capa - ferramenta *Camouflage*

A imagem de capa (Figura 12), assim como o arquivo a ser escondido, pode ser de qualquer tipo, até mesmo um executável, que mesmo depois de finalizado o procedimento da esteganografia, funciona normalmente.

A Figura 13 é o resultado final do procedimento, que neste caso mostra-se bastante eficiente, pois a olho nu não é possível apontar diferença entre a imagem de capa e a imagem resultante.

Figura 13- Imagem resultante - ferramenta *Camouflage*

Realizando análise desses arquivos, com a utilização do software *WinHex*, foi identificado que alguns arquivos de imagens são terminados sempre com a mesma combinação de caracteres, para JPEG os caracteres: “ÿÙ” (que em hexadecimal é correspondente à: FF D9) e para GIF os caracteres: “ ;” (que em hexadecimal é correspondente à: 00 3B).

A partir desses dados, foi observado que essa ferramenta trabalha com a adição das informações do texto secreto ao final do arquivo, ou seja, após os caracteres de identificação de final de arquivo, e por isso não causa modificações visíveis ao arquivo de capa, como no caso de se tratar de um arquivo executável, por exemplo, que não interfere na sua execução.

Quando se tem o arquivo original, é possível identificar que há diferença de tamanho, bastante relevante, entre o original e o arquivo resultante. Desta forma, com essas informações é possível identificar quando uma imagem contém mensagens além do seu verdadeiro conteúdo.

A combinação final de caracteres da imagem é seguida dos caracteres correspondentes ao texto secreto e a senha de acesso, juntamente com uma determinada repetição do hexadecimal “20” correspondente ao espaço em branco.

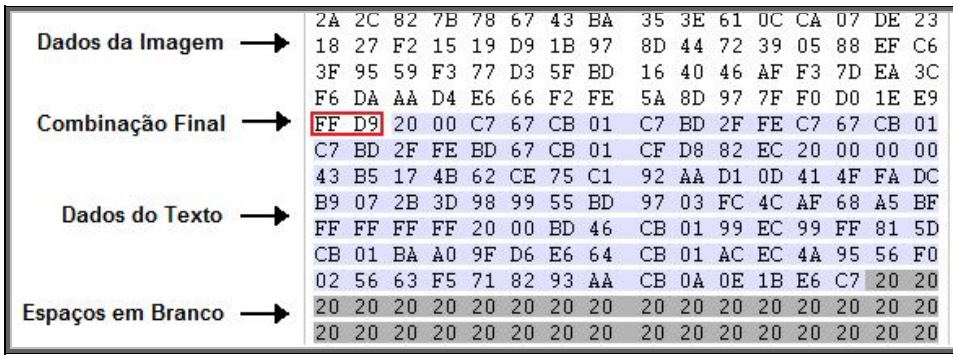

Figura 14- Trecho do arquivo resultante no formato hexadecimal - ferramenta *Camouflage*

Assim se segue a composição do arquivo camuflado, intercalando os espaços em branco e parte do texto secreto juntamente com a senha, sendo que esses dados são criptografados através da técnica de cifras de substituição, onde cada letra ou grupo de letras é substituído por outra letra ou grupo de letras.

A segunda ferramenta estudada é a *JPHide* que está disponível para os seguintes sistemas operacionais: DOS, *Windows* e *Linux*. Essa ferramenta está

disponível em uma versão gerada para o Windows, chamada de JPHS, que já contempla opção *Hide* e *Seek* que no DOS são executáveis diferentes (*jphide* e *jpseek*).

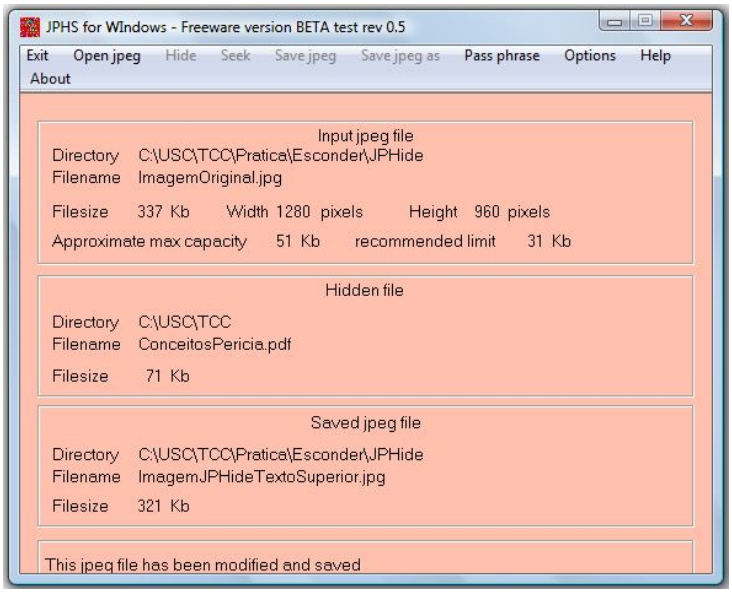

Figura 15- Tela inicial da ferramenta *JPHide*

A Figura 15 apresenta a tela inicial da ferramenta, que proporciona mais detalhes do processo, como por exemplo, a capacidade máxima aproximada que poderá ser utilizada pela esteganografia da mensagem a ser escondida, e o limite recomendado para que a alteração não seja perceptível a olho nu. Ao tentar esconder um arquivo com tamanho superior ao recomendado, a ferramenta apresenta um alerta, mas permite prosseguir.

