Resposta a Incidentes para Ambientes Corporativos Baseados em Windows

Flávio de Souza Oliveira
Instituto de Computação
Universidade Estadual de Campinas
13083-970 Campinas - SP
flavio.oliveira@ic.unicamp.br

Célio Cardoso Guimarães
Instituto de Computação
Universidade Estadual de Campinas
13083-970 Campinas - SP
celio@ic.unicamp.br

Paulo Lício de Geus
Instituto de Computação
Universidade Estadual de Campinas
13083-970 Campinas - SP
paulo@ic.unicamp.br

Resumo: *Os prejuízos causados por um incidente de segurança podem ser desastrosos para uma organização, desta forma, é vital que se crie políticas para minimizar as perdas durante esses episódios. A elaboração de procedimentos a serem aplicados durante tal situação é essencial para o restabelecimento das atividades normais da instituição, buscando assim, minimizar os prejuízos sofridos.*

1. Introdução

Em virtude do seu crescimento, atualmente pode-se afirmar que a preocupação com segurança é requisito essencial para a maioria das aplicações em rede. Um bom paralelo foi feito por Jeffrey J. Carpenter, engenheiro de segurança senior do CERT/CC (*Computer Emergency Response Team/Cordination Center*): "A história da segurança na Internet pode ser comparada à vida em uma cidade. Quando a cidade é pequena, as pessoas se conhecem e confiam umas nas outras, de modo que janelas e portas podem ser deixadas abertas (...). Contudo, quando a cidade cresce, crimes e segurança tornam-se preocupações mais comuns. A Internet pode hoje então ser comparada a uma metrópole, onde as portas e janelas devem permanecer fechadas a maioria do tempo"[1].

O problema é que mesmo tomando-se todas as medidas necessárias, falhas de segurança podem ocorrer, uma vez que alguma vulnerabilidade ainda não divulgada pode ser explorada ou um novo tipo de ataque pode ser utilizado. Dessa forma, não se pode afirmar que um dado aparato de segurança está isento de falhas. Isto se deve principalmente ao fato de que tais aparatos, bem como os serviços oferecidos através da Internet, são compostos por inúmeras peças de software, que por sua vez, possuem milhares de linhas de código que não estão imunes a erros de programação.

Tendo em vista que não há esquema de segurança imune a falhas, torna-se então necessário o estabelecimento de uma metodologia a ser adotada no caso de um ataque bem sucedido, além da presença de pessoal capaz de executar tais procedimentos (*Response Team*). No entanto, a preocupação com tal metodologia ainda é muito pequena dentro das organizações conectadas à Rede, por outro lado, a inexistência desses dois personagens pode causar a inviabilização de uma possível ação judicial contra o atacante, além do possível agravamento dos prejuízos financeiros.

Nesse artigo discute-se a instalação de um plano de resposta a incidente em uma corporação, buscando apresentar os principais pontos técnicos a serem abordados. Além disso, apresenta-se toda a problemática envolvida na análise de um sistema Windows 2000/NT sem o seu prévio desligamento, sistema esse largamente utilizado mas carente de tal abordagem.

2. Forense Computacional

A abordagem do termo forense remete automaticamente ao meio policial, onde na tentativa de solucionar um mistério, policiais e peritos devem analisar minuciosamente todo tipo de objetos, sinais e marcas que estejam presentes na cena do crime.

A análise forense inicia imediatamente após a chegada dos policiais ao local, começando pelo isolamento eficiente do perímetro, evitando assim, a exposição excessiva e possível contaminação das evidências. Passa-se então para a fase de identificação e coleta de todo tipo de dado e material que possa ter alguma relevância na resolução do caso em questão. Apenas após a realização dessas duas primeiras etapas inicia-se uma análise laboratorial das possíveis evidências, tais como: análise balística e de DNA. Tal fato demonstra o erro de muitas pessoas ao considerar apenas a análise laboratorial como análise forense.[6]

O sucesso da análise está então ligado diretamente ao sucesso de todas as três etapas que constituem o processo. Caso haja contaminação ou falhas na aplicação de metodologias para aquisição ou manipulação de evidências, tem-se por sua vez, uma grande possibilidade de não se conseguir resultados precisos, ou mesmo resultado algum durante uma análise laboratorial, isto avaliando apenas aspectos técnicos, uma vez que falhas desse tipo invalidam completamente uma evidência em um tribunal.

