Jens Nedon, Sandra Frings, Oliver Göbel (Hrsg.)

IT-Incident Management & IT-Forensics

Erste Tagung der Fachgruppe SIDAR der Gesellschaft für Informatik

24. – 25. November 2003 in Stuttgart, Deutschland

Gesellschaft für Informatik 2003

Lecture Notes in Informatics (LNI) - Proceedings
Series of the Gesellschaft für Informatik (GI)

Volume P-39

ISBN 3-88579-368-7
ISSN 1617-5468

Volume Editors
Jens Nedon
ConSecur GmbH
Schulze-Delitzsch-Strasse 2, D-49716 Meppen
Nedon@consecur.de
Sandra Frings
Fraunhofer-Institut für Arbeitswirtschaft und Organisation IAO
Nobelstraße 12, D-70569 Stuttgart
Sandra.Frings@iao.fhg.de
Oliver Göbel
RUS-CERT (Universität Stuttgart)
Breitscheidstr. 2, D-70174 Stuttgart
Goebel@CERT.Uni-Stuttgart.DE

printed by Köllen Druck+Verlag GmbH, Bonn

Post Mortem Analysen mit OpenSource Software

Matthias Hofherr

GeNUA Gesellschaft fuer Netzwerk- und
Unix-Administration mbH
Domagkstraße 7
D-85551 Kirchheim
Matthias_Hofherr@GeNUA.de

Abstract: Forensische Analysen mit freien OpenSource Programmen sind seit Dan Farmers und Wietse Venemas ersten und einzigen Forensik-Kurs und der darauf folgenden Veröffentlichung des Coroner's Toolkits [1] im Jahr 1999 kein unbekannter Faktor mehr. Allerdings mussten sich die OpenSource-Lösung von jeher den Vorwurf gefallen lassen, im Gegensatz zu ihrer kommerziellen Konkurrenz umständlich bedienbar und nur für Freunde der Kommandozeile anwendbar zu sein. Mit Brian Carriers Autopsy [2]/Sleuth Kit [3] Programmkombination soll diese Lücke geschlossen werden. Dieser Beitrag befasst sich mit dem professionellen Einsatz dieser Programme bei forensischen Laboranalysen und den neuen Möglichkeiten, die mit der grafischen Oberfläche Autopsy verbunden sind.

1 Einsatz von OpenSource Software in der Vergangenheit

Nachdem Systemadministratoren jahrelang mit Hilfe von Systemtools und selbstgeschriebenen Programmen Einbrüche in Computersysteme analysiert hatten, stellten Dan Farmer und Wietse Venema im August 1999 zum ersten Mal ein einheitliches Toolset zur Analyse von kompromittierten Systemen vor. Dieses Toolset wurde unter dem Namen „**The Coroner's Toolkit**" (TCT) bekannt. Das TCT stellt dem forensischen Experten Programme zur Beweissicherung und zur späteren Laboranalyse zur Verfügung. Im Lauf der folgende Jahre wurde das TCT zu einem Quasi-Standard, wenn für eine forensische Analyse OpenSource Software herangezogen wurde.

Das Coroner's Toolkit wurde nach-und-nach durch Zusatzprogramme erweitert. Brian Carrier implementierte seine „**tctutils**" [4], eine Reihe von Zusatzprogrammen die das TCT um die Möglichkeit ergänzte, Analysen auch auf Dateinamen-Ebene durchzuführen. Ebenso erschuf Carrier mit „**Autopsy**" ein webbasiertes Frontend für die Kommandozeilen-Programme. Etwa zeitgleich portierte Knut Eckstein des TCT für HP-UX 10.20 und 11.00 [5].

