Técnicas e Ferramentas Utilizadas em Análise Forense

Arnaldo Candido Junior
Almir Moreira Saúde

Prof. Dr. Adriano Mauro Cansian
Coordenador

ACME! Computer Security Research Labs
UNESP - Universidade Estadual Paulista
Campus de São José do Rio Preto

Roteiro

- Introdução.
- Metodologia para forense.
- Preparação, coleta e análise.
- Coleta e análise a partir de comandos nativos do UNIX.
- Ferramentas TCT, TCTUtils e TASK.
- Outras ferramentas.

Introdução

- Ciência Forense:
  - "A aplicação de princípios das ciências físicas ao direito na busca da verdade em questões cíveis, criminais e de comportamento social para que não se cometam injustiças contra qualquer membro da sociedade" [1].
- Forense computacional:
  - "O uso de métodos científicos para preservação, coleta, restauração, identificação, documentação e apresentação de evidências digitais" [2].

[1] Manual de Patologia Forense do Colégio de Patologistas Americanos
[2] Forensic Science Comunications

Evidência digital

- Evidências digitais são informações em formato digital capazes de determinar se um sistema computacional sofreu uma violação, ou que provêem uma ligação com a vítima ou com o atacante.
  - Evidências desta natureza podem ser duplicadas com exatidão.
  - É possível verificar se sofreram alterações com os métodos adequados.
  - São altamente voláteis, podendo ser alteradas durante a análise, caso as devidas precauções não sejam tomadas.
- Principio da Troca de Locard.
  - Toda a pessoa que passa pela cena de um crime, deixa algo de si e leva algo consigo.
  - De forma análoga, toda a pessoa que comete um crime digital, deixa rastros no sistema comprometido. Os rastros podem ser difíceis de serem seguidos, mas existem.

Forense Computacional

- Exames forenses tradicionais, como o exame de DNA, são realizados através de métodos e procedimentos bem definidos.
  - A análise é efetuada através de passos rotineiros, repetidos em cada caso.
- Geralmente, a estratégia para forense computacional é particular em cada caso.
  - Heterogeneidade de softwares.
  - Heterogeneidade de *hardwares*.
  - Uso de padrões distintos.
  - Constantes mudanças na tecnologia.
- Os métodos para a forense computacional devem ser genéricos o suficiente para acomodar todas essas mudanças.

Métodos e procedimentos

- Simplificam o processo de coleta, armazenamento e análise de evidências.
- Minimizam o pânico e reações negativas em circunstâncias em que a perícia é conduzida sobre níveis elevados de estresse, evitando um possível comprometimento das evidências.
- Contribuem para validar as evidências coletadas em um processo criminal.
- Necessitam de uma fase de planejamento para sua correta aplicação.

Resposta a Incidentes

- Metodologia de resposta em 6 passos (*SANS Institute*):
  - Preparação: envolve o planejamento e definições de políticas para lidar com o incidente quando detectado.
  - Identificação: caracterização da ameaça e seus efeitos nos sistemas afetados.
  - Contenção: consiste em limitar o efeito do incidente de modo que atinja o menor número de sistemas possíveis.
  - Erradicação: estágio no qual as conseqüências causadas pelo incidente são eliminadas ou reduzidas.
  - Recuperação: retomada das atividades em andamento antes do incidente ocorrer. Também restauração de dados caso necessário.
  - Continuação: medidas necessárias para evitar que a ocorrência do incidente seja repetida.

Metodologia para a perícia

- Coleta de informações.
- Reconhecimento das evidências.
- Restauração, documentação e preservação das evidências encontradas.
- Correlação das evidências.
- Reconstrução dos eventos.

Preparação (1)

- Definições de políticas a serem seguidas e ações a serem tomadas durante a perícia.
- Medidas preventivas para evitar o comprometimento do sistema computacional.
- Monitoramento para detectar incidentes quando ocorrerem.
- Escolha das ferramentas mais adequadas para coleta e análise de evidências.

Preparação (2)

- Os mecanismos de rede podem fornecer informações valiosas para análise quando corretamente configurados.
  - *Firewalls/Sniffers/*Detectores de Intrusão.
  - Roteadores e *gateways* em geral.
  - Servidores de *DNS* e *Proxy*.
  - Servidores de *backups*.
  - Coletores de fluxos.
- Avisos sobre alterações no sistema podem ser obtidas a partir do uso de ferramentas como o monitorador *Tripwire*.
- É recomendável o uso de um *host* exclusivo para coleta de *logs*. *Logs* locais podem ser facilmente removidos por um atacante com acesso administrativo.