As técnicas envolvidas na análise laboratorial de uma evidência são divididas em etapas que dependem diretamente do tipo de material que está sendo analisado, o que pode variar desde um cadáver a uma minúscula mancha de tinta. Tome-se por exemplo, a análise feita no DNA recolhido de uma amostra de sangue na cena de um crime. É possível aplicar exatamente o mesmo protocolo a toda amostra de DNA recebida: eliminam-se as impurezas e reduz-se-o à sua forma elementar [8]. Todos estes procedimentos devem ser padronizados, gerar resultados reproduzíveis e serem aceitos pela comunidade científica internacional.

Com o advento do computador e o surgimento dos primeiros casos envolvendo o meio computacional, tornou-se necessária a criação de uma nova disciplina forense que deveria preocupar-se em atuar nesse novo nicho, criando metodologias e acumulando conhecimentos para a aquisição, manipulação e análise de evidências digitais.

A resolução de um mistério computacional pode ser uma tarefa árdua e difícil. É necessário que se examine o sistema minuciosamente, assim como um detetive examina a cena de um crime [3]. Para isso a pessoa que está realizando a análise deve conhecer profundamente o sistema operacional em que está trabalhando, podendo então identificar e entender as relações de causa e efeito de todas as ações tomadas durante a análise.

As relações de causa e efeito não são suficientes; contudo, existe ainda a necessidade de mais uma série de habilidades para que um profissional possa conduzir uma análise forense de maneira eficaz. Segundo Venama e Farmer [3], felizmente muitas dessas habilidades são características aos programadores, tais como: raciocínio lógico e possuir uma mente aberta. Tais habilidades são largamente utilizadas durante a busca de um erro em um programa. Entretanto, a depuração de um programa ainda está distante do desafio representado por uma análise forense, já que ao se depurar um programa está se lutando contra si mesmo, enquanto em uma análise forense enfrenta-se outro programador que não tem o interesse de ser descoberto [3].

2.1. Manipulação de Evidências Digitais

Nessa seção apresenta-se um panorama do atual estágio dos esforços de padronização da forense computacional. A importância de tal esforço reside na necessidade de se garantir a integridade das evidências apresentadas em um tribunal, dado que, uma vez padronizados os procedimentos, torna-se juridicamente inviável o questionamento dos fatos apresentados

tomando como tese a metodologia utilizada na manipulação das provas, desde que esta tenha sido aplicada corretamente.

Apesar de haver diversos trabalhos na área, ainda nota-se uma certa escassez de metodologias para o manuseio desse tipo de evidência. Tal carência pode ser explicada pelo fato de existirem inúmeras mídias e sistemas operacionais, além de diversas mudanças de versão. Todos esses fatores tornam difícil a definição de padrões e metodologias, pelo menos da forma como acontece com as outras disciplinas forenses [9].

Atualmente já existem padrões internacionais definidos e sendo aplicados de forma experimental [8]. Eles foram desenvolvidos pelo SWGDE (*Scientific Working Group on Digital Evidence*), que é o representante norte-americano na *International Organization on Computer Evidence* (IOCE). Tais padrões foram apresentados durante a *International Hi-Tech Crime and Forensics Conference* (IHCFC), realizada em Londres, de 4 a 7 de outubro de 1999.

Os padrões desenvolvidos pelo SWGDE seguem um único princípio: o de que todas as organizações que lidam com a investigação forense devem manter um alto nível de qualidade a fim de assegurar a confiabilidade e a precisão das evidências. Esse nível de qualidade pode ser atingido através da elaboração de SOPs (*Standard Operating Procedures*), que devem conter os procedimentos para todo tipo de análise conhecida e prever a utilização de técnicas, equipamentos e materiais largamente aceitos na comunidade científica internacional [8].

No Brasil ainda não existe padronização em andamento, apenas trabalhos feitos a pedido da polícia federal, trabalhos esses direcionado para público leigo composto por promotores e juizes federais.

3. Resposta a Incidentes

Quando uma organização possui um plano de Resposta a Incidentes, atividades anômalas são mais facilmente detectadas, e a adoção de uma metodologia apropriada pode rapidamente identificar os sistemas afetados, podendo-se então levantar a extensão do incidente [10]. Tal levantamento é essencial para que se possa reportar o evento aos órgãos competentes, bem como dimensionar o montante do prejuízo. Mais detalhes sobre como se reportar um incidente serão abordados na seção 3.4.