Bei der Analyse mehrerer kompromittierter Systeme erwies sich TCT dahingehend als problematisch, dass pro analysiertem Betriebssystem eine eigene Version des TCT auf dem identischen Betriebssystem benötigt wurde. Carrier kombinierte TCT und tctutils mit einigen neuen Features und stellte damit „**The @stake Sleuth Kit**" (TASK) zur Verfügung. TASK erlaubt dem forensischen Experten die plattformunabhängig Analyse

von Dateisystemen. Unter anderem wird mit TASK auch die Analyse der Windows-Dateisysteme unterstützt.

Das Toolkit wurde mit Version 1.70 umgetauft auf „**The Sleuth Kit**“ (TSK). Damit wird es unabhängig von @stake und anderen Institutionen von Brain Carrier weiterentwickelt.

2 Das Sleuth Kit

Das Sleuth Kit (SK) stellt dem forensischen Experten eine Sammlung von Analyseprogrammen für die Laboranalyse von Bitstream-Images auf Dateisystemebene zur Verfügung. Im Gegensatz zum Coroner's Toolkit sind die Programme nicht dafür gedacht, ein Live-System zu analysieren. Unterstützte Dateisysteme sind gegenwärtig (Sleuth Kit 1.62) Ext2/Ext3, Berkeley FFS, FAT und NTFS. Zahlreiche der neueren Journaling Dateisysteme wie z.B. ReiserFS, XFS und JFS werden noch nicht unterstützt.

SK arbeitet auf verschiedenen Ebenen. Auf der **Meta-Daten-Ebene** werden die Meta-Daten des Dateisystems analysiert, z.B. Inodes oder Master File Table (MFT). Die Meta-Daten umfassen alle beschreibenden Informationen für einen Eintrag im Dateisystem. Auf der **Dateinamen-Ebene** zeigt SK Informationen zu den Namen von Dateien und Verzeichnissen an. Diese Namen werden bei den meisten Dateisystemen getrennt von den Meta-Informationen verwaltet. Auf der **Dateisystem-Ebene** werden Informationen analysiert, die spezifisch für das jeweilige Dateisystem sind. Zuletzt können Roh-Daten auch auf der **Daten-Ebene** analysiert werden. Dies erlaubt die Begutachtung einzelner Datenblöcke unabhängig von vorgegebenen Dateistrukturen.

Für jede dieser Ebenen bietet SK eigene Analyseprogramme [6]:

Ebene	Programm
Meta	ils, icat, istat, ifind
Dateinamen	fls, ffind
Dateisystem	fsstat
Daten	dls, dcat, dstat, dcalc

*ls-Tools liefern ebenenspezifische Auflistungen zurück, *cat zeigt Inhalte an, *stat liefert detaillierte Informationen zu einem Objekt dieser Ebene, *find erzeugt ein Mapping zwischen verschiedenen Ebenen und *calc führt Berechungen ebenenspezifisch durch.

Programme, die von SK zur Verfügung gestellt werden, aber nicht in obiges Muster passen sind:

- mactime: stellt die Datei-Zugriffszeiten anhand ihrer MAC (**M**odification|**A**ccess|**C**hange)-Zeiten auf einer Zeitachse dar

- hfind: durchsucht Hash-Datenbanken auf Hash-Muster (siehe Kapitel 4.3)
- sorter: sortiert Dateien nach Kategorien (siehe Kapitel 4.4)

3 Autopsy

Autopsy stellt eine grafische Oberfläche für die Kommandozeilen-Programme des Sleuth Kits dar. Die Oberfläche ist mit einem herkömmlichen Web-Browser zu bedienen. Autopsy benötigt keinen externen Webserver, sondern ist selbst ein Mini-Webserver. Aus Sicherheitsgründen deaktiviert Autopsy aktive Inhalte und externe Links in Webseiten, die auf kompromittierten Systemen zu finden sind. HTML-Seiten werden komplett als Text dargestellt. Optional kann die HTML-Seite interpretiert betrachtet werden. Allerdings sind hier ebenfalls aktive Inhalte und Hyperlinks deaktiviert.