Coleta (1)

- A coleta de evidências é feita principalmente com base nos dados obtidos a partir dos discos rígidos e das demais mídias físicas.
- Caso o sistema ainda esteja em funcionamento, pode-se recuperar evidências adicionais.
  - É recomendável interromper a energia ao invés de desligar o sistema de modo habitual.
    - Permite preservar o estado do sistema (*swap*, arquivos temporários, marcas de tempo nos arquivos, ...).
    - Pode evitar armadilhas programadas para disparar durante o desligamento do sistema.
  - Deve ser observado o tempo de vida de cada evidência.

Coleta “Online”

- Examinar uma parte do sistema irá perturbar outras partes:
  - O simples fato de observar informações de determinados tipos é capaz de alterá-las.
  - As informações devem ser preferencialmente coletadas de acordo com seu tempo de vida.

Tempo de vida

- Registradores, *cache*
- Memória dos periféricos (ex. vídeo)
- Memória RAM
- Tráfego de rede
- Estado do sistema Operacional (processos, usuários logados, conexões estabelecidas, ...)
- Disco Rígido
- Mídias secundária (*cd-roms*, *backups*)

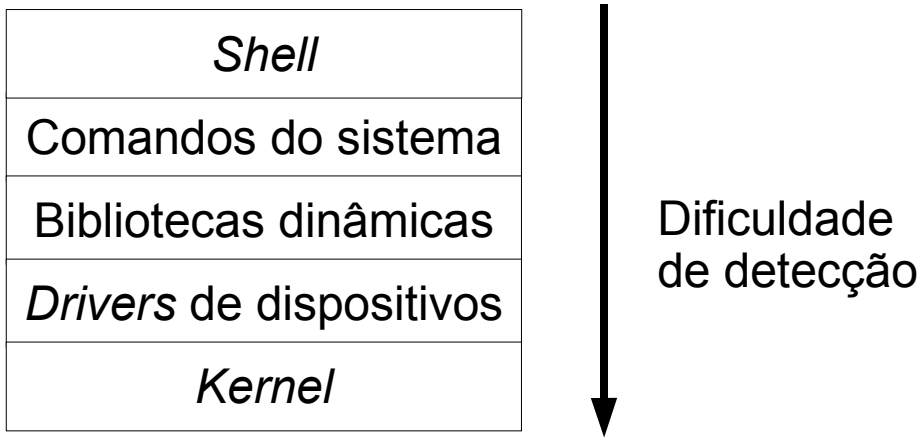
Volatilidade

Dispositivos de armazenamento

- Registradores e *cache*.
  - Contém pouca informação aproveitável.
- Memória de periféricos.
  - Podem possuir informações não disponíveis na memória principal como documentos enviados via *fax*, imagens exibidas no monitor, etc.
- Memória RAM.
  - Contém informações sobre o sistema operacional e os processos em execução. Pode conter senhas e informações em texto plano que estão cifradas no disco.
- Discos Rígidos e mídias secundárias.
  - Contém a maior parte das informações usadas para extração de evidências.

Alterações no sistema

- As informações obtidas podem não ser confiáveis uma vez que o sistema foi comprometido.
  - Como forma de minimizar o problema, alguns *kits* de forense para ambientes UNIX contém binários compilados estaticamente dos principais utilitários de sistema.
- Alvos de modificação:



Análise

- A análise deve ser efetuada sobre uma cópia das mídias originais. As mídias originais devem ser devidamente protegidas.
  - A cópia deve ser *bit* a *bit* com o intuito de preservar arquivos removidos e outras informações.
- As informações coletadas suas respectivas cópias devem ser autenticadas através de assinaturas criptográficas.
- A análise de dados brutos do disco e da memória é excessivamente lenta.
  - O uso de ferramentas para recuperação de arquivos e dump de processos pode agilizar a análise.
- Um ambiente de teste pode ser preparado para auxiliar o procedimento de análise.
  - O *hardware* deve ser preferencialmente similar ao *hardware* do ambiente original.
- Todo o processo deve ser devidamente documentado.

Reconstrução de eventos

- Correlacionamento de *logs*.
- Análise do tráfego da rede (*logs* de roteadores, *firewalls*, ...).
- Histórico do *shell* (quando houver).
- *MAC Times*.
- Recuperação de arquivos apagados ou análise do *dump* do disco.
- Análise de artefatos encontrados no sistema.

Modo de operação do atacante

- Entender o modo de operação é útil durante a busca por evidências:
  - Identificação do alvo.
  - Busca por vulnerabilidades.
  - Comprometimento inicial.
  - Aumento de privilégio.
  - Tornar-se "invisível".
  - Reconhecimento do sistema.
  - Instalação de *backdoors*.
  - Limpeza de rastros.
  - Retorno por um *backdoor*; inventário e comprometimento de máquinas vizinhas.