Durante o trato com incidentes reais, nota-se a grande dificuldade de se estabelecer uma política que englobe todas as situações, já que durante eventos dessa natureza lida-se com agentes cujos objetivos são a princípio desconhecidos. Tal deficiência pode ser minimizada através do contínuo aprimoramento da metodologia por meio de simulações periódicas. Embora exista essa dificuldade, segundo Ed deHart do CERT/CC, “Tudo que um *site* puder fazer para se preparar para um incidente será benéfico, podendo economizar tempo e dinheiro” [10].

3.1. Etapa Pré-Incidente

A preparação para a resposta a um incidente de segurança começa pela criação de um time de resposta (TR), uma vez que geralmente para se executar os procedimentos necessários em tempo, a organização deve possuir pessoal preparado no local [5]. O TR pode ser formado por um ou vários membros com conhecimento na área de segurança, o número depende do tamanho e das necessidades da corporação.

O primeiro objetivo do TR é a criação do plano de resposta a incidente. Não existe uma receita para a criação de tal documento, pois este deve se adequar às peculiaridades de cada organização e mesmo após tal adaptação requer aprimoramento contínuo, como já dito anteriormente. Porém, é possível estabelecer pontos-chave que na maioria das vezes devem estar presentes. Dentre eles pode-se citar:

- Identificação de Prioridades: do ponto de vista corporativo, um bom tratamento dado a um incidente seria aquele que mais amenizasse o impacto nos negócios da companhia. A metodologia deve considerar as prioridades da empresa, bem como saber identificar dentre os possíveis danos, quais seriam mais prejudiciais à organização. Dentre eles pode-se citar:
  1. Comprometimento da reputação da empresa; por exemplo através da desfiguração de páginas web.
  2. Roubo de propriedade intelectual;
  3. Modificação ou destruição dos bancos de dados da organização.

  Tal identificação serve também para amenizar o choque de objetivos entre o investigador forense, que naturalmente busca remediar o acesso ilegal de forma concomitante à identificação e neutralização do agente causador do incidente, e a empresa que deseja o retorno da normalidade dos negócios o mais rápido possível [10]. O ideal seria o restabelecimento das atividades normais, manipulando as possíveis evidências de forma a garantir sua integridade (e.g. hash MD5), não interferindo assim em futura busca ao agente causador do incidente;
- Política de utilização de recursos: um plano de resposta a incidentes passa também, pela elaboração de uma política de utilização dos recursos de tecnologia da informação que contemple a possibilidade de uma investigação, abordando assim, questões como: quebra de privacidade e monitoramento de atividades. A definição de tal política requer debate entre todos os usuários e discussão da filosofia da organização, o que por muitas vezes não é uma tarefa fácil. Para ajudar nessa etapa, existem diversos modelos disponíveis na Internet, um exemplo seria os presentes no site da SANS:
  http://www.sans.org/newlook/resources/policies/policies.htm;
- Preparação das máquinas: para facilitar uma futura análise, seria interessante que todas as máquinas da rede estivessem previamente configuradas para fornecer a maior quantidade de informações possível ao TR. Por exemplo:
  1. Habilitar as políticas de auditoria do Windows de forma conveniente. Uma ferramenta que poderia ser utilizada para facilitar tal processo é o DoIt4Me[1], ferramenta gratuita para automatização de tarefas administrativas no Windows. [1]
  2. Armazenar o hash dos arquivos críticos do sistema em local seguro. Ferramentas como o Tripwire automatizam este processo, contudo tal ferramenta não é gratuita para ambientes corporativos. Uma possível solução seria a utilização do *framework* desenvolvido em perl por Harlan Carvey e apresentado em [3].
- Kit básico de resposta: ver seção 3.2.

3.2. Kit Básico de Resposta

A reunião de um conjunto de ferramentas que possa diagnosticar o estado atual de uma máquina e garantir a integridade das informações apresentadas, é um dos pontos mais importantes na preparação para um incidente. Quando uma máquina é identificada como alvo de um ataque ou suspeita de uso ilegal por parte de algum agente interno, não é seguro utilizar qualquer tipo de software ou biblioteca presente na máquina. Uma solução muito adotada é a cópia em CD de todos os programas necessários, sejam eles especializados ou nativos, para neutralizar a tentativa de ocultar informações através da utilização de bibliotecas e comandos adulterados.