Autopsy ist in Perl geschrieben und wird unter der GPL veröffentlicht. Dies erlaubt dem Anwender, eigene Modifikationen und Ergänzungen schnell und problemlos vorzunehmen.

4 Die Laboranalyse mit Sleuth Kit und Autopsy

Das Sleuth Kit in Verbindung mit Autopsy wurde in den neueren Versionen gezielt dahingehend verbessert, um dem Anwender die Arbeit zu erleichtern und viele schematische Arbeitsschritte in einer konsequenten Reihenfolge über die Weboberfläche auszuführen. Damit wird die OpenSource Software eine aktive Konkurrenz für kommerzielle Analyse-Produkte, die sich bisher durch eine integrierte, einheitliche Oberfläche abheben.

Während Anwender von Autopsy es früher gewohnt waren, das Programm über eine ASCII-Textdatei zu konfigurieren, wird das neue Autopsy komplett über die Weboberfläche konfiguriert und bedient. Alle Konfigurationsdateien liegen aber weiterhin im ASCII-Format vor und erlauben so manuelle Eingriffe.

4.1 Case Management

Mit der Einführung von Autopsy 1.70 stehen dem forensischen Experten zum ersten Mal Case-Management Möglichkeiten zur Verfügung. Autopsy verwaltet beliebig viele verschiedene Fälle gleichzeitig, wobei sich ein Fall aus der Datensammlung aller untersuchten Systeme, der Spezifikation der beteiligten forensischen Ermittler und den spezifischen Autopsy-Konfigurationen zusammensetzt. Damit kann ein Fall komplett auf ein zweites Autopsy-Analysesystem übernommen und die Beweise dort verifiziert werden.

Jedem Fall können verschiedene kompromittierte Systeme zugeordnet werden, die Bestandteil der Untersuchung werden sollen.

Abbildung 4.1: Autopsy Host Gallery

Für jedes untersuchte System werden die gesicherten Bitstream-Images mit dem passenden Mountpoint und dem zugehörigen Dateisystem-Typ zugeordnet.

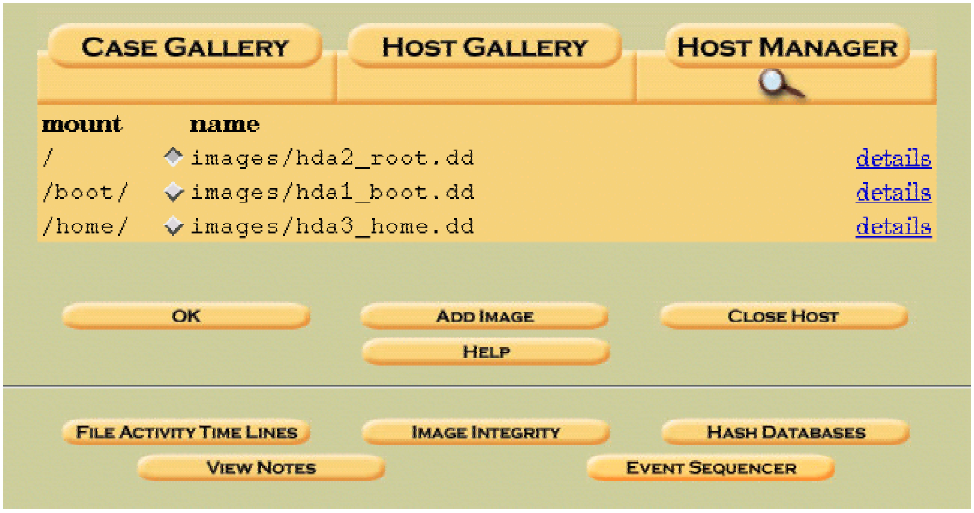

Abbildung 4.2: Autopsy Host Manager

Einziger begrenzender Faktor ist hier die Größe der Festplatte des Analysesystems.