As opções dessa ferramenta visual (*JPHS*) são as seguintes: "*Open jpeg*" onde o usuário escolhe a imagem de capa; "*Hide*" (que é habilitada em seguida) onde o usuário é obrigado a digitar uma senha, que uma vez digitada só poderá ser alterada na opção "*Pass phrase*", logo em seguida, deve ser escolhido o arquivo que será escondido; "*Save jpeg*" salva a imagem resultante com o mesmo nome da imagem de capa, substituindo-a; "*Save jpeg as*" possibilita criar uma imagem resultante, mantendo a original; e a opção "*Seek*" é utilizada para fazer o processo inverso, ou seja, tendo escolhida a imagem resultante de uma esteganografia, é solicita a senha e gerado um arquivo como aquele escolhido na opção "*Hide*".

A *JPHide* utiliza a criptografia de chave simétrica, que através de uma tabela de chaves substitui ou transpõe os dados originais.

A imagem resultante dessa ferramenta, normalmente tem tamanho inferior ou igual à imagem original, como por exemplo, uma imagem com 165 kb resulta em uma imagem de apenas 164 kb, mesmo contendo a imagem de capa, o texto secreto e a senha. A Figura 16 mostra uma imagem repleta de detalhes e a Figura 17 mostra a imagem resultante da esteganografia com um texto de tamanho superior ao indicado pela ferramenta:

Figura 16- Imagem original (165 kb) Limite recomendado: 15 kb – ferramenta *JPHide*

Figura 17- Imagem resultante (164 kb) Texto utilizado: 51 kb – ferramenta *JPHide*

Comparando visualmente essas imagens, não é possível determinar aspectos que possam diferenciá-las, mesmo tendo utilizado um texto com tamanho superior ao recomendado.

A Figura 17 permaneceu com a combinação de caracteres hexadecimais de fim de arquivo JPEG (FF D9), o que determina uma grande vantagem dessa técnica, pois se torna mais difícil de provar que a imagem tem alguma mensagem escondida, pois a ferramenta não altera a estrutura original da imagem. Uma imagem que tenha a taxa de inserção baixa, ou seja, limite recomendado baixo, e que não se tenha a imagem original, não é possível estabelecer de primeira instância, que a imagem está esteganografada. A Figura 18 mostra o trecho final do arquivo original e a Figura 19, o trecho final do arquivo resultante:

```
33 5A E7 CB 03 EF 58 B9  FC 58 C8 60 52 48 8B 44
C8 54 A3 7C 81 A6 A3 C5  EB AA C0 70 A3 90 9D 22
3C 80 58 03 F5 9C 6D 97  88 56 60 0D 3A 96 F5 23
E5 2B 1E 90 31 FF D9
```

Figura 18- Trecho do arquivo original – ferramenta JPHide

```
58 B9 FC 58 C8 60 52 48  8B 44 C8 54 A3 7C 81 A7
91 8B 4F 4A 58 2C 20 6C  1F 28 6E 80 25 80 FD 5E
36 C9 C4 67 29 8C B9 96  F1 23 E8 E9 6F 06 11 FF
D9
```

Figura 19- Trecho do arquivo esteganografado – ferramenta JPHide

Analisando esse aplicativo, têm-se algumas propriedades importantes a destacar, como por exemplo, a resolução horizontal e vertical foi alterada, como por exemplo, uma imagem original de 96 dpi resulta numa imagem de 72 dpi (dpi – pontos por polegada); informações particulares da origem da foto, como o fabricante da câmera, são perdidas ou criptografadas na imagem resultante.

A técnica dessa ferramenta é modificar alguns coeficientes DCT (Transformada discreta do cosseno) e a inserção LSB (*Bit* menos significativo), sendo que esses *bits* não são selecionados de forma contínua, o que dificulta a detecção da mensagem escondida. Essa modificação não se dá apenas aos *bits* menos significativos, mas também aos segundo *bits* menos significativos.

A seguir, as Figuras 20 e 21 demonstram os processos realizados nessa transformada:

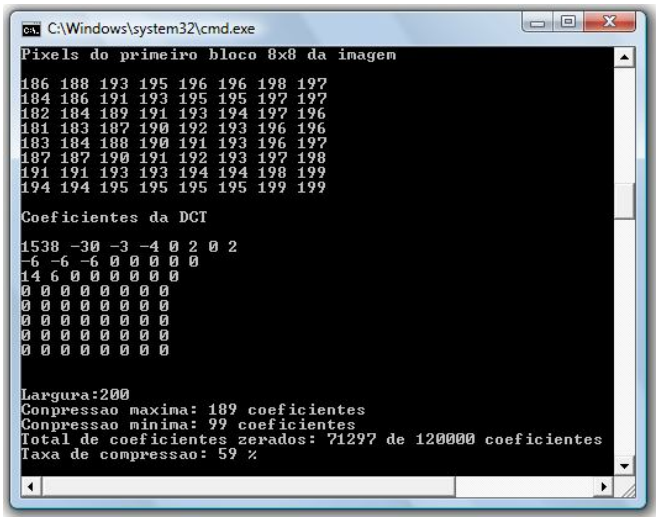

Figura 20- Representação passo a passo do processo da DCT - Programa JPEG *Compressor*
Fonte: Pozzer, s/d.

Na Figura 20, está representado passo a passo da DCT: a matriz de *pixels* da imagem, dividindo a mesma em blocos 8 x 8; seus coeficientes após a aplicação da transformada e a taxa de compressão. Esse tipo de compressão é utilizado pelo formato JPEG e a ferramenta *JPHide* utiliza o mesmo princípio.