1. http://www.ic.unicamp.br/~ra990866/doit4me/welcome.html

A análise de um sistema Windows sem o seu prévio desligamento (*live analysis*) é crucial para a investigação, uma vez que tem-se a oportunidade de coletar dados que não estarão mais disponíveis após um possível religamento. A tais informações da-se o adjetivo de voláteis, pelo fato de desaparecerem facilmente com o passar do tempo ou com a reinicialização da máquina.

A *live analysis* ainda representa um grande problema, isto porque a maioria das ferramentas para a plataforma Windows utiliza bibliotecas dinâmicas, enquanto o ideal seria que todas fossem compiladas estaticamente para que nenhum código presente na máquina analisada seja executado. Contudo a indisponibilidade dos códigos fonte, seja de ferramentas nativas ou de ferramentas especializadas, característica dessa plataforma, inviabiliza tal procedimento. Uma solução muito adotada é a inclusão de todas as DLL's (*Dynamic Link Libraries*) necessárias no CD de ferramentas supracitado, uma vez que a plataforma em questão procura pelas bibliotecas no diretório corrente em primeiro lugar. No entanto, várias DLL's já estão carregadas na memória e no cache do sistema no início da análise, o que inibe a busca pelas DLL's contidas no CD. Este assunto ainda está em discussão, uma vez que o W2k não possui mecanismos para remoção de DLL's da memória, sem que a estabilidade do sistema e das aplicações seja comprometida.

A escolha das ferramentas que vão estar inclusas no CD também merece atenção, deve-se procurar por programas que forneçam resultados relevantes causando o mínimo de distorção possível no sistema analisado. O conjunto de ferramentas varia de acordo com as necessidades do TR de cada organização, no entanto pode-se identificar algumas que devem ser considerados, dentre as quais:

- auditpol: Informa as políticas de auditoria do sistema - NTRK (*NT Resource Kit*);
- cmd: Interpretador de comandos do Windows;
- dumpel: Cria um arquivo texto com os eventos do *event log*. Este programa é uma alternativa à execução de uma cópia simples dos arquivos de *log*, uma vez que a descrição dos eventos é na maioria das vezes gerada dinamicamente através de uma consulta ao seu causador, inviabilizando assim a consulta *offline* de seu conteúdo - NTRK;
- findcom.pl: ver seção 3.3.
- fport: Indica em que portas os atuais processos estão escutando. URL: http://www.foundstone.com/knowledge/zips/FPortNG.zip
- GNU Unix Utils for Win32: Versão para Windows de diversas ferramentas GNU, dentre elas o *dd* e o *md5sum*. A vantagem dessas ferramentas é o fato de necessitarem apenas da biblioteca msvcrt.dll. Endereço: ftp://ftp.uni-koeln.de/pc/win32/misc/unxutils.zip;
- handle: Exibe todos os *handles* de determinado processo, permitindo descobrir por exemplo quais arquivos ou chaves do registro estão sendo utilizadas por ele. URL: http://www.sysinternals.com/files/handle.zip
- ipconfig: Ferramenta nativa que manipula configurações de rede;
- listdlls: Lista todas as DLL's carregadas por determinado processo. URL: http://www.sysinternals.com/files/listdlls.exe
- nbtstat: Ferramenta nativa que lista estatísticas das conexões em atividade utilizando NetBIOS;
- net: Ferramenta nativa do Windows que manipula compartilhamentos, dentre outras diversas funcionalidades;
- netstat: Ferramenta nativa do Windows que indica todas as conexões TCP/IP ativas;

- ntlast: Linha de comando que exibe os eventos relacionados ao *login* dos usuários; para isso eles devem estar sendo auditados. URL: http://www.foundstone.com/knowledge/zips/ntlast30.zip
- pslist: Lista os processos em execução na máquina. URL: http://www.sysinternals.com/files/pslist.zip
- psloggedon: Exibe os usuários logados. URL: http://www.sysinternals.com/files/PsLoggedOn.zip
- pwdump: Copia a base de dados do SAM (*Security Account Manager*), para que as senhas possam ser submetidas a uma tentativa de quebra. URL: http://www.webspan.net/~tas/pwdump2/pwdump2.zip
- rasusers: Lista todas as contas de usuários de um domínio ou servidor que têm permissão para acessar servidores remotamente através de *dial-up* - NTRK;
- reg: Exibe o conteúdo de determinada chave do registro - NTRK;
- regdump: Cria arquivo texto contendo todo o conteúdo do registro - NTRK;
- rmtshare: Lista, adiciona e remove compartilhamentos remotamente - NTRK;
- route: Ferramenta nativa que manipula a tabela de roteamento;
- startup.pl: Vasculha a pasta *startup* e as chaves apropriadas do registro e informa quais programas que são executados quando o sistema inicia ou algum usuário executa o *login*. URL: http://patriot.net/~carvdawg/scripts/startup.pl
- strings: Recupera todas as cadeias de caracteres contidas em um arquivo executável. Muito útil durante a identificação da função de um dado software. URL: http://www.sysinternals.com/files/strings.zip;