Ferramentas para monitoração

- Monitorando mudanças no sistema de arquivos.
  - *Tripwire* (http://www.tripwire.com).
  - *Aide* (http://www.cs.tut.fi/~rammer/aide.html).
- Centralizando *logs*.
  - *syslog-ng* (http://www.balabit.hu/en/downloads/syslog-ng).
- Identificação de rootkits.
  - *chkrootkit* (http://www.chkrootkit.org).
- Registro de conexões.
  - *TCPwrapper*.

Coletando dados brutos

- *Dump* da memória:
  - *dd if /dev/mem of <destino>*
  - *dd if /dev/kmem of <destino>*
  - *dd if /dev/rswap of <destino>*
- *Dump* nos discos:
  - *dd if <dipostivo> of <destino>*
- Obtendo dados da memória da placa de vídeo:
  - *xwd -display :0 -root > screen.xwd*
  - *xwud -in screen.xwd*
- Transferindo informações coletadas através do *Netcat:*
  - Servidor: *nc -p <porta> -l > <saída>*
  - Cliente: *<comando_do_sistema> | nc -w 3 to <porta>*

Estado da rede

- Configurações da rede:
  - *ifconfig, iwconfig*
  - *route*
  - *arp*
  - *netstat -tupan, lsof -i*
- Coletando o tráfego:
  - *tcpdump -l -n -e -x -vv -s 1500*
  - *ethereal*

Coletando informações sobre o estado do sistema

- Login de usuários:
  - Usuários atualmente logados: *w, who*
  - Últimos logins efetuados: *last*
  - Último acesso de cada usuário: *lastlog*
- Processos:
  - Processos atualmente em execução: *ps auxeww, ps ealf, ls /proc/*
  - Últimos comandos executados: *lastcomm*
  - Arquivos abertos: *lsof*
- Kernel e módulos:
  - Configurações gerais: *uname -a*
  - Módulos carregados: *lsmod, cat /proc/modules*
  - Módulos ocultos: *kstat -M* (http://www.s0ftpj.org/tools/)

Informações do sistema de arquivos

- *MAC Times*:
  - Horário de último acesso: *ls -altu*
  - Tempo de alteração: *ls -alt*
  - Tempo de mudança nas permissões: *ls -altc*
- Propriedades de um arquivo:
  - *stat <arquivo>*
- Registro de todos os arquivos presentes no sistema:
  - *find / -type f -print0 | xargs -0 md5sum > <saída>*

Análise dos dados coletados

- Buscas nos *dumps* de disco e memória?
  - *strings* e *grep*.
  - Podem encontrar informações em blocos marcados como defeituos.
  - Útil para recuperação de trechos de arquivos apagados, em particular, de *logs* apagados.
- Análise de binários suspeitos
  - Tabela de símbolos: *nm <binário>*, *nm -Du <binário>*
  - Bibliotecas dinâmicas associadas: *ldd <binário>*
  - Visualização em hexadecimal: *cat <binário> | xxd*
  - Chamadas de sistema (em um ambiente de testes): *strace <binário>*
  - Pausando a execução de um processo: *kill -STOP <pid>*

Ferramentas para análise

- *TCT – The Coroner's Toolkit (http://www.porcupine.org/forensics/tct.html):*
  - Conjunto de ferramentas para forense.
- *TCTUtils (http://www.porcupine.org/forensics/tct.html):*
  - Utilitários que provêem funcionalidades extras ao TCP.
- *TASK - The @stake Sleuth Kit (http://www.sleuthkit.org/):*
  - Engloba as funcionalidades do TCT e do TCTUtils para anállise do sistema de arquivos, além de recursos adicionais.
  - Portado para uma série e plataformas.
- *AFB - Autopsy Forensic Browser (http://www.sleuthkit.org/autopsy):*
  - Provê uma interface gráfica para o TASK.

The Coroner's Toolkit (1)

- É composto por quatro partes:
  - *grave-robber*
    - Automatiza a coleta de evidências com os comandos citados anteriormente.
    - Executa ações adicionais (geração de assinaturas criptográficas, lista com arquivos apagados ainda em uso, históricos de shell, ...).
    - A coleta é feita de acordo com a ordem de volatilidade.
  - *mactime*
    - Utiliza informações produzidas pelo grave-robber para criar um histórico de arquivos modificados e acessados em um dado intervalo de tempo.
  - *lazarus*
    - *lazarus*: ferramenta para recuperação de arquivos apagados.
    - *unrm*: efetua *dump* do espaço não alocado no disco.
  - Utilitários
    - Utiliários usados pela ferramenta *grave-roober*.