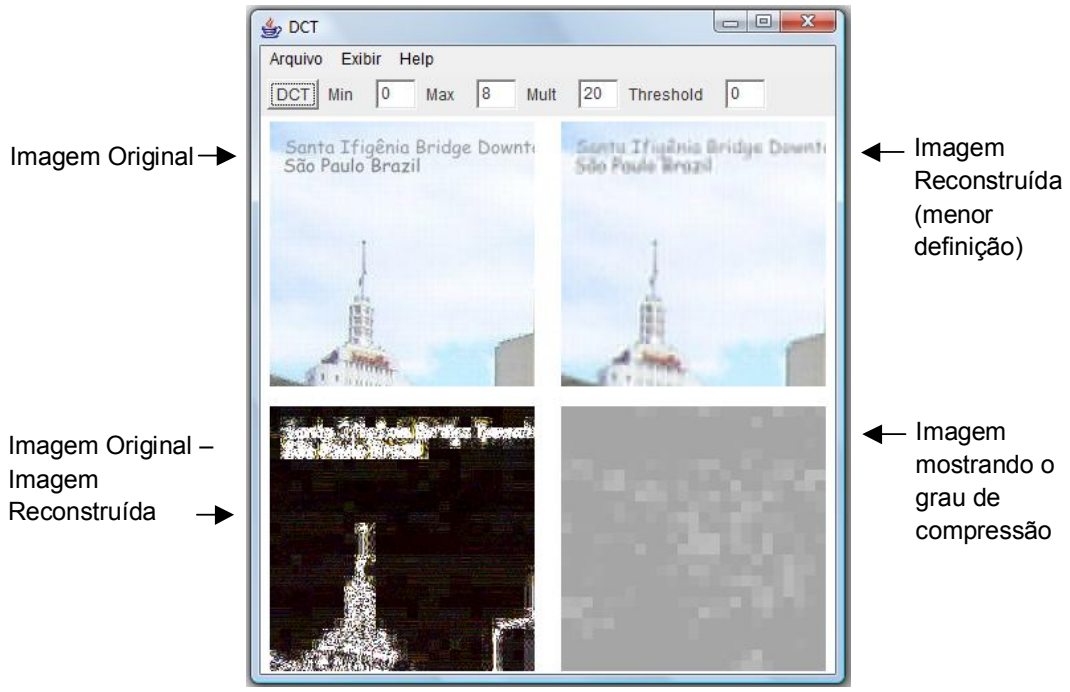

Figura 21- Representação visual do processo de DCT - Programa JPEG *Compressor*
Fonte: Pozzer, s/d.

Na Figura 21, está a representação visual desse processo de compressão, onde o resultado final mostra o grau de compressão da imagem. E como podem ser observadas, as regiões mais claras têm maior potencial de compressão.

Resumidamente a técnica DCT em imagens para esteganografia funciona da seguinte forma: primeiramente separa-se a imagem em blocos de 8x8 *pixels* e selecionando da esquerda para direita e de cima para baixo, aplica-se a fórmula DCT a cada bloco; em seguida cada bloco é comprimido através de uma tabela de quantização (discretiza uma informação contínua, que resulta em uma compressão com perdas); feito isso, é preciso calcular os *bits* menos significativo (LSB) de cada coeficiente DC e substituir cada LSB por cada *bit* da informação que se deseja esteganografar.

Por esse processo, é possível perceber que a *JPHide* é mais eficiente que a *Camouflage*, pois não se trata de uma simples adição das informações, mas sim de esconder informações aleatoriamente, e não em locais fixos; tornando sua descoberta uma tarefa mais criteriosa.

A ferramenta *Camouflage_Password_Finder*, também fez parte desse estudo, seu objetivo é extrair a senha utilizada para esconder informações através do

software Camouflage (já apresentado anteriormente). Assim, de posse da senha utilizada no processo, basta utilizar o *Camouflage* para obter as informações ocultas da imagem.

Como as informações estão criptografadas, é necessário saber a chave que foi utilizada para realizar o procedimento, e então descriptografar a senha. Como verificado no código fonte da ferramenta em questão, essa chave é composta por uma seqüência fixa de bytes, o que deixou claro que a mesma independe da senha escolhida, facilitando sua descoberta.

Portanto o objetivo dessa ferramenta é obter a senha utilizada no *Camouflage*. Veja na Figura 22 algumas análises:

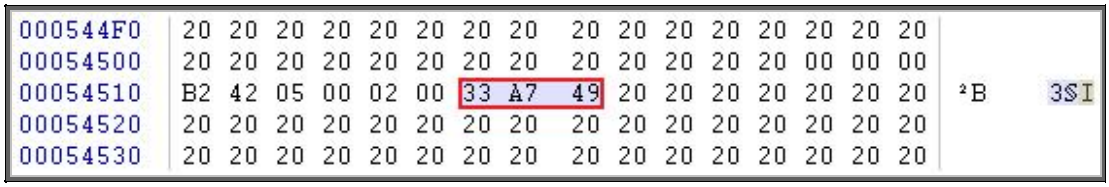

Figura 22- Trecho do arquivo esteganografado com senha - ferramenta *Camouflage_Password_Finder*

A Figura 22 contém um pequeno trecho da região onde se encontra os dados referentes à esteganografia. No destaque, está localizada a senha criptografada, que foi identificada através da comparação com a Figura 23, que também corresponde à mesma região, porém de um arquivo esteganografado sem senha.