Für jedes definierte System kann ein eigener Bearbeiter angegeben werden. Dies wird später beim Erzeugen der Analyse-Reports benötigt. In der momentan aktuellen Autopsy-Version werden noch keine zentralen Reporte serverseitig gespeichert. Diese

Funktionalität ist für zukünftige Autopsy-Versionen aber bereits angekündigt. Gegenwärtig lassen sich Teilreporte bereits browserseitig abspeichern.

Ein Problem bei der Analyse und Event-Korrelation mehrerer Hosts stellen die Zeitangaben der Systeme dar. Unter Umständen stehen die Systeme in verschiedenen Zeitzonen oder die Uhren laufen nicht exakt. Hier bietet Autopsy die Möglichkeiten, pro System eine eigene Zeitzone anzugeben, sowie eine Zeitabweichung in Sekunden von der korrekten Zeit. Alle in Autopsy dargestellten Zeitangaben berücksichtigen diese modifizierten Zeitangaben.

Sämtliche Funde und Konfigurationsdateien zu einem Fall liegen in einem eigenen Verzeichnis mit dem Namen des Falls. Dies erlaubt eine einfache Sammlung und Archivierung sämtlicher Beweise zu einem Fall. Alle Konfigurationen liegen als ASCII-Text-Dateien vor und können somit einfach verifiziert und verändert werden.

Alle von Autopsy verwendeten und erzeugten Dateien werden mit md5 [7] Hashsummen versehen und können jederzeit über die grafische Oberfläche verifiziert werden. Dies ermöglicht es dem Anwender zu erkennen, ob Beweise versehentlich oder absichtlich modifiziert wurden.

4.2 Suchfunktionen

Autopsy unterstützt den Anwender bei gezielten Suchen innerhalb der Beweisdaten. Um die Suche zu beschleunigen, erlaubt Autopsy die Erstellung von Strings-Dateien aller Images. Dies vermeidet, dass bei jeder Suche das gesamte Bitstream-Image erneut nach einem Suchterm untersucht werden muss. Stattdessen wird nur noch die Strings-Datei ausgewertet. Ebenso lässt sich per Knopfdruck eine Datei mit allen nicht-allokierten Datenblöcken eines Analyse-Images erzeugen und mit einer Strings-Datei versehen.

Autopsy bietet standardmäßig zwei vordefinierte Suchfunktionen an. Hierbei handelt es sich um die Suche nach IP-Adressen und die Suche nach Datumsangaben. Mittels einer zentralen Konfigurationsdatei lassen sich sehr einfach eigene Standard-Suchfunktionen in Autopsy integrieren.

Für generische Suchen steht ein eigenes Interface zur Verfügung, das Suchen mit regulären Ausdrücken erlaubt. Hier liegen die „extended regular expressions“ des grep-Kommandos zugrunde. Die Suchen können wahlweise auf dem Bitstream-Image oder auf einer Datei mit allen nicht-allokierten Datenblöcken erfolgen. Die Suchergebnisse lassen sich im Hex/ASCII-Mode analysieren oder als Beweisdatei abspeichern.

4.3 Hash Datenbanken

Sowohl Autopsy als auch SK unterstützen in den neueren Versionen den Dateivergleich mit Hashsummen-Datenbanken. Unter SK kommt hier das Programm hfind in Kombination mit sorter zum Einsatz. hfind prüft eine oder mehrere Hashsummen (md5 oder sha-1 [8]) gegen eine Datenbank. Gegenwärtig besteht diese Datenbank aus einer ASCI-Listen-Datei die mit einer Index-Datei versehen ist. Als Datenquellen werden md5sum, Hashkeeper und NSRL (National Software Reference Library) unterstützt. hfind liefert

als Rückgabewert entweder „0“ (nicht gefunden) oder „1“ (gefunden) bzw. Bei Bedarf ausführlichere Angaben zur Datei..