3.3. FindCom.pl

O FindCom.pl é um *script* perl que foi desenvolvido com o intuito de auxiliar a análise forense de máquinas Windows, no entanto também pode ser utilizado como uma ferramenta de auditoria. A funcionalidade central do programa é efetuar busca por padrões nos campos de controle de versão (figura 1), presentes nos executáveis e nas DLL's, em um dado diretório.

É de conhecimento geral que diversos processos de desinstalação, tanto manual quanto automatizado, mantém DLL's e chaves no registro por questões de compartilhamento ou manutenção de preferências. Entretanto o programa regedit.exe possui mecanismos de busca que podem revelar traços de uma possível instalação. No caso das DLL's não havia uma maneira simples e automatizada de tentar fazer tal confirmação. Apesar de ser pequena, a possibilidade de haver DLL's com comentários esquecidas em alguma parte da máquina existe e pode servir para confirmar alguma teoria pré-estabelecida pelo TR. Note também que o TR não lida apenas com casos de invasão externa, mas também eventos internos em desacordo com a política de uso pré-estabelecida (e.g. instalação ilegal de software), além disso, nomes de arquivos podem ser facilmente alterados, o que já não ocorre com as informações do controle de versão. Uma versão preliminar desse *script* pode ser obtida no endereço: http://www.ic.unicamp.br/~ra000494.

3.4. Pós-Incidente

Quando um incidente ocorre, é crucial que se evite o pânico para que o plano de resposta previamente elaborado seja seguido com exatidão. Nessa nova etapa, novamente não há receita, uma vez que vários imprevistos podem ocorrer; entretanto o plano deixa o TR em grande vantagem.

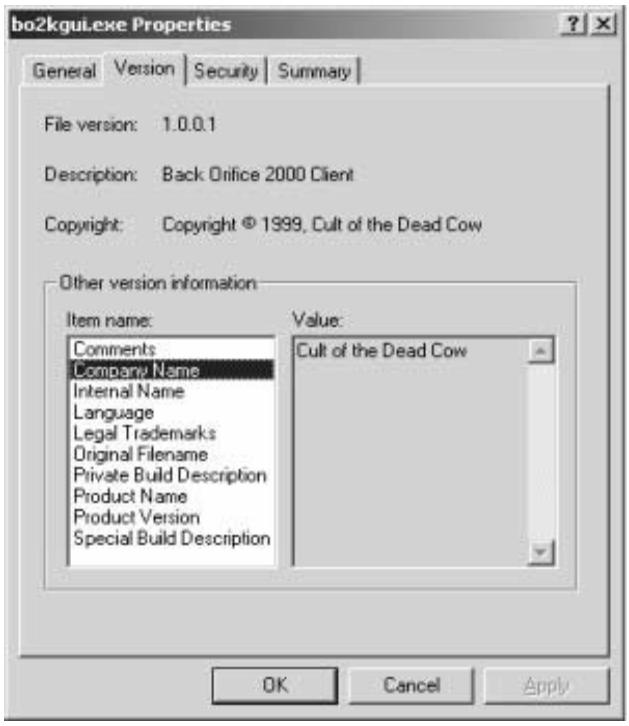

Figura 1: Informação sobre versão.