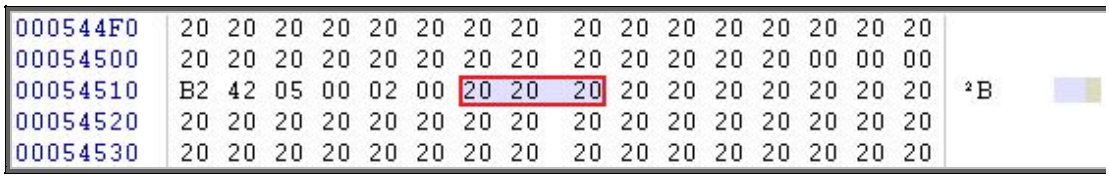

Figura 23- Trecho do arquivo esteganografado sem senha - ferramenta *Camouflage_Password_Finder*

Portanto, analisando as duas figuras fica fácil identificar onde foi alocada a senha. Mas na prática não será possível fazer essa comparação, pois o perito provavelmente não terá as duas imagens, porém, com os vários testes da ferramenta *Camouflage*, foi possível perceber que a senha encontra-se sempre ao final do arquivo, onde há uma divisão entre os dois últimos blocos de “20” da imagem.

A partir de então, a ferramenta realiza um XOR com os valores hexadecimais da senha, e prossegue realizando a comparação entre o byte atual e o valor hexadecimal 20H, verificando se os seguintes elementos são uma seqüência de 20H (correspondente ao espaço em branco), indicando que está no final da senha.

Veja na Figura 24 a senha encontrada no arquivo esteganografado e os valores correspondentes a chave da criptografia utilizada na ferramenta em questão:

Valores hexadecimais correspondentes a senha:	→	**33h**	**A7h**	**49h**
Valores hexadecimais correspondentes a chave da criptografia da ferramenta:	→	**02h**	**95h**	**7Ah**

Figura 24- Identificação da senha (valores hexadecimais) - ferramenta *Camouflage_Password_Finder*

A partir dessas informações, é necessário realizar a operação lógica do "Ou exclusivo - XOR", que utiliza como base a *"XOR TABLE (HEX)"* (Anexo 1). Veja na Figura 25 essa representação:

3	3		A	7	4	9
XOR	XOR		XOR	XOR	XOR	XOR
0	2		9	5	7	A
=	=		=	=	=	=
3	**1**		**3**	**2**	**3**	**3**

Senha criptografada: → **31h** **32h** **33h**

Figura 25- Resultado da operação XOR - ferramenta *Camouflage_Password_Finder*

Utilizando a tabela ASCII (Anexo 2), é possível converter os valores hexadecimais encontrados, em caracteres ASCII, que no exemplo, correspondem a seqüência de caracteres "123", que corresponde a senha utilizada para esteganografar.

Esse será o resultado apresentado pela ferramenta em questão. De posse dessa informação é possível identificar o arquivo escondido através da ferramenta

Camouflage, utilizando a opção *Uncamouflage*, que solicitará a senha e exibirá uma lista com os dois arquivos envolvidos nesse processo: a imagem de capa e o arquivo oculto.

Utilizando como suspeito um arquivo falso, ou seja, que não possua a aplicação da ferramenta *Camouflage*, o resultado será o mesmo apresentado na Figura 26:

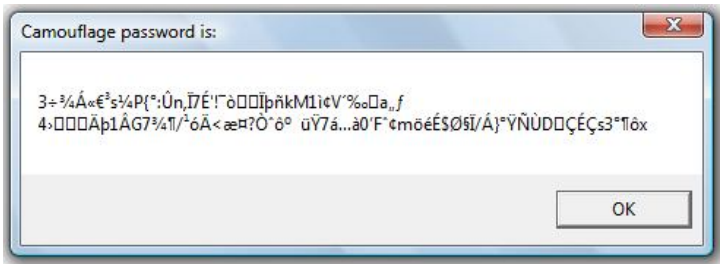

Figura 26- Resultado da opção *Uncamouflage* de um arquivo que não apresenta esteganografia

Dessa forma, analisando os resultados obtidos com essa ferramenta, é possível constatar que a técnica utilizada pelo software *Camouflage*, é bastante simples e primária, e por isso a facilidade em descobrir a senha.

Outra ferramenta estudada foi a *StegDetect* que tem como objetivo detectar a presença de esteganografia em arquivos, através de análises estatísticas, e também, quando possível, apontar qual software deu origem àquela esteganografia. Essa ferramenta se mostra eficiente, dentre as ferramentas utilizadas neste estudo, com a esteganografia da *JPHide*, que como já visto anteriormente, utiliza as técnicas DCT e LSB, assim como a *StegDetect*. E por isso é eficiente apenas em arquivos do tipo JPEG.

Escopo para utilização:

stegdetect [−nqV] [−s <float>] [−d <num>] [−t <tests>] [file.jpg ...]

Onde:

- -n → se utilizada, essa opção ignora as informações de cabeçalho contidas num arquivo JPEG;
- -q → apresenta um teste de todas as imagens JPEG do diretório, mostrando se contém, ou não, esteganografia e a ferramenta originária;
- -V → apresenta a versão da ferramenta *StegDetect*;

- -s <float> → propriedade importante dessa ferramenta, que é a manipulação do parâmetro de sensibilidade, ou seja, à medida que o valor <float> aumenta, o teste é mais sensível; caso não seja informado, será considerado o nível 1;
- -d <num> → apresenta o passo a passo do que esta sendo realizado;
- -t <tests> → avalia a presença da esteganografia realizada pelas ferramentas jsteg (j), outguess (o) e jphide (p).