4.4 Sorter

Sorter sortiert alle Dateien aus einem Bitstream-Image nach Kategorien. Dies erleichtert eine methodische Auswertung der Dateien. Sorter unterstützt per default 12 Kategorien, allerdings können eigene Kategorien frei definiert werden. Beispiele für solche Kategorien sind Audio, Bilder, Videos und ausführbare Dateien. Sorter überprüft darüber hinaus auch die Datei-Endung mit dem erkannten Dateityp und meldet Diskrepanzen. Die Dateierkennung geschieht aufgrund der „magic numbers“ der Dateien, wie sie mit dem dem „file“-Kommando abgefragt werden können. Um die anfallende Datenmenge zu reduzieren kann ein Vergleich mit Hash-Datenbanken vorgenommen werden. Dieser Vergleich erlaubt die Angabe von Positiv-Listen (bekannte Dateien, die zum Umfang einer Standard-Installation gehören) und Negativ-Listen (bekannte Dateien von Rootkits, Trojanern u.Ä.). Positiv-Vergleiche werden bei der Sortierung ausgeblendet, Negativ-Vergleiche eigens markiert. Die dabei erzielte Daten-Reduktion erlaubt eine wesentlich zeiteffizientere Analyse eines kompromittierten Systems, als dies bisher möglich war.

5 Ergänzende Analyse Programme

5.1 Dateisystem-Analyse

Neben dem Sleuth Kit stehen einem forensischen Experten ergänzende OpenSource Programme zur Dateisystemanalyse zur Verfügung. Als erstes Beispiel sei hier **bmap** [9] genannt. Bmap erlaubt unter ext2 Dateisystemen unter anderem die Analyse von Slackspace[2]. Normalerweise gibt es unter ext2 Dateisystemen keinen verwertbaren Slackspace, da dieser bei einer Allokierung mit Nullen überschrieben wird. Allerdings ist es einem Angreifer durchaus möglich, diesen Slackspace im Nachhinein mit eigenen Daten zu befüllen. Eine Analyse mit bmap verrät, ob Slackspace in dieser Art und Weise genutzt wird und gibt den Inhalt aus.

Eine weitere Form der Dateisystemanalyse kann mit Programmen aus dem TCT durchgeführt werden. Mit Hilfe des TCT-Programms "unrm" oder mittels Autopsy lässt sich aus einem Bitstream-Analyse-Image ein weiteres Image mit allen nicht allokierten Datenblocks des kompromittierten Systems erzeugen. Das TCT-Programm "**lazarus**" nimmt dieses Image als Basis zur Erkennung und Wiederherstellung von gelöschten Dateien. Lazarus versucht, die Datenblöcke zu klassifizieren und sie dann sinnvoll zu vollen Dateien zusammenzusetzen. [10] Dieser Vorgang benötigt allerdings erheblich Zeit und Festplatte-Platz.

Foremost [11] ist ein ergänzendes Programm zu TSK, geschrieben von den AFOSI (Air Force Office of Special Investigations) Special Agents Kendall und Kornblum. Es er-

[2] Slackspace: Der freie Platz zwischen Dateiende und dem Ende des Datenblocks.

möglicht die Suche nach Dateien in einem Bitstream Image basierend auf vorgegebenen, und vom Anwender frei modifizierbaren Headern und Footern von Zieldateien. Momentan enthält foremost ca. 40 vorgegebene Muster. Brian Carrier hat bereits seine Absicht angekündigt, foremost in zukünftigen Versionen von Autopsy zu integrieren.

5.2 Steganographie

Steganographie erlaubt das Verstecken von Daten in Grafik- und Tondateien. Diverse freie Programme machen diese Technik für Jedermann verfügbar. Damit steigt die Wahrscheinlichkeit, im Rahmen einer forensischen Analyse auf derart versteckte Daten zu treffen. Das Programm **stegdetect** [12] von Niels Provos bietet die Möglichkeit, alle Grafikdateien eines kompromittierten Systems auf Spuren von Steganographie zu untersuchen. In Kombination mit dem Programm sorter (siehe Kapitel 4.4) bietet es sich an, sämtliche Dateien der Sorter-Gruppe „Grafik“ mit stegdetect zu überprüfen.