Vários procedimentos são bastante discutidos e utilizados internacionalmente. Sendo assim, devem ser considerados durante a preparação da metodologia. Dentre os mais importantes pode-se citar:

- Documentação das ações tomadas: todas as ações e decisões tomadas durante a abordagem da máquina vítima devem ser documentadas sem exceção. O rigor na documentação do processo permite uma futura avaliação da resposta.
- Cálculo de *hashes*: assim como a documentação, outro procedimento que deve ser seguido com rigor é o cálculo e armazenamento seguro (e.g. CD com acesso restrito) do hash de cada informação coletada; dessa forma pode-se comprovar a integridade das evidências futuramente.
- Coleta de informações voláteis: utilizando o CD de ferramentas previamente criado deve-se executar os programas necessários para se coletar o maior número de informações voláteis possível, uma vez que só haverá uma chance de fazê-lo. As evidências colhidas devem ser armazenadas de maneira segura.
- Reporte aos órgãos competentes: caso o incidente seja causado por um agente externo, o TR deve entrar em contato com os órgãos de resposta a incidente a fim de obter ajuda, ou apenas registrar a ocorrência do mesmo, eviando o trecho dos *logs* que identifica o ocorrido. No Brasil o NBSO (*NIC Brazilian Security Office*. http://www.nic.br/nbso.html) é o órgão mais indicado. É sabido o quanto vale a reputação de uma organização, por isso o TR deve julgar muito bem a necessidade do envolvimento da polícia no caso.
- Desligamento da máquina: O desligamento da máquina vítima deve ser feito de forma não convencional para se evitar a possibilidade de execução de algum programa destrutivo. Apenas cortar a fonte de alimentação de energia pode ser uma saída.

- Cópia dos discos: A coleta preliminar de informações (*live analysis*) na maioria das vezes não é suficiente para a solução do evento, por isso deve ser seguida de uma análise posterior (*postmortem*) minuciosa. Os discos devem ser copiados bit a bit (*dd*) e seus *hashes* armazenados. Isto para que:
  1. Os discos da máquina possam ser liberados, se for o caso;
  2. A análise *postmortem* possa ser feita sobre cópias, garantindo assim, que um erro não venha a comprometer o original.

4. Conclusões

A complexidade envolvida no trato com incidentes de segurança somadas ao nervosismo causado pela iminência de prejuízos morais e financeiros, justificam a criação de um TR, que por sua vez, estará encarregado da definição dos procedimentos a serem adotados no caso de uma emergência. Apesar de raras as corporações que possuem programas de resposta, o constante aumento da importância da Internet no cotidiano dos mais diferentes tipos de organização, tende a consolidar tal prática.

5. Referências Bibliográficas

[1] AUGUSTO, Alessandro, GUIMARAES, Célio and de GEUS, Paulo Lício; *Administration of Large Windows NT Network with DoIt4Me*; Proc. of SANS'2001, the 10th International Conference on System Administration, Networking and Security, May, 2001;

[2] CARPENTER, Jeffrey J.; *Welcome to The Big City*; ;login: The Magazine of USENIX & SAGE; Novembro 1999;

[3] CARVEY, Harla; *System Security Administration for NT*; Proc. of USENIX LISA-NT, The 3rd Large Installation System Administration of Windows NT Conference; julho 2000;

[4] FARMER, Dan; VENEMA, Wietse; *Forensic Computer Analysis: An Introduction*; Dr. Dobb's Journal; setembro 2000; http://www.ddj.com/articles/2000/0009/0009f/0009f.htm;

[5] HAASE, Norman; *Computer Forensics: Introduction to Incident Response and Investigation of Windows NT/2000*; SANS Intitute; dezembro 2001; http://rr.sans.org/incident/comp_forensics3.php;

[6] LEE, Henry; PALMBACH, Timothy; MILLER, Marilyn; *Henry Lee's Crime Scene Handbook*; Academic Press;

[7] MANDIA, Kevin; PROSISE, Chris; *Incident Response:Investigating Computer Crime*; Osborne/McGraw Hill; 1ª Edição; 2001;

[8] NOBLETT, Michael G.; POLLITT, Mark M.; PRESLEY, Lawrence A.; *Recovering and Examining Computer Forensic Evidence*; Forense Science Communications, outubro 2000, Vol. 2 N. 4; Federal Bureau of Investigation;

[9] OLIVEIRA, Flávio; GUIMARÃES, Célio; REIS, Marcelo; GEUS, Paulo; *Forense Computacional: Aspectos Legais e Padronização*; Anais do Wseg'2001 (I Workshop de Segurança em Sistemas Computacionais); Florianópolis - Brasil;

[10]van WYK, Kenneth R.;FORNO, Richard; *Incident Response*; O'Reilly; 1ª Edição; 2001;