Essas são as opções disponíveis no executável que deve ser utilizado pelo DOS, já a ferramenta visual tem apenas a opção da sensibilidade, como mostra a Figura 27:

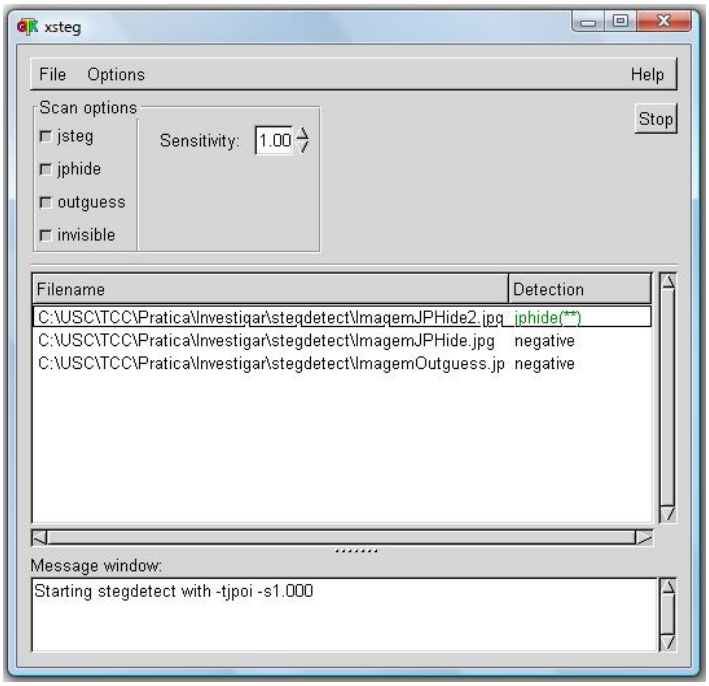

Figura 27- Tela inicial da ferramenta *StegDetect*

Foram testados três tipos de esteganografia, as três com a mesma imagem de capa, sendo que as duas últimas com textos bem menores que a imagem de capa, que apresentou resultado negativo no teste. E a primeira com tamanho acima de 14% do tamanho da imagem, apresentou resultado positivo e ainda conseguiu detectar a ferramenta que a originou (*JPHide*).

Para fazer essa detecção a ferramenta faz o processo similar à *JPHide*, porém com objetivo inverso: divide a imagem investigada, em blocos de dimensão 8x8 *pixels*; aplica a DCT e comprime (através da tabela de quantização) cada bloco,

calcula o *bit* menos significativo de cada coeficiente encontrado, recupera cada 8 *bits* e converte para um caractere da tabela ASCII (Anexo 2).

Acompanhando a ferramenta *StegDetect*, está a ferramenta *StegBreak*, que utiliza um dicionário de adivinhação para quebrar a senha utilizada na esteganografia. Baseando-se em vários testes, o resultado não garantiu que toda imagem detectada pela *StegDetect* pode apresentar sucesso na quebra da senha.

Outra ferramenta estuda, é a *MP3Stego*, que também utiliza a técnica da LSB, e está disponível para as plataformas *Windows* 95/98/NT e *Linux*.

Sua utilização parte de um arquivo do tipo *WAVE*, que juntamente com o texto a ser escondido, resulta em um arquivo tipo *MP3*, que consistem em um arquivo de trilhas sonoras compactado no formato *MPEG Audio Layer III*, o que potencializa o fator para esconder informações. Dessa forma, a ferramenta faz uma análise apurada, para saber se o arquivo *WAVE* em questão tem capacidade para servir como arquivo de capa, ou seja, resultar em um *MP3*.

Ao realizar a compressão MP3, que é o processo em que as informações do texto secreto serão escondidas, a ferramenta compacta e criptografa os dados e então às dispõem no fluxo de bits disponíveis no arquivo. Essa prática pode ser utilizada não apenas para esconder informações, mas também para marca os direitos autorais do áudio, sem haver percepção do ouvinte.

A ferramenta encontra-se disponível na versão gráfica e na versão DOS (linha de comando), sendo que nesta existem alguma opções que podem ser configuradas durante o processo de codificação. Veja a seguir:

- -b <*bitrate*> → taxa de compressão, se não informada será de 128 kbit;
- -c → define a utilização de *copyright*;
- -P <*text*> → define a senha;
- -E <*filename*> → nome do arquivo a ser escondido.

```
C:\USC\TCC\Pratica\Esconder\MP3Stego\MP3Stego>encode -b 256 -c -E
TextoSecreto.txt -P 123 som.wav som_stego.mp3
MP3StegoEncoder 1.1.17
See README file for copyright info
Microsoft RIFF, WAVE audio, PCM, mono 44100Hz 16bit, Length:  0: 0:20
MPEG-I layer III, mono  Psychoacoustic Model: AT&T
Bitrate=256 kbps  De-emphasis: none  CRC: off  (C)
Encoding "som.wav" to "som_stego.mp3"
Hiding "TextoSecreto.txt"
[Frame  791 of  791] (100.00%) Finished in 0: 0: 0
```

Figura 28- Exemplo de utilização: *Encode* - ferramenta *MP3Stego*

Na Figura 28, está um exemplo de utilização dessa ferramenta que foi realizado com sucesso, pois ao verificar o arquivo resultante, não foi possível perceber quaisquer alterações audíveis.

Essa ferramenta também tem a opção para decodificação, ou seja, o processo inverso, como mostra na Figura 29:

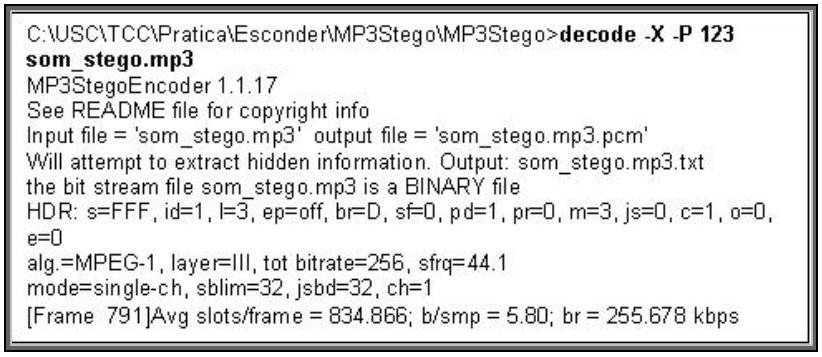

Figura 29- Exemplo de utilização: *Decode* - ferramenta *MP3Stego*

- -X → extrai o arquivo escondido;
- -P <*text*> → informa a senha para decodificação;
- <*filename*> → nome do arquivo esteganografado.