5.3 Viren

Während der Analyse eines kompromittierten Systems sollte eine Überprüfung erfolgen, ob das System von Viren, Würmern oder Ähnlichem befallen wurde. Im OpenSource-Bereich bietet sich hier allerdings nur eine eingeschränkte Auswahl. Beispiele für Open-Source Virenscanner sind Clam Antivirus [19] oder der ScannerDaemon des Open AntiVirus Projects [20]. Aufgrund der zum Teil sehr frühen Projekt-Stati dieser OpenSource Virenscanner stellt sich die Frage, ob sie im Verlauf einer professionellen forensischen Untersuchung zum Einsatz gebracht werden sollten. Ein kommerzielles Produkt bietet hier unter Umständen bessere Treffer-Raten und eine seriösere Form der Beweisführung.

Wünschenwert ist die Möglichkeit, den Virenscanner sowohl bei einer Live-Analyse einsetzen zu können als auch bei der Labor-Analyse. Für eine Live-Analyse muss der Scanner auf eine CD bzw. Floppy passen und darf nicht auf Betriebssystem-Komponenten, wie dynamisch ladbare Bibliotheken, zugreifen. Darüber hinaus darf kein schreibender Zugriff auf das untersuchte Medium erfolgen, damit nicht Beweise zerstört werden für eine anschliessende Labor-Analyse.

5.4 Pornographie

Sollte sich im Verlauf einer forensischen Untersuchung herausstellen, dass nach Beweisen für Kinder- oder Tierpornographie gesucht werden muss, dann kommt entweder eine sehr aufwändige, manuelle Untersuchung aller Bilddateien in Frage oder der Einsatz eines kommerziellen Scanner-Systems. Als Beispiel für einen solchen Scanner ist Perkeo++ [21] zu nennen. Das Programm wurde in Zusammenarbeit mit deutschen Strafverfolgungsbehörden erstellt und arbeitet auf der Basis von sog. „elektronischen Fingerabdrücken“.

Perkeo++ erleichtert einem Ermittler in jedem Fall die manuelle Suche erheblich. Der Hersteller gibt die Fehlerwahrscheinlichkeit mit „weniger als $1:10^{34}$“ an. Um Bilder zu

erkennen, die nicht in der Datenbank gespeichert sind, wird der Ermittler allerdings nicht umhin kommen, sämtliches Bildermaterial im Rahmen einer Labor-Anaylse zu sichten.

5.5 Eigene Erweiterung: Hashsummen-DB

Gegenwärtig bieten Sleuth Kit und Autopsy die Möglichkeit eines Hashsummen-Vergleichs auf „Known-Good“ / „Known-Bad“ – Basis (siehe Kapitel 4.3). Die bestehende Lösung auf Basis von ASCII-Dateien mit Index-Ergänzung erweist sich im Alltag als relativ umständlich. Es existieren gegenwärtig zahlreiche Quellen für Hashsummen. Beispiele hierfür sind:

- NIST NSRL [13]
- Known Goods Database [14]
- Hashkeeper [15]
- Solaris Fingerprint Database [16]
- The Cyberabuse RootkID Project [17]
- Dan Farmer’s FUCK Baseline-Sammlung [18]
- Selbsterstellte Baselines mit md5sum, sha1sum oder grave-robber

Jede dieser Datenquellen hat ein anderes Format. Von hfind unterstützt werden momentan NSRL, Hashkeeper und md5sum. Der Untersuchungs- und Pflegeaufwand kann hier aber stark vereinfacht werden durch eigene Anpassungen. Ziel dieser Anpassungen ist es, die verschiedenen Datenquellen in einem RDBMS (Relationales Datenbank Management System) zu vereinheitlichen. Mit Hilfe eines Import-Parserskripts kann die Datenbank bestückt und gepflegt werden. Ein Skript hfind_db ersetzt das bestehende hfind-Programm und vergleicht Hashsummen direkt gegen die Datenbank. Eine Anpassung von sorter sorgt dafür, dass das neue Programm hfind_db anstatt hfind aufgerufen wird um bekannte Betriebssystemdateien auszublenden. Damit lässt sich die neue Datenbank auch direkt von Autopsy aus verwenden.