The Coroner's Toolkit (2)

- Funcionamento do *lazarus:*
  - Sistemas de arquivos *unix* tem baixa fragmentação.
  - Tenta identificar o tipo de arquivo ao qual pertence um espaço não alocado do disdo de 100 *bytes*.
  - Armazena em um arquivo os blocos de dados lidos enquanto o tipo de arquivo identificado não for alterado.
  - Não funciona bem com arquivos grandes.
- Exemplos de utilitários:
  - *pcat*: efetua o dump de um processo na memória.
  - *icat* (ou *inode-cat*): visualiza o conteúdo de um arquivo a partir no número do seu inode. Pode recuperar arquivos apagados ou parte deles.
  - *ils*: lista informações *inodes* de arquivos removidos.

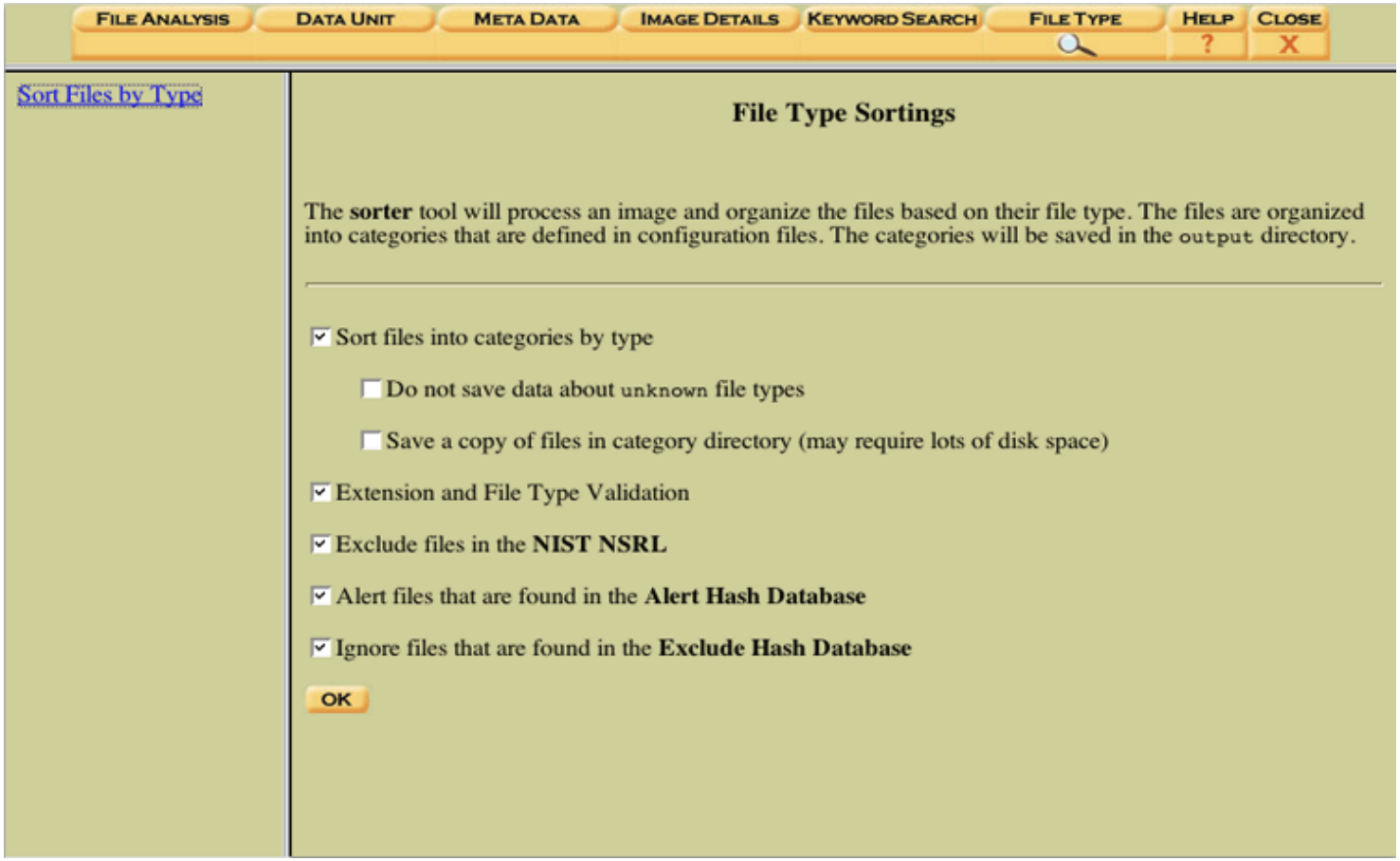
TCTUtils

- Exemplos de utitários:
  - *bcat*: exibe o conteúdo de um bloco de dados presente no sistema de arquivos.
  - *blockcalc*: mapeia blocos do sistema de arquivos orginal com a imagem gerada pela ferramenta *unrm*.
  - *fls*: lista as entradas de um bloco de dados pertencente a um diretório.
  - *find_file*: tenta encontrar o nome de arquivo associado a um *inode*.
  - *find_inode*: tenta encontrar o inode que tem alocado um determinado bloco de dados do sistema de arquivos.
  - *istat*: exibe informações sobre um determinado *inode*.

The @stake Sleuth Kit

- Suporte a sistemas de arquivos de diversos SOs (*BSD*, *Linux*, *Solaris*, *Windows*).
- Exemplos de utilitários:
  - *dstat*: exibe informações sobre um determinado bloco.
  - *fsstat*: informações detalhas sobre o sistema de arquivos em uma determinada partição.
  - *icat*: semelhante ao *icat* do TCT.
  - *dcat*: semelhante ao *bcat* do TCTUtils.
  - Outras ferramenas com funções similares as exibidas anteriormente

Autopsy Forensic Browser



Outras ferramentas úteis (1)

- LiveCDs:
  - *Biatchux Fire (http://biatchux.dmzs.com).*
- *DUMP* de sistemas de arquivos FAT e NTFS:
  - *EnCase (http://www.guidancesoftware.com).*
  - *DriveSpy. DriveSpy (http://www.digitalintel.com).*
  - *Byte Back (http://www.toolsthatwork.com).*
- Ferramentas para recuperação de senhas (*office*, *rar*, *zip*, ...):
  - *Lostpassword (http://www.lostpassword.com).*

Outras ferramentas úteis (2)

- Análise de logs:
  - *Netforensics - (http://www.netforensics.com).*
- Análise de sistemas de arquivos:
  - *Foremost tool. (http://foremost.sourceforge.net).*
- Visualizador de disco e memória em ambientes windows:
  - *Winhex (http://www.winhex.com/).*
- Visualizador de processos em Windows 9x:
  - *Wintop (http://www.dewassoc.com/support/useful/wintop.htm).*

Conclusões

- A Forense computacional é um importante ramo da ciência forense aplicado à coleta de evidências digitais.
- O uso de métodos e procedimentos adequados aumentam a eficiência da coleta, análise e armazenamento de evidências.
- As informações mais voláteis devem ser coletadas primeiro, sendo que não é possível coletar todas as informações presentes em um sistema.
- O sistema operacional fornece diversos mecanismos de contabilidade que podem ser utilizados no processo de coleta de evidências.
- O domínio de ferramentas específicas para forense computacional permite a obtenção de melhores resultados durante a coleta e análise.

Referências

- Anton Chuvakin, Cyrus Peikari. **Security Warrior**. O'Reilly, January, 2004
- Cesar Eduaro Atílio - **Padrão "ACME!" para análise forense de intrusões em sistemas computacionais**.
  - http://www.acmesecurity.org/hp_ng/imagens/download3.jpg
- Dan Farmer, Wietse Venema. **Computer Forensics Analysis Class Handouts**. Agosto, 1999.
  - http://www.trouble.org/forensics/class.html
- Eugene E. Schultz, Russell Shumway. **Incident Response: A Strategic Guide (...)**. Oreilly, November 2001
- Michael G. Noblett et al. **Recovering and Examining Computer Forensic Evidence.**
  - http://www.fbi.gov/hq/lab/fsc/backissu/oct2000/computer.htm
- RecGeus, P. L., Reis, M. A. Análise forense de intrusões em sistemas computacionais: . Anais do I Seminário Nacional de Perícia em Crimes de Informática. Maceió, 2002.
- Christopher Klaus. Compromissed FAQ
  - http://www.faqs.org/faqs/computer-security/compromise-faq/

Obrigado!

Para entrar em contato e obter mais informações:

almir at acmesecurity dot org Key ID: 0x77F86990
arnaldo at acmesecurity dot org Key ID: 0x85A6CA01
adriano at acmesecurity dot org Key ID: 0x3893CD28

Agradecimentos a Cesar Eduardo Atílio
Pelo material cedido para a produção deste trabalho

http://www.acmesecurity.org

ACME! Computer Security Research Labs
UNESP – IBILCE
São José do Rio Preto - Brasil