Essa utilização gera alguns arquivos temporários que fazem parte da decodificação, dentre esses arquivos gerados um é do tipo “TXT”, que contém parte legível e parte ilegível, como mostra a Figura 20:

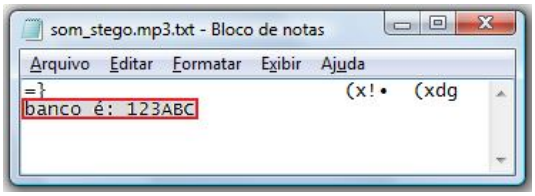

Figura 30- Arquivo resultante da decodificação - ferramenta *MP3Stego*

A região em destaque na Figura 30 mostra parte do texto esteganografado através do arquivo MP3. Sabendo-se que a mensagem original do arquivo é “A minha senha do banco é: 123ABC” é possível dizer que não são todos os arquivos resultantes dessa ferramenta que podem ser decodificados completamente. Porém o que foi atingido já poderia identificar alguma informação importante, como é o caso desse arquivo.

Ao efetuar a esteganografia com a ferramenta *MP3Stego*, observa-se que não é possível identificar qualquer ruído que possa diferenciar o arquivo resultante do original, o que demonstra a eficiência dessa técnica.

4 CONSIDERAÇÕES FINAIS

Esteganálise, como foi abordada, desempenha um papel muito importante na Perícia Forense Computacional, tratando de abrir caminhos e disponibilizar com maiores detalhes as informações presentes num dispositivo eletrônico envolvido na investigação.

Quanto às ferramentas estudadas, foi possível diagnosticar que em alguns casos as mesmas não foram completamente eficazes, tanto na esteganografia quanto na esteganálise. A ferramenta *Camouflage*, por exemplo, utiliza uma técnica básica, que não afeta o conteúdo da imagem de capa, ou seja, não causa qualquer alteração aparente, o que consequentemente torna sua esteganálise, relativamente simples, como foi apresentado anteriormente, e por isso os resultados obtidos com a ferramenta de esteganálise da *Camouflage*, a *Camouflage_Password_Finder*, foram plenamente satisfatórios.

As ferramentas que utilizam a técnica LSB, demonstram maior nível de dificuldade para alcançar o arquivo original e a senha utilizada, pois não deixa grandes vestígios do seu processo e trabalha de forma aleatória. Como por exemplo, no caso de um arquivo JPEG, que realizando esteganografia com essa técnica LSB, não altera a estrutura original da imagem de capa, e por isso analisando o arquivo resultante, não é possível afirmar com certeza que há informações escondidas no mesmo.

Utilizar arquivos de capa com maiores detalhes, ou seja, sem muitas regiões que tenham trechos contínuos ou repetidos, pode ser decisivo para o sucesso da esteganografia por conter diversas regiões em que a alteração não interfere no visual ou no áudio do arquivo original. Por exemplo, numa imagem de paisagem, com uma região de fundo, que normalmente retrata o céu com algumas nuvens, apresenta maiores regiões claras e com trechos contínuos e por isso uma maior compressão, portanto essa região é mais propícia a apresentar diferenças em relação à imagem original.

De modo geral, fica clara a idéia de que o perito forense computacional precisa ter conhecimento suficiente para analisar um arquivo nos seus aspectos mais específicos, a fim de identificar alterações em sua estrutura original, através de

ferramentas que possibilitem a visualização da estrutura interna do arquivo, como é o caso da ferramenta *WinHex*. A partir daí é necessário fazer uso de ferramentas que identifiquem a presença da esteganografia nos arquivos suspeitos, como por exemplo, a *StegDetect*, que atualmente procura identificar as principais ferramentas de esteganografia. Juntamente é possível utilizar ferramentas como a *StegBreak* que pode identificar a senha utilizada no processo.

Com base neste estudo, é possível identificar que as ferramentas estudadas precisam evoluir ainda mais para garantir uma maior quantidade de acertos, impedindo que arquivos realmente importantes passem despercebidos ou sejam classificados como inocente.

REFERÊNCIAS

ADAMS, D. E. **Digital Evidence: Standards and Principles**, 2000. Disponível em: <http://www.fbi.gov/hq/lab/fsc/backissu/april2000/swgde.htm>. Acesso em: Maio de 2010.

ASSUNÇÃO, M. F. A. **Guia do Hacker brasileiro**. [S.l.]: Visual Books, 2002, 139 p.

CARUSO, C. A. A.; STEFFEN, F. D. **Segurança em informática e de informações**. 2. ed. São Paulo: Editora SENAC São Paulo, 1999, 362 p.

CHESWICK, W. R.; BELLOVIN, S. M.; RUBIN, A. D. **Firewalls e Segurança na Internet:** Repelindo o hacker ardiloso. 2. ed. Porto Alegre: Bookman, 2005, 401 p.

COELHO, L. C. M.; BENTO, R. J. Comparações entre ferramentas de esteganografia. **Evidência digital magazine**, ano 1, n° 4, Out/Nov/Dez 2004. Disponível em: < http://www.guiatecnico.com.br/EvidenciaDigital>. Acesso em: 07 Maio 2010.