6 Fazit

Die Qualität einer forensischen Analyse hängt sowohl vom Know-How des forensischen Experten als auch von der Güte der verwendeten Programme ab. OpenSource Programme zur forensischen Analyse brauchen den Vergleich mit ihrer kommerziellen Konkurrenz heute nicht mehr zu scheuen. Klare Vorteile gegenüber kommerziellen Analyseprogrammen haben OpenSource-Programme in puncto einfacher Erweiterbarkeit und guter Nachvollziehbarkeit von Analyse-Ergebnissen. Insbesondere das Zweigespann Sleuth Kit/Autopsy setzt hier Maßstäbe. Einfache, schematische Aufgaben werden durch die grafische Oberfläche stark vereinfacht. Bei komplexeren Probleme, die sich über die Oberfläche nicht so einfach lösen lassen, steht die volle Mächtigkeit der Kommandozei-

len-Programme zur Verfügung. Zu guter letzt lassen sich beide Programme einfach erweitern und auf die persönlichen Bedürfnisse und Präferenzen des Anwenders anpassen.

Nichtsdestotrotz sollte der Einsatz von OpenSource Software nicht dogmatisch betrachtet werden. Gerade im Bereich von Viren- und Pornographie-Scannern weißt die kommerzielle Konkurrenz Produkte vor, für die es keine gleichwertigen Gegenstücke im OpenSource Bereich gibt. Ziel sollte sein, das jeweils optimale Produkt für eine bestimmte Aufgabe einzusetzen. Durch die Vielzahl hervorragender forensischer Open-Source Programme ist dieses „optimale Produkt“ mittlerweile allerdings häufig im OpenSource Lager zu finden.

7 Referenzen

[1] The Coroner's Toolkit: www.porcupine.org/forensics/tct.html
[2] Autopsy Forensic Browser: www.sleuthkit.org/autopsy/
[3] The Sleuth Kit: www.sleuthkit.org/sleuthkit/
[4] tctutils: www.cerias.purdue.edu/homes/carrier/forensics/
[5] TCT für HP-UX: www.isd.uni-stuttgart.de/~knut.eckstein/tct-hp.html
[6] Brian Carrier. The Sleuth Kit Informer – February 2003. http://www.sleuthkit.org/informer/sleuthkit-informer-1.html
[7] R. Rivest. The MD5 Message-Digest Algorithm. RFC1321,1992
[8] D. Eastlake. US Secure Hash Algorithm 1 (SHA1). RFC3174,2001
[9] bmap: ftp://ftp.scyld.com/pub/forensic_computing/bmap/
[10] Dan Farmer, Wietse Venema. Doctor Dobb's Journal: Bring out out your Dead. http://www.ddj.com/documents/s=871/ddj0101h/0101h.htm
[11] foremost: http://foremost.sourceforge.net
[12] stegdetect: www.outguess.org/detection.php
[13] National Software Reference Library: www.nsrl.nist.gov
[14] Known Goods Database: www.knowngoods.org
[15] Hashkeeper Database: http://www.hashkeeper.org
[16] Solaris Fingerprint Database: http://sunsolve.sun.com/pub-cgi/fileFingerprints.pl
[17] The CyberAbuse Rootk(it)ID Project: http://rk.cyberabse.org/?page=infos
[18] MD5-Summen Archive: www.fish.com/fuck
[19] Clam AntiVirus: http://sourceforge.net/projects/clamav/
[20] Open AntiVirus Project: http://www.openantivirus.org
[21] Perkeo++: http://www.perkeo.net