COSTA, M. A. S. L. **Computação forense:** Tratado de perícias criminalísticas. 2. ed. Campinas: Millennium, 2003, 264 p.

DEITEL, H. M. et al. **Perl Como Programar**. Porto Alegre: Bookman, 2002, 952 p.

FARMER, D.; VENEMA, W. Forensic Computer Analyses: An Introduction. **Dr. Dobb's Journal**, Inglaterra House Oakwood, 01 set. 2000. Disponível em: <http://www.drdobbs.com/184404242>. Acesso em: Maio de 2010.

FARMER, D.; VENEMA, W. **Perícia forense computacional:** Teoria e prática aplicada. Tradução Edson Furmankievicz, Carlos Schafranski, Docware Traduções Técnicas. São Paulo: Pearson Prentice Hall, 2007, 190 p.

FOUROUZAN, B. R. **Comunicação de dados e redes de computadores**. 3 ed. São Paulo: Bookman, 2004, 840 p.

FREITAS, A. R. de. **Perícia Forense aplicada à informática**. Rio de Janeiro: Brasport, 2006, 216 p.

JOHNSON, N.F. e JAJODIA, S. **Exploring Steganography:** Seeing the Unseen. IEEE Computer, 1998. Disponível em: < http://www.jjtc.com/pub/r2026.pdf>. Acesso em: Maio de 2010.

KATZAN JÚNIOR, H. **Segurança de dados em computação**. Tradução José Abe Royo dos Santos. Rio de Janeiro: Livros Técnicos e Científicos, 1977, 136 p.

KIPPER, G. **Investigator's guide to steganography**. [S.l.]: Auerbach, 2004, 221 p.

KUROSE, J. F.; ROSS, K. W. **Redes de computadores e a Internet:** Uma abordagem top-down. Tradução de Arlete Simille Marques; Revisão técnica Wagner Luiz Zucchi. 3. ed. São Paulo: Pearson Addison Wesley, 2006, 656 p.

LAUDON, K. C.; LAUDON, J. P. **Sistemas de informação gerenciais:** administrando a empresa digital. Tradução de Arlete Simille Marques; Revisão técnica de Erico Veras Marques, Belmiro João. 5. ed. São Paulo: Prentice Hall, 2004, 562 p.

LYRA, M. R. **Segurança e Auditoria em Sistemas de Informação**. Rio de Janeiro: Editora Ciência Moderna Ltda., 2008, 253 p.

McCLURE, Stuart; SCAMBRAY, Joel; KURTZ, George. **Hackers expostos:** Segredos e soluções para a segurança de redes. Tradução de Daniel Vieira. Rio de Janeiro: Campus, 2003, 782 p.

PARODI, L. **Manual das Fraudes**. 2. ed. Rio de Janeiro: Brasport, 2008, 412 p.

PHILIPP, A.; COWEN, D.; DAVIS, C. **Hacking Exposed Computer Forensics:** Secrets e Solutions. 2. ed. [S.l.]: McGraw-Hill, 2010, 518 p.

POZZER, C. T. Programa JPEG Compressor. Disponível em: <http://www-usr.inf.ufsm.br/~pozzer>. Acesso em: Out, 2010.

SCRIMGER, R. et al. **TCP/IP, a bíblia**. Tradução de Edson Furmankievicz, Docware Traduções Técnicas. Rio de Janeiro: Elsevier, 2002, 645 p.

SILES, R. Wireless Forensics: Tapping the Air - Part Two. **Security Articles**, 8 jan. 2007. Disponível em: <http://www.symantec.com/connect/pt-br/articles/wireless-forensics-tapping-air-part-two>. Acesso em: Maio, 2010.

SPITZNER, L. **Honeypots: tracking hackers**. Boston: Pearson Education Inc, 2003, 452 p.

STALLINGS, W. **Criptografia e segurança de redes:** Princípios e práticas. Tradução Daniel Vieira. 4. ed. São Paulo: Pearson Prentice Hall, 2008, 492 p.

STEWART, J. M.; TITTEL, E.; CHAPPLE, M. **CISSP: certified information system security professional:** Study guide. 4.ed. [S.l]: Sybex, 2008, 843 p.

TANENBAUM, A. S. **Redes de computadores**. Tradução de Vandenberg D. de Souza. 4. ed. Rio de Janeiro: Elsevier, 2003, 945 p.

ULBRICH, H. C.; VALLE, J. D. **Universidade Hacker**. 5. ed. São Paulo: Digeratti Books, 2006, 352 p.

BIBLIOGRAFIAS CONSULTADAS

COELHO, L. C. M.; BENTO, R. J. Ferramentas de esteganografia e seu uso na infowar. **Evidência digital magazine**, ano 1, n° 3, Jul/Ago/Set 2004. Disponível em: < http://www.guiatecnico.com.br/EvidenciaDigital>. Acesso em: 07 Maio 2010.

FARMER, D.; VENEMA, W. Being Prepared for Intrusion: An Introduction. **Dr. Dobb's Journal**, Inglaterra House Oakwood, 01 abr. 2001. Disponível em: <http://www.drdobbs.com/184404565;jsessionid=NLCCRU3FCAM0TQE1GHOSKHWATMY32JVN?queryText=Being+prepared+for+intrusion>
Acesso em: Maio de 2010.

MORAZ, E. **Treinamento profissional anti-hacker**. São Paulo: Digerati Books, 2007, 128 p.

ANEXOS

ANEXO A – Tabela XOR (Hexadecimal)

XOR TABLE (HEX)

XOR	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
1	1	0	3	2	5	4	7	6	9	8	B	A	D	C	F	E
2	2	3	0	1	6	7	4	5	A	B	8	9	E	F	C	D
3	3	2	1	0	7	6	5	4	B	A	9	8	F	E	D	C
4	4	5	6	7	0	1	2	3	C	D	E	F	8	9	A	B
5	5	4	7	6	1	0	3	2	D	C	F	E	9	8	B	A
6	6	7	4	5	2	3	0	1	E	F	C	D	A	B	8	9
7	7	6	5	4	3	2	1	0	F	E	D	C	B	A	9	8
8	8	9	A	B	C	D	E	F	0	1	2	3	4	5	6	7
9	9	8	B	A	D	C	F	E	1	0	3	2	5	4	7	6
A	A	B	8	9	E	F	C	D	2	3	0	1	6	7	4	5
B	B	A	9	8	F	E	D	C	3	2	1	0	7	6	5	4
C	C	D	E	F	8	9	A	B	4	5	6	7	0	1	2	3
D	D	C	F	E	9	8	B	A	5	4	7	6	1	0	3	2
E	E	F	C	D	A	B	8	9	6	7	4	5	2	3	0	1
F	F	E	D	C	B	A	9	8	7	6	5	4	3	2	1	0

Identities:

x ¤ 0 = x (i.e., 0 ¤ 0 = 0, 1 ¤ 0 = 1)
x ¤ 1 = ¬x (i.e., 0 ¤ 1 = 1, 1 ¤ 1 = 0)
x ¤ x = 0 (i.e., 0 ¤ 0 = 0, 1 ¤ 1 = 0)

Fonte: http://www.garykessler.net/library/byte_logic_table.html

ANEXO B – Tabela ASCII

Dec	Hx	Oct	Char	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr	Dec	Hx	Oct	Html	Chr
0	0	000	NUL (null)	32	20	040		Space	64	40	100	@	@	96	60	140	`	`
1	1	001	SOH (start of heading)	33	21	041	!	!	65	41	101	A	A	97	61	141	a	a
2	2	002	STX (start of text)	34	22	042	"	"	66	42	102	B	B	98	62	142	b	b
3	3	003	ETX (end of text)	35	23	043	#	#	67	43	103	C	C	99	63	143	c	c
4	4	004	EOT (end of transmission)	36	24	044	$	$	68	44	104	D	D	100	64	144	d	d
5	5	005	ENQ (enquiry)	37	25	045	%	%	69	45	105	E	E	101	65	145	e	e
6	6	006	ACK (acknowledge)	38	26	046	&	&	70	46	106	F	F	102	66	146	f	f
7	7	007	BEL (bell)	39	27	047	'	'	71	47	107	G	G	103	67	147	g	g
8	8	010	BS (backspace)	40	28	050	(	(	72	48	110	H	H	104	68	150	h	h
9	9	011	TAB (horizontal tab)	41	29	051	)	)	73	49	111	I	I	105	69	151	i	i
10	A	012	LF (NL line feed, new line)	42	2A	052	*	*	74	4A	112	J	J	106	6A	152	j	j
11	B	013	VT (vertical tab)	43	2B	053	+	+	75	4B	113	K	K	107	6B	153	k	k
12	C	014	FF (NP form feed, new page)	44	2C	054	,	,	76	4C	114	L	L	108	6C	154	l	l
13	D	015	CR (carriage return)	45	2D	055	-	-	77	4D	115	M	M	109	6D	155	m	m
14	E	016	SO (shift out)	46	2E	056	.	.	78	4E	116	N	N	110	6E	156	n	n
15	F	017	SI (shift in)	47	2F	057	/	/	79	4F	117	O	O	111	6F	157	o	o
16	10	020	DLE (data link escape)	48	30	060	0	0	80	50	120	P	P	112	70	160	p	p
17	11	021	DC1 (device control 1)	49	31	061	1	1	81	51	121	Q	Q	113	71	161	q	q
18	12	022	DC2 (device control 2)	50	32	062	2	2	82	52	122	R	R	114	72	162	r	r
19	13	023	DC3 (device control 3)	51	33	063	3	3	83	53	123	S	S	115	73	163	s	s
20	14	024	DC4 (device control 4)	52	34	064	4	4	84	54	124	T	T	116	74	164	t	t
21	15	025	NAK (negative acknowledge)	53	35	065	5	5	85	55	125	U	U	117	75	165	u	u
22	16	026	SYN (synchronous idle)	54	36	066	6	6	86	56	126	V	V	118	76	166	v	v
23	17	027	ETB (end of trans. block)	55	37	067	7	7	87	57	127	W	W	119	77	167	w	w
24	18	030	CAN (cancel)	56	38	070	8	8	88	58	130	X	X	120	78	170	x	x
25	19	031	EM (end of medium)	57	39	071	9	9	89	59	131	Y	Y	121	79	171	y	y
26	1A	032	SUB (substitute)	58	3A	072	:	:	90	5A	132	Z	Z	122	7A	172	z	z
27	1B	033	ESC (escape)	59	3B	073	;	;	91	5B	133	[	[	123	7B	173	{	{
28	1C	034	FS (file separator)	60	3C	074	<	<	92	5C	134	\	\	124	7C	174			|
29	1D	035	GS (group separator)	61	3D	075	=	=	93	5D	135	]	]	125	7D	175	}	}
30	1E	036	RS (record separator)	62	3E	076	>	>	94	5E	136	^	^	126	7E	176	~	~
31	1F	037	US (unit separator)	63	3F	077	?	?	95	5F	137	_	_	127	7F	177	&#127;	DEL

Fonte: http://www.virose.pt/ml/blogs/a2m/?p=471